

Complexity

Social Big Data: Mining, Applications, and Beyond

Lead Guest Editor: Xiuzhen Zhang

Guest Editors: Shuliang Wang, Gao Cong, and Alfredo Cuzzocrea





Social Big Data: Mining, Applications, and Beyond

Complexity

Social Big Data: Mining, Applications, and Beyond

Lead Guest Editor: Xiuzhen Zhang

Guest Editors: Shuliang Wang, Gao Cong, and Alfredo Cuzzocrea



Copyright © 2019 Hindawi. All rights reserved.

This is a special issue published in “Complexity.” All articles are open access articles distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Editorial Board

- José A. Acosta, Spain
Carlos F. Aguilar-Ibáñez, Mexico
Mojtaba Ahmadih Khanesar, UK
Tarek Ahmed-Ali, France
Alex Alexandridis, Greece
Basil M. Al-Hadithi, Spain
Juan A. Almendral, Spain
Diego R. Amancio, Brazil
David Arroyo, Spain
Mohamed Boutayeb, France
Átila Bueno, Brazil
Arturo Buscarino, Italy
Guido Caldarelli, Italy
Eric Campos-Canton, Mexico
Mohammed Chadli, France
Émile J. L. Chappin, Netherlands
Diyi Chen, China
Yu-Wang Chen, UK
Giulio Cimini, Italy
Danilo Comminiello, Italy
Sara Dadras, USA
Sergey Dashkovskiy, Germany
Manlio De Domenico, Italy
Pietro De Lellis, Italy
Albert Diaz-Guilera, Spain
Thach Ngoc Dinh, France
Jordi Duch, Spain
Marcio Eisencraft, Brazil
Joshua Epstein, USA
Mondher Farza, France
Thierry Floquet, France
Mattia Frasca, Italy
José Manuel Galán, Spain
Lucia Valentina Gambuzza, Italy
Bernhard C. Geiger, Austria
Carlos Gershenson, Mexico
Peter Giesl, UK
Sergio Gómez, Spain
Lingzhong Guo, UK
Xianggui Guo, China
Sigurdur F. Hafstein, Iceland
Chittaranjan Hens, Israel
Giacomo Innocenti, Italy
Sarangapani Jagannathan, USA
Mahdi Jalili, Australia
Jeffrey H. Johnson, UK
M. Hassan Khooban, Denmark
Abbas Khosravi, Australia
Toshikazu Kuniya, Japan
Vincent Labatut, France
Lucas Lacasa, UK
Guang Li, UK
Qingdu Li, Germany
Chongyang Liu, China
Xiaoping Liu, Canada
Xinzhi Liu, Canada
Rosa M. Lopez Gutierrez, Mexico
Vittorio Loreto, Italy
Noureddine Manamanni, France
Didier Maquin, France
Eulalia Martínez, Spain
Marcelo Messias, Brazil
Ana Meštrović, Croatia
Ludovico Minati, Japan
Ch. P. Monterola, Philippines
Marcin Mrugalski, Poland
Roberto Natella, Italy
Sing Kiong Nguang, New Zealand
Nam-Phong Nguyen, USA
B. M. Ombuki-Berman, Canada
Irene Otero-Muras, Spain
Yongping Pan, Singapore
Daniela Paolotti, Italy
Cornelio Posadas-Castillo, Mexico
Mahardhika Pratama, Singapore
Luis M. Rocha, USA
Miguel Romance, Spain
Avimanyu Sahoo, USA
Matilde Santos, Spain
Josep Sardanyés Cayuela, Spain
Ramaswamy Savitha, Singapore
Hiroki Sayama, USA
Michele Scarpiniti, Italy
Enzo Pasquale Scilingo, Italy
Dan Selişteanu, Romania
Dehua Shen, China
Dimitrios Stamovlasis, Greece
Samuel Stanton, USA
Roberto Tonelli, Italy
Shahadat Uddin, Australia
Gaetano Valenza, Italy
Dimitri Volchenkov, USA
Christos Volos, Greece
Zidong Wang, UK
Yan-Ling Wei, Singapore
Honglei Xu, Australia
Yong Xu, China
Xinggang Yan, UK
Baris Yuce, UK
Massimiliano Zanin, Spain
Hassan Zargarzadeh, USA
Rongqing Zhang, USA
Xianming Zhang, Australia
Xiaopeng Zhao, USA
Quanmin Zhu, UK

Contents


Social Big Data: Mining, Applications, and Beyond

Xiuzhen Zhang , Shuliang Wang , Gao Cong, and Alfredo Cuzzocrea
Editorial (2 pages), Article ID 2059075, Volume 2019 (2019)

A Multi-Granularity Backbone Network Extraction Method Based on the Topology Potential

Hanning Yuan, Yanni Han , Ning Cai, and Wei An
Research Article (8 pages), Article ID 8604132, Volume 2018 (2019)

Behavior-Interior-Aware User Preference Analysis Based on Social Networks

Can Wang, Tao Bo, Yun Wei Zhao , Chi-Hung Chi, Kwok-Yan Lam, Sen Wang, and Min Shu
Research Article (18 pages), Article ID 7371209, Volume 2018 (2019)

Supervised Learning for Suicidal Ideation Detection in Online User Content

Shaoxiong Ji , Celina Ping Yu, Sai-fu Fung, Shirui Pan , and Guodong Long 
Research Article (10 pages), Article ID 6157249, Volume 2018 (2019)




Weibo Attention and Stock Market Performance: Some Empirical Evidence

Minghua Dong, Xiong Xiong, Xiao Li, and Dehua Shen 
Research Article (8 pages), Article ID 9571848, Volume 2018 (2019)

A Trip Purpose-Based Data-Driven Alighting Station Choice Model Using Transit Smart Card Data

Kai Lu , Alireza Khani, and Baoming Han 
Research Article (14 pages), Article ID 3412070, Volume 2018 (2019)


A Methodology for Evaluating Algorithms That Calculate Social Influence in Complex Social Networks

Vanja Smailovic , Vedran Podobnik , and Ignac Lovrek 
Research Article (20 pages), Article ID 1084795, Volume 2018 (2019)



Self-Adaptive K -Means Based on a Covering Algorithm

Yiwen Zhang , Yuanyuan Zhou , Xing Guo , Jintao Wu, Qiang He, Xiao Liu , and Yun Yang
Research Article (16 pages), Article ID 7698274, Volume 2018 (2019)

Robust Semisupervised Nonnegative Local Coordinate Factorization for Data Representation

Wei Jiang , Qian Lv, Chenggang Yan, Kewei Tang, and Jie Zhang
Research Article (16 pages), Article ID 7963210, Volume 2018 (2019)

AIRank: Author Impact Ranking through Positions in Collaboration Networks

Jun Zhang, Yan Hu , Zhaolong Ning, Amr Tolba , Elsayed Elashkar, and Feng Xia 
Research Article (16 pages), Article ID 4697485, Volume 2018 (2019)

Research of Deceptive Review Detection Based on Target Product Identification and Metapath Feature Weight Calculation

Ling Yuan , Dan Li , Shikang Wei , and Mingli Wang 
Research Article (12 pages), Article ID 5321280, Volume 2018 (2019)

Editorial

Social Big Data: Mining, Applications, and Beyond

Xiuzhen Zhang ¹, **Shuliang Wang** ², **Gao Cong**,³ and **Alfredo Cuzzocrea**⁴

¹RMIT University, Australia

²Beijing Institute of Technology, China

³Nanyang Technological University, Singapore

⁴University of Trieste, Italy

Correspondence should be addressed to Xiuzhen Zhang; xiuzhen.zhang@rmit.edu.au and Shuliang Wang; slwang2011@bit.edu.cn

Received 28 October 2018; Accepted 5 December 2018; Published 1 January 2019

Copyright © 2019 Xiuzhen Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The social nature of Web 2.0 leads to the unprecedented growth of discussion forums, product review sites, microblogging, and other social media platforms. Existing social media data mining research can be broadly divided into two groups. The content-based approach focuses on extracting insights from user generated contents on various social media platforms. The network-based approach focuses on extracting knowledge by analyzing the networks from the interactions among online users.

The rich user- and device-generated data and user interactions generate complex social big data that is different from classical structured attribute-value data. The data objects take various forms including unstructured text, geo-tagged data objects, and data object streams. The social networks formed from interactions among data objects also carry rich information for analyzing user behavior.

In this special issue, we have invited state-of-the-art research contributions addressing prominent research issues for social big data to advance our knowledge in social big data mining and analytics and extend the knowledge to related disciplines. We received 20 submissions from across the world. After a rigorous reviewing process, we finally accepted 10 papers. The accepted papers address challenging issues for the social big data technology, ranging from novel data mining applications from complex data and general methodological machine learning models to network analysis and evaluation.

(i) Three papers proposed advanced data mining techniques for novel applications using user- and device-generated data, including “Supervised Learning for

Suicidal Ideation Detection in Online User Content”, “Weibo Attention and Stock Market Performance: Some Empirical Evidence”, and “A Trip Purpose-Based Data-Driven Alighting Station Choice Model Using Transit Smart Card Data”.

- (ii) Two machine learning methodological papers for cluster analysis and data representation learning are included, namely, “Self-Adaptive k -Means Based on a Covering Algorithm” and “Robust Semisupervised Nonnegative Local Coordinate Factorization for Data Representation”.
- (iii) Three papers reported research results on social network analysis for information credibility and social influence, ranging from “Research of Deceptive Review Detection Based on Target Product Identification and Metapath Feature Weight Calculation” and “Behavior-Interior-Aware User Preference Analysis Based on Social Networks” to “AIRank: Author Impact Ranking through Positions in Collaboration Networks”.
- (iv) Two papers, “A Multi-Granularity Backbone Network Extraction Method Based on the Topology Potential” and “A Methodology for Evaluating Algorithms That Calculate Social Influence in Complex Social Networks”, address the under investigated issues of network summarization and social influence evaluation. The research results can benefit network analysis in general and social network analysis specifically.

In the modern digital society, the mobile network and the Internet of Things are transforming what is meant to be social online. Humans, everyday objects, and smart devices interact and form an intelligent social network that is a highly adaptive complex system. The papers in this special issue are mainly contributed by the data science, machine learning, and network science communities. Research results in these papers highlight the wide range of complex research issues for the social big data research. Looking ahead, we call for research from other disciplines such as human-computer interaction, pervasive computing and computational social science to work together with the data science community to advance social big data research.

Last but not least, we would like to express our deep gratitude to reviewers for their valuable contributions that improve the quality of papers in this special issue.

Conflicts of Interest

The authors declare that they do not have any conflicts of interest.

*Xiuzhen Zhang
Shuliang Wang
Gao Cong
Alfredo Cuzzocrea*

Research Article

A Multi-Granularity Backbone Network Extraction Method Based on the Topology Potential

Hanning Yuan,¹ Yanni Han ,^{2,3} Ning Cai,^{2,3} and Wei An^{2,3}

¹International School of Software, Beijing Institute of Technology, Beijing 100081, China

²Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China

³School of Cyber Security, University of Chinese Academy of Sciences, Beijing 100049, China

Correspondence should be addressed to Yanni Han; hanyanni@iie.ac.cn

Received 25 December 2017; Accepted 11 October 2018; Published 22 October 2018

Guest Editor: Xiuzhen Zhang

Copyright © 2018 Hanning Yuan et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Inspired by the theory of physics field, in this paper, we propose a novel backbone network compression algorithm based on topology potential. With consideration of the network connectivity and backbone compression precision, the method is flexible and efficient according to various network characteristics. Meanwhile, we define a metric named compression ratio to evaluate the performance of backbone networks, which provides an optimal extraction granularity based on the contributions of degree number and topology connectivity. We apply our method to the public available Internet AS network and Hep-th network, which are the public datasets in the field of complex network analysis. Furthermore, we compare the obtained results with the metrics of precision ratio and recall ratio. All these results show that our algorithm is superior to the compared methods. Moreover, we investigate the characteristics in terms of degree distribution and self-similarity of the extracted backbone. It is proven that the compressed backbone network has a lot of similarity properties to the original network in terms of power-law exponent.

1. Introduction

Complex networks hide a variety of relationships among members of complex systems. Recently the driving application is motivated by discovering knowledge and rules hidden in complex systems using network mining method [1, 2]. It has been found in complex network to reveal some unique statistical characteristics and dynamics features, such as agglomeration and network evolution. However, the increasingly large network data and huge network scale pose an urgent challenge to understand network characteristics from the global perspective. Extracting backbones from large-scale network will contribute to understanding the network topology and identifying kernel members, which is a pressing problem for various applications in practice.

Taking the field of sociology, for example, when we study the collaborations among scientists, social network can be described at different granularities shown in Figure 1. Smyth.net is a publication network centered with Dr. Padhraic Smyth [3]. Figure 1(a) presents the co-authorship network with famous computer scientist Padhraic Smyth as

the core. If they collaborate with other authors to write a paper, then an edge exists between them. The Smyth publication network consists of 286 nodes and 554 edges. With the increment of granularity, we can regard the scientific group as a node and collaborations between scientific groups as edges. Then the network topology consists of 71 nodes shown in Figure 1(b). Furthermore, if the granularity keeps increasing, the universities or research institutions of scientists are defined as nodes, and the collaborations between them are defined as edges, the core network structure consists of 17 nodes simplified in Figure 1(c). Therefore, motivated by the same problem, different granularities determine different scale of the network topology. In order to describe complex networks in the real world, it is inevitable to observe the topology properties from different perspectives, such as large nodes at fine-grained or little nodes at coarse-grained. In particular, the focus problem depends on the mining granularity and the expected knowledge space.

Therefore, research on backbone extraction is to explore the core element structures without loss of the topology properties. The backbone extraction achieves data acquisition

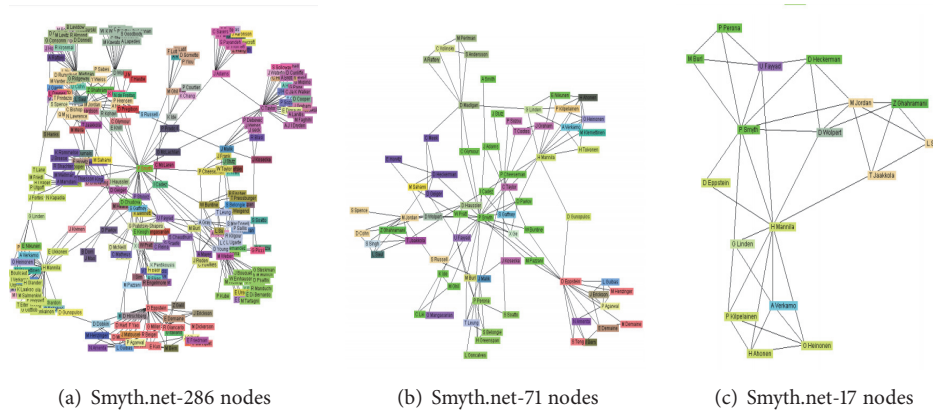


FIGURE 1: Multi-granularities of the Smyth publication network [3].

and process, data reduction, network compression, and other steps. By obtaining backbone structures and analyzing the extracted backbone network, it can help to discover the evolution process, which provides valuable contributions for the fields of biology, physics, and computer science.

In this paper, we introduce the topology potential model to solve the backbone network extraction problem and describe the nodes joint interaction. Based on the topology potential model, an algorithm is proposed to extract backbone network from large-scale networks. To detect the optimal backbone extracting granularity, an evaluation metric based on topology connectivity is presented. We choose the public Internet autonomy system network and the Hep-th network as the experiment available datasets. Through the evaluation with precision ratio and recall ratio, our proposed backbone extraction algorithm is proved to be more effective compared to the baselines.

The remainder of this paper is organized as follows. In Section 2 we briefly introduce the background and motivation. Then the backbone extraction model is detailed in Section 3. In Section 4 we present an algorithm to detect the backbone network based on topology potential. Section 5 is devoted to the analysis of the experiment results from different views. Conclusion appears in Section 6.

2. Background

In this section, we conclude the backbone extraction problem as two parts, application and algorithm.

From the point view of application, current research works focus on the improvement of the previous graphics or network simplification methods. By applying the research results of complex networks in recent years, it will contribute to the actual engineering compared with the superiority of new methods and understanding them in more simplified forms. For example, based on edge betweenness and edge information, Scellato devised a method to extract the backbone of a city by deriving spanning trees [4]. Hutchins detected the backbones in criminal networks in order to target suspects exactly [5]. Also urban planners attempted

to examine the topologies of public transport systems by analyzing their backbones [6].

In terms of backbone extraction algorithm, main researches are aimed at the large-scale network. Most work emphasizes the efficiency of compression algorithm, the structure analysis of the backbone topology, and the comparison between the extracted backbone and the actual backbone of the network. Nan D proposed a method of mining the backbone network in a social network [7]. In order to obtain the backbone network with minimum spanning tree, it needs to find all the clusters in the network. The algorithm complexity is mainly focused on searching all clusters. Hence, the applicability of the algorithm depends on the scale of clusters in the network. In 2004, Gilbert C. proposed a novel network compression algorithm [8] including two important parts, i.e., importance compression and similarity compression. Because the mining backbone is fixed, the experiment results show that this method has a high precision, but the recall rate is very low.

In short, the current researches have some shortcomings about these algorithms. It is known that extracting the backbone structure must be guided with a certain rule, such as the numbers of clusters, or the importance of network nodes, etc. Therefore, the structure of backbone network is fixed and the recall rate is usually low. The filtering technology based on the weight distribution of edges is able to obtain backbone networks with different sizes. However, the filter-based methods often suffer from the computational inefficiency, which is quite expensive during the exhaustive search of all nodes or edges [9–11].

3. Backbone Extraction Model

In this section, to solve the uncertainty of different granularities backbones, we introduce the topology potential theory to measure the backbone network topology. Furthermore, to validate an optimal backbone with the most suitable granularity, we define a metric named compression ratio and discuss the extraction performance.

3.1. Inspired by the Topology Potential. According to the field theory in physics, the potential in a conservative field is a function of position, which is inversely proportional to the distance and is directly proportional to the magnitude of particle's mass or charge. Inspired by the above idea, we introduce the theory of physical field into complex networks to describe the topology structure among nodes and reveal the general characteristic of underlying important distribution [12].

Given the network $G = (V, E)$, V is the set of nodes and E is the set of edges. For $\forall u, v \in V$, let $\varphi_v(u)$ be the potential at any point v produced by u . Then $\varphi_v(u)$ must meet all the following rules:

- (i) $\varphi_v(u)$ is a continuous, smooth, and finite function;
- (ii) $\varphi_v(u)$ is isotropic in nature;
- (iii) $\varphi_v(u)$ monotonically decreased in the distance $\|v-u\|$. When $\|v-u\|=0$, it reaches maximum, but does not go infinity, and when $\|v-u\| \rightarrow \infty$, $\varphi_v(u) \rightarrow 0$.

So the topology potential can be defined as the differential position of each node in the topology, that is to say, the potential of node in its position. This index reflects the ability of each node influenced by the other nodes in the network, and vice versa. In essence the topological potential score of each node can reflect nodes importance in the topology by optimizing influence factor, which can reveal the ability of interaction between nodes in the network.

There are many kinds of field functions in physics, such as gravitational field, nuclear force field, thermal field, magnetic field, etc. From the scope of field force, we can classify two types, short-range fields and long-range fields. The range of the former fields is limited and forces decrease sharply as the distance increases, while the latter is just the other way. As the characteristics of small-world and modularity structure imply that interactions among nodes are within the locals in real-world network, each node's influence will quickly decay as the distance increases in accordance with the properties of short-range fields. Meanwhile, owing to the limited scopes of short-range among nodes in the topology structure, it is feasible to ignore the iterated calculation of topology potential far away from the influence range. By this way, we can reduce the cost and computing complexity effectively. Hence, we define the topology potential in the form of Gaussian function, which belongs to the nuclear force field. The potential of node $V_i \in V$ in the network can be formalized as

$$\varphi(V_i) = \sum_{j=1}^n \left(m_j \times e^{-\frac{d_{ij}}{\sigma}} \right) \quad (1)$$

where d_{ij} is the distance between node V_i and V_j ; the parameter σ is used to control the influence region of each node and called influence factor; and $m_i \geq 0$ is the mass of node V_i ($i=1..n$), which meets a normalization condition $\sum_{i=1}^n m_i = 1$.

In order to measure the uncertainty of topological space, potential entropy has been presented to be similar to the essence of information entropy. Intuitively, if each node's

topology potential value is different, then the uncertainty is the lowest accounting for the smallest entropy. So a minimum-entropy method can be used for the optimal choice of influence factor σ . This way is more reasonable and without any pre-defined knowledge. Given a topological potential field produced by a network $G=(V, E)$, let the potential score of each node V_1, \dots, V_n be $\varphi(V_1), \dots, \varphi(V_n)$, respectively; a potential entropy H can be introduced to measure the uncertainty of the topological potential field, namely,

$$H = - \sum_{i=1}^n \frac{\varphi(V_i)}{Z} \log \left(\frac{\varphi(V_i)}{Z} \right) \quad (2)$$

where Z is a normalization factor. Clearly, for any $\sigma \in (0, +\infty)$, potential entropy H satisfies $0 \leq H \leq \log(n)$ and H reaches the maximum value $\log(n)$ if and only if $\varphi(V_1) = \varphi(V_2) = \dots = \varphi(V_n)$.

3.2. Definition of the Backbone Network. Backbone network consists of hub nodes and important edges. The hub nodes are nodes with great influence in the topology network, which can be measured by the values of topology potential. Generally, the edges connected by these hub nodes are also important. In the process of extracting backbone network, whether to add these edges to backbone network is determined by the network connectivity.

Definition 1 (hub nodes). For the given parameter α ($0 \leq \alpha \leq 1$), the nodes whose topology potential values are ranked in Top α are the hub nodes to be extracted. The extraction of backbone networks is divided into two steps:

(1) Find the hub nodes as the original backbone members, denoted by *source*. As this step is completed, each isolated node in *source* is an island subnet.

(2) Find the bridge ties to connect those island subnets and join the ties to the *source*. Loop the two operations until *source* is connected. We define the distance between two island subnets as follows:

$$\begin{aligned} dist(subg1, subg2) \\ = \min_{v1 \in subg1, v2 \in subg2} |shortestpath(v1, v2)| \end{aligned} \quad (3)$$

where v_1 and v_2 are arbitrary nodes of subnets *subg1* and *subg2*, respectively. The connection is added by the shortest distance between the two subnets when we extract the connections of backbone. If the shortest distance is 1, the bridge tie is added directly to connect the subnet. Otherwise, the connection is added between the subnet and the corresponding neighbor node which has the largest topology potential value in all the neighbor nodes. Intuitively the distance between the two island subnets is very likely to be reduced.

3.3. Metrics of the Reduction Effectiveness. According to the specific attributes of nodes, we can calculate the topology properties of all nodes in the original network and sort them in descending order. For the arbitrary node v of generated

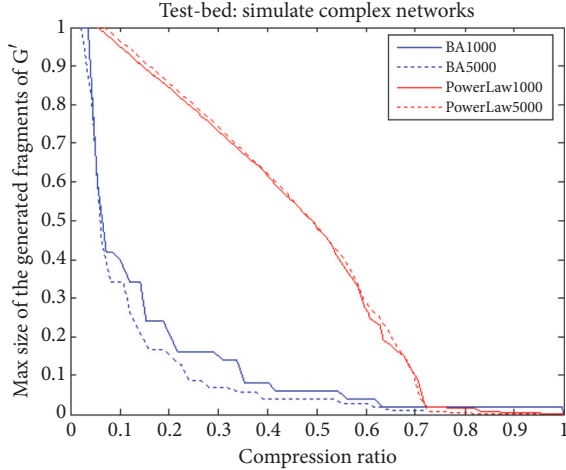


FIGURE 2: The changing trend for different size of the isolated subnet.

network with different scales, $rank(v)$ denotes its sorting value in the backbone network and $Rank(v)$ denotes its sorting value in the original network. The measurement $coverage(v)$ is defined in

$$coverage(v) = \frac{rank(v)}{Rank(v)} \quad (4)$$

where $coverage(v)$ denotes the coverage that backbone network nodes cover the important nodes of the whole network. The larger the $coverage(v)$ value, the higher the accuracy and the better the quality of the extracted backbone network. The overall quality of the backbone network depends on the distribution of $coverage(v)$ values for all nodes. The expected $coverage(v)$ of all nodes is used to evaluate the overall performance of the backbone network. $Compress_ratio$ is defined in

$$compress_ratio = \frac{\sum_{v \in V(\text{backbone})} coverage(v)}{|V(\text{backbone})|} \quad (5)$$

The most important metric of backbone networks is the available compression ratio, which is related to the network scale. If the size of the isolated subnet in G' is small enough, then the probability of the backbone member is small and the network G' has collapsed after removing the backbone from G . Based on the model of BA and the *Eppstein Power law* simulated by computer, we build the experimental networks at different scales to study the effective compression ratio. It is observed that the compression ratio changes of $lar_subgs_size(G')$ are shown in Figure 2. When the compression ratio $compress_ratio$ is large enough, then the size of maximum isolated subnet $lar_subgs_size(G')$ changes very little.

4. The Backbone Network Detect Algorithm

The traditional backbone compression scheme is divided into the importance based on node and the shortest path. The former considers that the larger the degree, the more

important the node. The weight of a node is defined as (6). Considering the definition focuses on global elements and the density is too large, the node weight is defined as shown in formula (7).

$$w_{deg}(v) \equiv \frac{|\{u \in V : deg(u) \leq deg(v)\}|}{|V|} \quad (6)$$

$$w_{beta}(v) \equiv \frac{|\{u \in N(v) : deg(u) \leq \beta \cdot deg(v)\}|}{|N(v)|} \quad (7)$$

where β is a parameter and $N(v)$ is the set of nodes connected to v .

The definition of node importance based on the shortest path considers that the greater the number of nodes, the greater the importance of nodes. The definition of weight is shown in

$$w_{path}(v) \equiv \sum_{x,y \in V} \frac{|\{\pi \in \Pi(x,y) : v \in \pi\}|}{|V|^2 |\Pi(x,y)|} \quad (8)$$

where $\Pi(x,y)$ is the shortest path between node x and node y .

4.1. Extraction Process. In this paper, we propose an algorithm to extract backbone network with specific granularities according to user's requirement, which is independent of network topology structure. The practical procedure includes two steps. In the first step, the initial hub node set $H1$ according to the topology potential of nodes is found. Secondly, the path is added based on the shortest path till the network is connective, and finally the backbone network is generated.

A detailed description of these algorithm is given in Algorithm 1.

4.2. Discussion of the Algorithm Complexity. The shortest paths between all nodes in the network are calculated by using the breadth first search method. The time complexity is $O(|V||E|)$ for undirected networks. The time complexity of calculating the topology potential of each node is $O(|V||E|)$. Search backbone connections until the network is connective. The average shortest path length of the network is $avg(Sp)$. The original subnet number of source is $\alpha|V|$. To make the original subnets connected, the backbone network is a tree structure, which means we need at least search $O(\alpha|V| * |V|avg(Sp))$ links to make the network connective. So the complexity of the algorithm is $O(\max\{\alpha|V|^2 avg(Sp), |V||E|\})$.

5. Evaluation

To assess the efficiency of our backbone extraction approaches, we choose the public available datasets as the experiment dataset. We introduce the datasets briefly.

Internet autonomy system networks (AS) are a collection of routers and links mapped from all ten ISPs with the biggest networks: AT&T, Sprint, and Verio, etc. These real networks are publicly available from [14]. All the data networks have nodes with scale from 600 to 900 and edges


```

Input: network  $G$ ,  $\alpha(0 \leq \alpha \leq 1)$ 
Output: backbone  $B(\alpha)$ 
Matrix  $Sp$ : compute the shortest path length of all pairs of nodes; var  $i = 1$ ;
Evaluate hops: = avg( $Sp$ ); evaluate factor: =  $\sqrt{2}/3 * \text{avg}(Sp)$ ;
Begin:
repeat:
 $i = i + 1$ ;
for each node  $v \in G$ , compute  $\phi^i(v)$ : topology potential within  $i$  hops;
sort( $\phi^i(v), v \in G$ ); source: =  $\emptyset$ ;
for each node  $v \in G$ ,
if  $\phi^i(v), v \in G$  rank Top  $\alpha$ , source := source  $\cup \{v\}$ ;
repeat:
for each pair of island subnets  $subg1, subg2 \in \text{source}$ ,
if distance between  $subg1$  and  $subg2$  is the shortest,
if distance( $subg1, subg2$ ) = 1
merge( $subg1, subg2$ );
else
find  $neig1 \in subg1, \phi^i(neig1) = \max\{\phi^i(v \in subg1)\}$ , tie1 link  $neig1$  and  $subg1$ 
 $neig2 \in subg2, \phi^i(neig2) = \max\{\phi^i(v \in subg2)\}$ , tie2 link  $neig2$  and  $subg2$ 
source := source  $\cup \{neig1, neig2, tie1, tie2\}$ ;
end if
end if
until network generated from source is connected
 $B(\alpha) := B(\alpha) \cup \text{source}$ ;
until  $i \geq \text{hops}$ 
End

```

ALGORITHM 1: Detecting the backbone network based on topology potential.

with scale from 4000 to 10000. Each of them has about 400 backbone routers.

High-energy physics theory citation network (hep-th) is collected from the e-print arXiv and covers all the citations within a dataset of 27,770 papers with 352,807 edges [15]. If paper i cites paper j , a directed edge is connected from i to j . If a paper cites or is cited by a paper outside the dataset, then the graph does not contain any information about this.

5.1. Compression Ratio. In this paper we take the networks named as3356, as4755, as2914, and as7018 randomly and the numbers of nodes are 1786, 226, 11745, and 6253, respectively. In order to obtain the relevant parameters of backbone networks at different granularity, the number of isolated subnets $\text{cut_subgs}(G')$ obtained by the backbone network under different selection ratios is evenly calculated. For instance, the scale control parameter starts from 0.01 to 1 and the step is set to 0.01. After the backbone network is generated, the compression ratio with the corresponding granularity can be obtained. Figure 3 shows the number of isolated subnets generated by each network with different compression ratios. Each pair of compression ratio and $\text{cut_subgs}(G')$ corresponds to a point on the coordinate system, and the curves are fitted to these points.

It is depicted that fitted curve is monotonically decreasing after increasing at the beginning, as illustrated in Figure 3. When the compression ratio increases to a certain value, the number of generated isolated subnets no longer changes. That is to say, it is no longer effective to compress the network

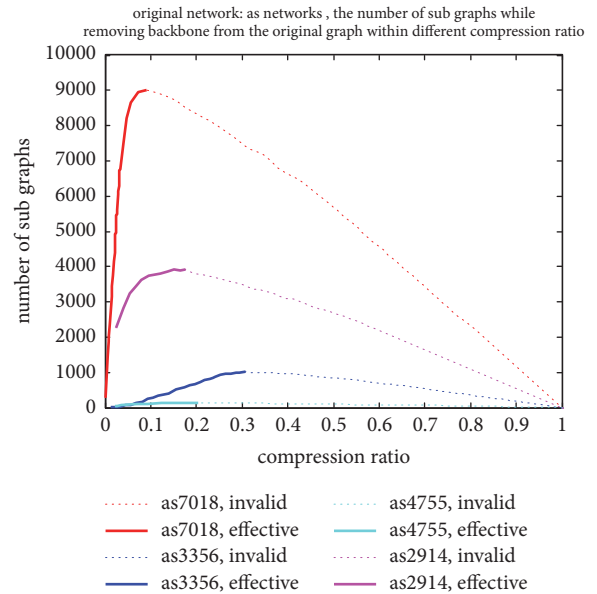


FIGURE 3: The number of generated isolated subnets with different compression ratio.

continuously to reduce the connectivity of the network. The solid line in the fitting curve denotes effective compression, and the dashed part denotes invalid compression. Measuring the performance of backbones networks needs to exclude the situation of invalid compression ratio. In the Internet

TABLE 1: Comparison with the precision ratio and recall ratio [13].

CM	as1239		as2914		as3356		as7018	
	PR	RR	PR	RR	PR	RR	PR	RR
Deg/All	0.91	0.27	0.97	0.19	0.93	0.18	0.91	0.21
Beta/All	0.94	0.35	0.89	0.27	0.97	0.22	0.91	0.24
Path/all	0.95	0.17	1.00	0.14	0.97	0.16	0.96	0.11
TP method	0.77	0.47	0.76	0.28	0.86	0.73	0.61	0.58

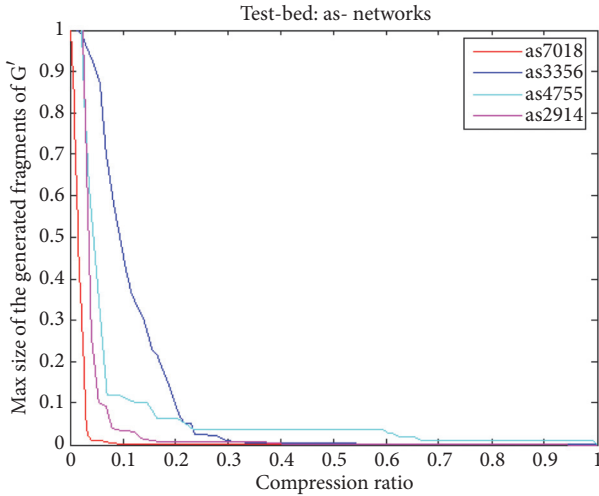


FIGURE 4: The optimal compression ratios of different networks.

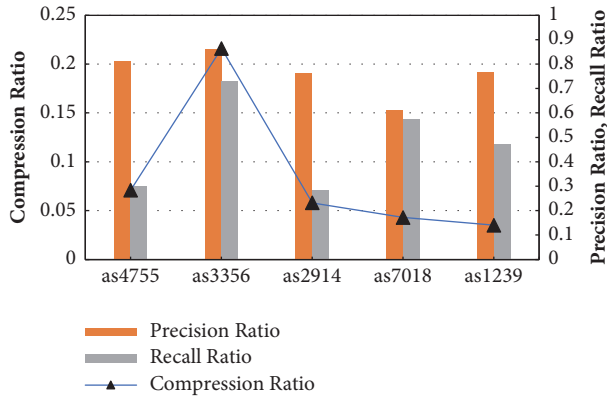


FIGURE 5: The optimal parameters of the extracted Internet mapping network.

mapping results, the optimal compression ratios of the networks as3356, as4755, as2914, and as7018 are about 0.23, 0.16, 0.08, and 0.035, respectively, as illustrated in Figure 4.

5.2. Precision Ratio and Recall Ratio. Measuring the performance of backbone network is to explore the optimal high-performance network metrics. For a large-scale network, it is impossible to calculate the backbone at the whole granularities, as the time complexity will be quite high. Using the binary optimization strategy, when the dichotomous range is small enough, we can determine the maximum effective

compression ratio. For example, if the range is set to 0.01, the search time is $\log_2(0.01) \sim 7$.

After discovering the maximum effective compression ratio, we search the optimal compression ratio and the corresponding optimal backbone network. The Internet mapping network has real backbone node data; thus we can compare the extracted backbone network to verify the extraction results on the real backbone network. The optimal parameters to evaluate the extracted backbone are shown in Figure 5.

Compared with the traditional methods adopted in [7], it is found that these methods can obtain high precision ratios about the value of 0.9, while the recall ratios of the traditional methods are lower than 0.2. On the other hand, the precision ratio of the topology potential extraction method (named TP method) is approximately 0.8 and the recall ratio increased to about 0.5. Since an excellent extraction method requires a higher recall ratio, our method is superior to the traditional methods from this aspect. Other related extraction algorithms do not have real instance verification, and the extraction quality is unknown and lacks verification. Part of the experimental results is listed in Table 1. The abbreviation of compressing method is CM, precision ratio is PR, and the recall ratio is RR.

5.3. Coverage of Backbone with Various Hops. Taking the Hep-th network as experimental data, we analyze the coverage performance of backbone networks with different hops. In this paper, the range of hops is adopted from 2 to 7. Firstly, we take the traditional centrality measurement, degree, betweenness, and closeness to analyze, as shown in Figure 6. We compute the node important properties of the generated backbone network with various hops. The coordinate point indicates the nodes proportions of the backbone network sorted the top i to the nodes of the original network sorted the top $rank_i$, defined as $coverage(i)$. The important attributes are node degree (the upper left), node betweenness (the upper right), node closeness (the lower left), and edge betweenness (the lower right).

The results show that using different centrality metrics to measure the extraction results with various hops has different advantages. For example, when the metric is degree, using 2 hops can get the best extraction effect. When the metric is closeness, using 7 hops can get the best extraction effect. Therefore, in this paper, we use the topology potential to extract backbone networks with specific granularities according to user's requirement, which is independent of network topology structure. We can get the comprehensive results of extracted backbone network.

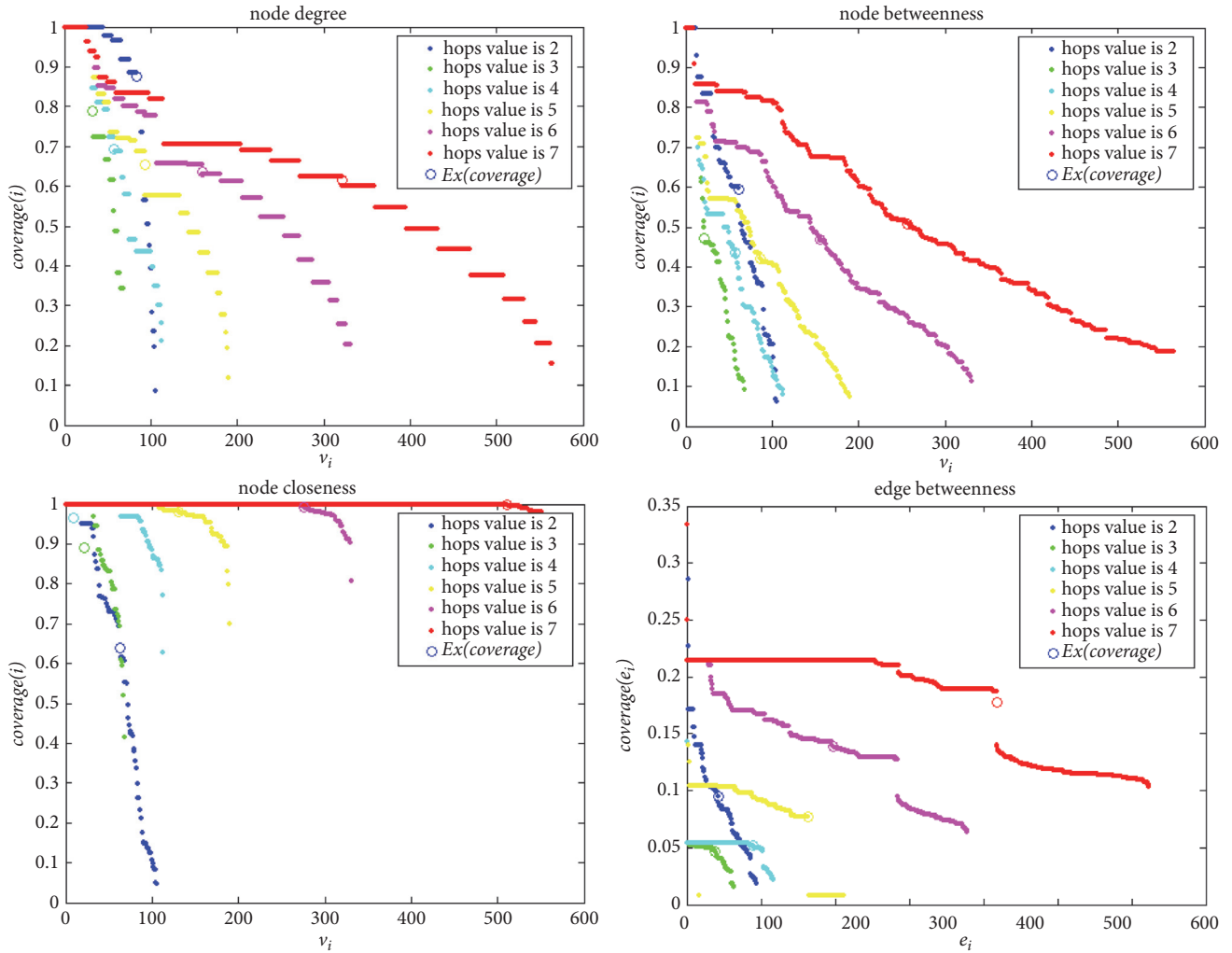


FIGURE 6: The coverage of backbone in various hops.

6. Conclusion

In this paper, we introduced the topology potential to solve the problem of backbone network extraction. Based on the novel topology measurement, an algorithm is proposed to extract backbone networks at different granularities. In order to detect the optimal backbone extraction granularity, an evaluation metric that considers the tradeoff between network connectivity and network properties is presented. By experiments on the public available datasets of Internet AS network and the Hep-th Network, it is proven that the precision ratio and recall ratio to extract the backbone network are superior to current methods. In the future, we will investigate the performance of backbone network at different scale and the dynamic evolution properties.

Conflicts of Interest

There is no conflict of interests related to this paper.

Acknowledgments

This work was supported by National Key Research and Development Plan of China (2016YFB0502600, 2016YFC0803000), National Natural Science Fund of China (61472039), International Scientific and Technological Cooperation and Academic Exchange Program of Beijing Institute of Technology (GZ2016085103), and Frontier and Interdisciplinary Innovation Program of Beijing Institute of Technology (2016CX11006).

References

- [1] R. K. Darst, C. Granell, A. Arenas, S. Gómez, J. Saramäki, and S. Fortunato, "Detection of timescales in evolving complex systems," *Scientific Reports*, vol. 6, 2016.
- [2] A. Holzinger and I. Jurisica, "Knowledge discovery and data mining in biomedical informatics: The future is in integrative, interactive machine learning solutions," in *Interactive knowledge discovery and data mining in biomedical informatics*, pp. 1–18, Springer, Berlin, Germany, 2014.

- [3] A. Y. Wu, M. Garland, and J. Han, "Mining scale-free networks using geodesic clustering," in *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 719–724, August 2004.
- [4] S. Scellato, A. Cardillo, V. Latora, and S. Porta, "The backbone of a city," *The European Physical Journal B*, vol. 50, no. 1-2, pp. 221–225, 2006.
- [5] C. E. Hutchins and M. Benham-Hutchins, "Hiding in plain sight: criminal network analysis," *Computational and Mathematical Organization Theory*, vol. 16, no. 1, pp. 89–111, 2010.
- [6] J. H. Choi, G. A. Barnett, and B.-S. Chon, "Comparing world city networks: a network analysis of Internet backbone and air transport intercity linkages," *Global Networks*, vol. 6, no. 1, pp. 81–99, 2006.
- [7] D. Nan, W. Bin, and W. Bai, "A Parallel Algorithm for enumerating all Maximal Cliques in Complex Networks," in *Proceedings of the The 6th ICDM 2006 Mining Complex Data Workshop*, IEEE Computer Society, pp. 320–324, New Orleans, LA, USA, 2006.
- [8] C. Gilbert and K. Levchenko, "Compressing Network Graphs[C]," in *Proceedings of the Link KDD Workshop at the 10th ACM Conference on KDD*, 2004.
- [9] W. Yao, Y. Yang, and G. Tan, "Recursive Kernighan-Lin algorithm (RKL) scheme for cooperative road-side units in Vehicular networks," *Communications in Computer and Information Science*, vol. 405, pp. 321–331, 2014.
- [10] S. A. Stoev, G. Michailidis, and J. Vaughan, "On global modeling of backbone network traffic," in *Proceedings of the 29th Conference on Computer Communications (INFOCOM '10)*, pp. 1–5, San Diego, CA, USA, March 2010.
- [11] Liqiang Qian, Zhan Bu, Mei Lu, Jie Cao, and Zhiang Wu, "Extracting Backbones from Weighted Complex Networks with Incomplete Information," *Abstract and Applied Analysis*, vol. 2015, Article ID 105385, 11 pages, 2015.
- [12] H. Yanni, H. Jun, L. Deyi, and Z. Shuqing, "A novel measurement of structure properties in complex networks," in *Proceedings of the International Conference on Complex Sciences*, pp. 1292–1297, 2009.
- [13] N. He, W. Gan, and D. Li, "Evaluate Nodes Importance in the Network Using Data Field Theory," in *Proceedings of the 2007 International Conference on Convergence Information Technology (ICCIT 2007)*, pp. 1225–1234, November 2007.
- [14] N. Spring, R. Mahajan, and D. Wetherall, "Measuring ISP topologies with Rocketfuel," *ACM SIGCOMM Computer Communication Review*, vol. 32, no. 4, pp. 133–145, 2002.
- [15] "SNAP network datasets," <https://snap.stanford.edu/data/cit-HepTh.html>.

Research Article

Behavior-Interior-Aware User Preference Analysis Based on Social Networks

Can Wang,¹ Tao Bo,² Yun Wei Zhao ,³ Chi-Hung Chi,⁴ Kwok-Yan Lam,⁵ Sen Wang,¹ and Min Shu³

¹Griffith University, Australia

²Beijing Earthquake Agency, China

³CN-CERT, China

⁴CSIRO, Australia

⁵Nanyang Technological University, Singapore

Correspondence should be addressed to Yun Wei Zhao; zhaoyw@cert.org.cn

Received 28 December 2017; Revised 2 May 2018; Accepted 14 May 2018; Published 9 October 2018

Academic Editor: Xiuzhen Zhang

Copyright © 2018 Can Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

There is a growing trend recently in big data analysis that focuses on behavior interiors, which concern the semantic meanings (e.g., sentiment, controversy, and other state-dependent factors) in explaining the human behaviors from psychology, sociology, cognitive science, and so on, rather than the data per se as in the case of exterior dimensions. It is more intuitive and much easier to understand human behaviors with less redundancy in concept by exploring the behavior interior dimensions, compared with directly using behavior exteriors. However, they usually approach from a unidimensional perspective with a lack of a sense of interrelatedness. Thus, integrating multiple behavior dimensions together into some numerical measures to form a more comprehensive view for subsequent prediction processes becomes a pivotal issue. Moreover, these studies usually focus on the magnitude but neglect the associated temporal features. In this paper, we propose a behavior interior dimension-based neighborhood collaborative filtering method for the top- N hashtag adoption frequency prediction that takes into account the interdependence in temporal dynamics. Our proposed approach couples the similarity in user preference and their impact propagation, by integrating the linear threshold model and the enhanced CF model based on behavior interiors. Experiments on Twitter demonstrate that the behavior-interior-aware CF models achieve better adoption prediction results than the state-of-the-art methods, and the joint consideration of similarity in user preference and their impact propagation results in a significant improvement than treating them separately.

1. Introduction

Under big data era, dynamic behaviors of an entity, human, or object are often revealed through multiple interrelated data sources, each of which gives a “partial view” of the instantaneous behavior of the entity or the context that the entity is currently in. Traditional data mining approaches offer many solutions trying to discover the cooccurrence patterns among multiple data sources, but these solutions often do not emphasize the use of domain knowledge and semantics to uncover the causations behind. From this perspective, we are motivated to categorize the dimensions (or features) used to characterize users/topics in two groups: interior

dimensions and exterior dimensions. They differ in whether the transformation from the raw behavior sequences into a description of them carries semantic meanings (e.g., sentiment, controversy, and other state-dependent factors in explaining human behaviors from psychology, sociology, cognitive science, etc.) or concerns the data per se (e.g., tweet volume, number of users, and other behavior statistics and data representation techniques in computer science summarizing the raw behavioral data captured). Simply put behavior interior dimensions transform the data into the knowledge that domain people are familiar with. While infinite exterior dimensions could be extracted in theory, many of them revolve around the same interior dimensions. For instance,

both the two previously mentioned exterior metrics: tweet volume and number of users, could be viewed as feasible ways of quantifying the so-called interior dimension “virality.” Behavior interiors can be regarded as an aggregation of exterior dimensions. It is believed that better decisions can be made by considering these relevant, interdependent data sources (for example, in Twitter, such interdependent data sources include tweet content, transactional data (e.g., posting time), and follower-following relationship) in the analytics process simultaneously.

Understanding the interior aspects of behaviors is a pivotal issue in various fields. We can think about its value in behavioral biology [1], psychology [2], marketing management [3], and so on. For example, in marketing research [4], rather than based on external transactional dimensions (e.g., amount of purchase and purchase frequency) that are in theory infinite, the factors influencing consumer behavior can be classified into four categories: cultural factors (e.g., basic values and habits from common life experience and situations, such as bargaining or fixed-price preference), social factors (e.g., reference group), personal factors (e.g., economic condition, occupation, and lifecycle), and psychological factors (e.g., motivations, beliefs, and attitudes). Apart from being the key focus in traditional domains such as the previously mentioned psychology and marketing, we note that behavior interior dimensions are also investigated in other domains such as user-generated content- (e.g. Twitter, Facebook) based analysis in political election, stock market trending, and so on. It has received wide attention in box office revenue prediction, stock market trending, political elections [5], and opinion tracking in environmental affairs [6]. In Bollen et al.’s work [7], the authors created Google-Profile of Mood States that measures mood in terms of six dimensions: calm, alert, sure, vital, kind, and happy. Other examples also include happiness (or “bullishness” in stock terms) and controversy (or referred to as “disagreement in stock blogs”); they are the common indicators used in stock market trending analyses [8].

However, behavior interior dimensions are usually studied separately. There is not much research effort to go one step further, to integrate multiple behavior dimensions to form a more comprehensive view for some phenomenon and for subsequent prediction processes. In this paper, we focus on how to utilize the behavior interior dimension-based approach to learn user preference and enhance the prediction of a user’s hashtag adoption behavior. Moreover, these studies usually focus on the magnitude but neglect the associated temporal features. In this paper, we propose a behavior interior dimension-based neighborhood collaborative filtering method for the top- N hashtag adoption frequency prediction. Both the interdependence between multiple behavior interior dimensions and temporal relations are considered in learning user preference from their neighbors (i.e., with high similarity in behavior interior dimensions) to make future predictions. Furthermore, we expand the neighbor sets by considering the users that impact information propagation. We give a coupling mechanism that integrates the linear threshold model and neighborhood CF models in this paper. This work is important because hashtag

adoption is a good indicator for a user’s preference. Once the adoption behavior can be predicted accurately, better understanding about a user’s topic interests can be made. Extensive experimental results evidence that our proposed behavior-interior-aware models achieve significant accuracy improvement, when compared with existing approaches.

We summarize the contributions of our work as follows:

- (i) Firstly, we propose a behavior-interior-aware approach that captures the semantic meaning in the raw behavior traces instead of the exterior transactional features; the effectiveness of the proposed approach is verified empirically using big data of Twitter
- (ii) Secondly, we enhance the prediction accuracy in user-hashtag adoption by learning user preference through a behavior interior-based approach with the interdependence between multiple behavior interior dimensions and temporal relations both considered
- (iii) Thirdly, we offer a Jaccard index-based metric to gauge the difference in interior dimensions and exterior dimension-based approaches in learning users’ preferences to illustrate the effectiveness of the proposed approach
- (iv) Lastly, the explainability of hashtag recommendation models is greatly enhanced with the introduction of the behavior interiors

The rest of paper is organized as follows. We discuss the related work in Section 2. Behavior interior dimensions are defined and captured in Section 3. We describe the proposed models in Section 4. Experiments are extensively evaluated on Twitter in Section 5. Discussions and implications in terms of behavior interior explanations are provided in Section 6. Finally, we conclude this paper and present future work in Section 7.

2. Related Work

The central theme of this paper is the proposal of using behavior interior dimensions to support better hashtag adoption prediction from heterogeneous behavior data which contains various types of data sources that are interdependent on each other. In this section, we will review related research efforts in analytics coping with these issues. The focus and limitations of these approaches will be discussed in detail.

2.1. Data Heterogeneity and Interdependence: Their Ramifications in Analytics. One important aspect of big data research is that these data capture different aspects of human behaviors in different forms [9]. For example, data sources of Twitter include tweet content, transactional data (e.g., posting time), and follower-following relationship. In most cases, these multiple data sources are in various data formats. They may often be variables of completely different types. For example, some are categorical (e.g., hashtag adopted), some are numerical (e.g. tweet amount), some are graph-based

(e.g., in-degree/follower amount), and some are text-based (e.g., sentiment).

To cope with such problem, one approach to this problem is to perform scale conversion [10], i.e., categorization. Categorization methods of numerical data include direct categorization by dividing the range into N intervals, k -means-based categorization, and least squares-based categorization. However, this approach is not satisfactory because there is data loss in the discretization in the scale conversion from numerical to categorical data. Furthermore, additional information (e.g., ordering information) is added in the scale conversion from categorical to numerical data.

This problem becomes more complicated with data interdependence [11]. Very often, an object is not unidimensional, and different multidimensional data may correlate with each other in different aspects [12]. For example, common fate occurs when both dyad members are exposed to the same causal factor [9], and when happiness is doubled, sadness is halved [13]. An alternative method is to carry out separate analyses on the same set of data, with each involving variables from a single data source only [1, 14–16]. Some are based on the transactional statistics (e.g., tweet amount, mention amount) [1], some are based on the content (e.g., TF-IDF) [15], and others are based on the network structure (e.g., in-degree/follower amount) [16]. Those models are limited due to the constraint that multiple data sources are assumed independently.

Moreover, this problem is complicated with data interdependence. Very often, an object is not unidimensional, and different dimensional data may correlate with each other in different aspects. Consider a simple example with three objects: “a red cup,” “a red mouse,” and “a blue keyboard.” “A red cup” is similar to “a red mouse” because of color proximity; “a red mouse” is similar to “a blue keyboard” because of their functional affinity, both are electronic devices. Thus, focusing on the data per se without considering the environment setting and domain knowledge is sometimes problematic. Take the most commonly adopted geometric model-based similarity measures as an example. In these models, each object is represented by a point in some multidimensional coordinate space, and the metric distance between points reflects the similarities between the respective objects. The assumptions made to a distance metric δ in this approach include at least the following three axioms: (a) “minimality,” $\delta(a, b) \geq \delta(a, a) = 0$; (b) “symmetry,” $\delta(a, b) = \delta(b, a)$; and (c) “triangle inequality,” $\delta(a, b) + \delta(b, c) \geq \delta(a, c) \geq |\delta(a, b) - \delta(b, c)|$ [17]. When applied to categorical data (e.g., the example above), these assumptions might not hold. For example, the triangle inequality sets a lower limit to the similarity between a and c in terms of the similarities between a and b and between b and c . However, “a red cup” and “a blue keyboard” are not similar at all in either color proximity or functional affinity, despite the similarity between these two items and “a red mouse.”

The interdependence among users includes intra- and interpersonal types, with extensive research efforts from various domains. Intrapersonal type refers to the situation where a person’s behavior at time t is not independent of his/her behavior at time $t-1$. For example, a user’s web browsing

behavior is usually modeled with a Markov process [18]. Interpersonal type refers to the situation where a person’s behavior is not independent of other people’s behavior. For example, common fate occurs when both dyad members are exposed to the same causal factor [9], and when happiness is doubled, sadness is halved [13].

Behavior interior dimensions integrate multiple data sources that are in various formats and are interdependent on each other together. One example of such behavior interior dimensions is openness. Openness refers to a strong intellectual curiosity or a preference for novelty and variety [19]. Novelty preference is usually measured with time difference between a user that first encounters a hashtag and the user that first adopts this hashtag. Variety preference is usually measured with the number of different hashtags adopted. Of these two measures, hashtag adoption time is timestamp, while hashtags adopted are categorical. The integration of these different measures is worth investigation as well. Broadly speaking, when taking multiple data sources into consideration, its effects fall within the following three cases: (a) zero effect, where the individual data source is independent; (b) negative effect, where integrating multiple data sources will lead to poorer performance than considering the data sources separately; and (c) positive effect, where integrating multiple data sources will lead to better performance (additive effect or even multiplier effect) than considering each individual data source separately.

As a summary, behavior interior dimensions provide a domain knowledge rooted way to transform and integrate multiple data sources. Even though at the risk of information loss, the advantages of this approach are prominent, i.e., more concise, intuitive, and easy to understand.

2.2. Roots of Behavior Interior Dimensions and Its State of the Art in UGC-Based Research. The analytical or logical behaviorism theory in philosophy aptly defines “interior dimensions” as follows: “when we attribute a belief, for example, to someone, we are not saying that he or she is in a particular internal state or condition. Instead, we are characterizing the person in terms of what he or she might do in particular situations or environmental interactions” [20, 21]. Understanding the interior aspect of behaviors is a pivotal issue in various fields, e.g., behavioral biology [1], psychology [2], and marketing management [3]. Apart from being the key focus in traditional domains such as the abovementioned psychology and marketing, we note that behavior interior dimensions are also investigated in user-generated content (e.g., Twitter, Facebook) based analysis in political election, stock market trending, and so on.

Researchers are beginning to do an in-depth study in this largely uncharted territory of the analytics. It has received wide attention in box office revenue prediction, stock market trending, political elections [5], and opinion tracking in environmental affairs [6]. Bai et al. [22] predicted the big-five personality based on user behaviors at social network sites. Romero et al. proposed an IP (Influence-Passivity) model based on PageRank [16], assigning a relative influence and a passivity score to every users based on the ratio at which they forward information. In stock analysis, Google-Profile of Mood States measures mood in terms of six dimensions:

calm, alert, sure, vital, kind, and happy. Another piece of work in stock microblogs [8] studies how to predict the stock market features (e.g., returns, trading volume, and volatility) based on bullishness and the level of agreement between postings and message volume.

There are two key observations. First, we can see that different from the statistics on external dimensions provided in most social media analytics systems, a number of interior dimensions have already been incorporated in these studies. Second, even though interior dimensions are addressed in these studies, the focus is either unidimensional without considering interdependence or static without considering temporal dependence. There is much less research effort to go one step further, to integrate multiple behavior dimensions together to form a more comprehensive view for some phenomenon and for subsequent prediction processes. For example, in some studies of sentiment-based electoral result prediction, sentiments were proved to have a positive correlation with telephone poll results in consumer confidence and presidential job approvals [23]. In some other work [24], they were applied to other electoral data set, but without success. This might indicate that single dimension-based sentiment alone might not be sufficiently robust. Moreover, in Sprenger et al.'s work [8], even though multiple dimensions (i.e., bullishness, message volume, and disagreement in stock microblogs) were analyzed, research on the interdependence among these dimensions is still missing. In Guerini et al.'s work [14], the interdependence between sentiment and controversy and raising discussion was analyzed. However, the analysis is static and lacking an evolutionary view. In our work, we utilize multivariate time series (MTS) analysis techniques [25, 26] which are widely adopted in areas such as sensor recordings in aerospace systems, medical monitoring, and financial systems [27]. MTS techniques are originally expanded from univariate time series analysis, e.g., DFT (discrete Fourier transformation), and later extended to consider the interaction among multiple time series variables, e.g., PCA (principal component analysis). We also adopted the following analytics methods in our study (see Section 4): (a) empirical mean, (b) DFT (discrete Fourier transformation), (c) DWT (discrete wavelet transformation) [28], and (d) PCA (principal component analysis) [26].

3. Capture Correlated Behavior Interior Dimensions in Social Media

To figure out behavior interior dimensions, we apply both “top-down” and “bottom-up” approaches from multiple literatures. On one hand, it needs to be rooted from domain knowledge. On the other hand, these dimensions have to be automatically measured or approximated. The upper half of Figure 1 summarizes our surveyed results of user-oriented behavior interior dimensions from sociology and psychology [3]. Subdimensions refer to the constituent elements found in the literatures on social network analysis. There is a certain degree of overlapping in the concept for primary category, e.g., extraversion includes positive affect and energy level which is also the activeness in primary category. While there lack precise and universally agreed term definitions at the

first level, there is often consensus at the sublevels, with more quantitative definitions that can be automatically measured or approximated from social data. For example, the dominance out from extraversion can be approximated with affect dominance and textual dominance by using linguistic tools ANEW and LIWC [29].

There are still quite a few dimensions such as motivational dimensions that are difficult to measure from user exterior behavioral data, e.g., whether a topic content bears a certain entertainment value (surprising/awe inspiring) so that it will reflect positively on the people who transmit it. Figure 1 depicts the decision-making process in coming up with a set of behavior interior dimensions to describe the analytic object in a specific domain. It includes the following six steps.

The first step is to determine exterior dimensions through literature review in the given domain. Once determined, the second step is to come up with a draft set of behavior interior dimensions based on the similarities and differences in the concept of these determined behavior exterior dimensions, corresponding to (a) in Figure 1.

Then, the belongingness of each exterior dimension determined in the first step is examined with respect to the draft set of behavior interior dimensions. The fourth step continues to examine its appropriateness: if the current behavior exterior dimension can be put under more than one interior dimension or cannot be put under any of the behavior interior dimension, then the current set of behavior interior dimensions is not very appropriate, and a modification is required. This can be done in two ways: first, if the current exterior dimension can be put under more than one interior dimension, conduct a resegmentation of the behavior exterior dimensions from a different perspective based on the concept similarities to each other (corresponds to (a) in Figure 1); otherwise, if the current behavior exterior dimension cannot be put under any of the behavior interior dimension, add a new behavior interior dimension (corresponds to (b) in Figure 1). Note that, if the categorization involves a hierarchy, the assignment should be the lowest category. The process shown is repeated until all the behavior exterior dimensions identified in the first step have been classified.

The fifth step examines the similarities in the identified behavior interior dimensions to ensure that a proper classification is obtained with as much similarity in the behavior exterior dimensions classified under each behavior interior dimension as possible and as much difference in the behavior exterior dimensions across different behavior interior dimensions as possible. The following two ways can be done to achieve this aim: the first way is to resegment the behavior exterior dimensions based on its concept relatedness to the identified interior dimensions (corresponds to (a) in Figure 1); the second way is to examine whether there exists a hierarchy in the identified behavior interior dimensions, remove the redundant part, and reduce the hierarchy to the lower level (corresponds to (c) in Figure 1).

Continuing through the decision-making process, once the behavior interior dimensions are determined, the sixth step is to examine the measurability through automatically processing the raw big data. Then, it leads us to the final set

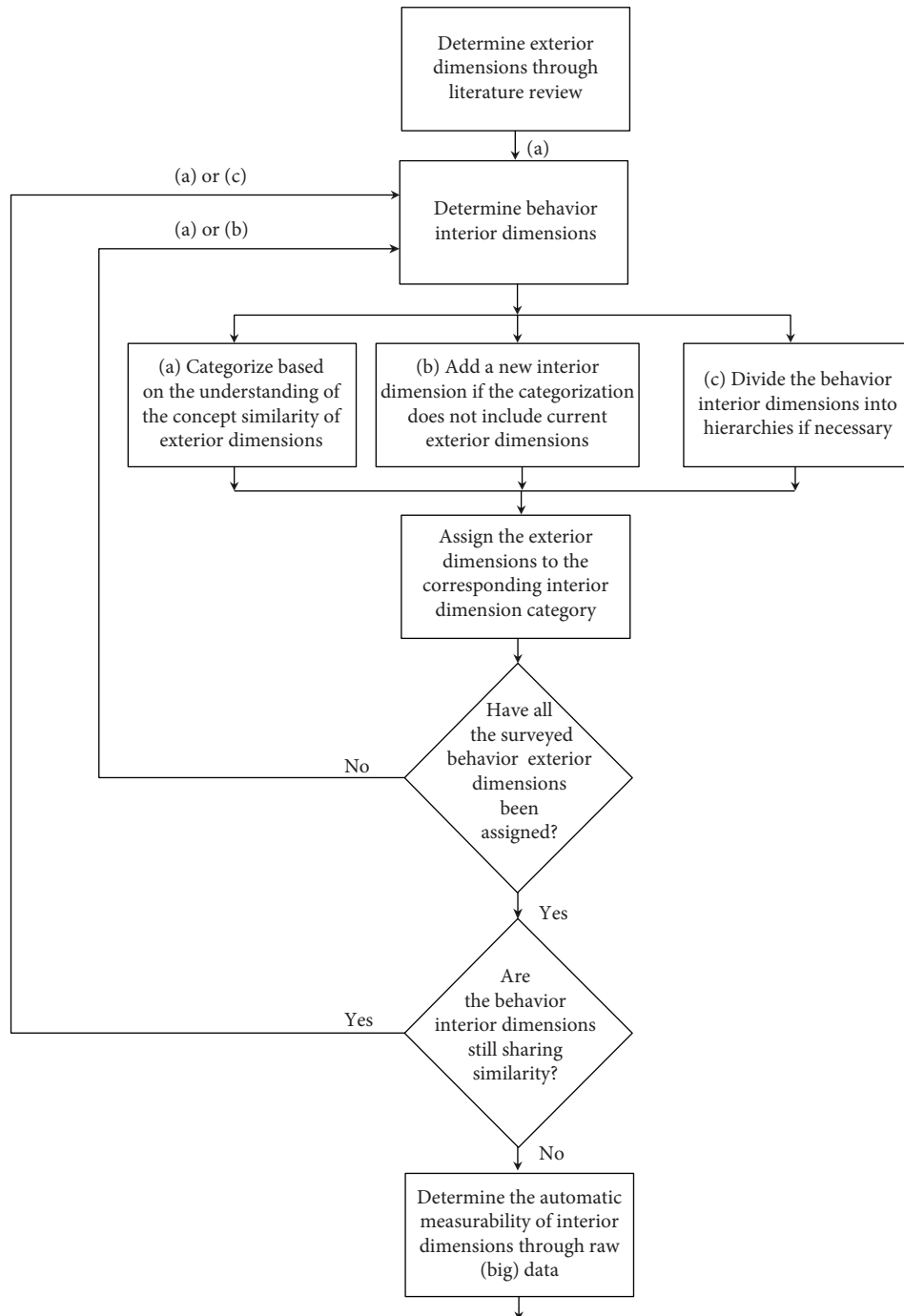


FIGURE 1: Identification of behavior interior dimensions.

of behavior interior dimensions under study in the given domain. The measurements usually include the following:

- (i) Assign. Most measurements fall within category shall refer to external database (e.g., linguistic databases) and assign the text-based values to corresponding dimensions, as in the case of affect dimension (see Tables 1 and 2). The widely adopted linguistic-based tool is LIWC (Linguistic Inquiry and Word Count) [29] and ANEW (Affective Norms of English Words) [30]
- (ii) Aggregate. The measures of a behavior interior dimension within this category are based on the aggregates of its subdimensions. Here, by “aggregate,” we mean that the operations are no more complex than algebraic operations. For example, in the Twitter context under study in this thesis, user disturbance is the average sum of LIWC “negative emotion,” “anxiety,” and “sadness”; topic controversy is the average difference of all the two consecutively posted tweets (see Tables 1 and 2), topic content richness is the average sum of content volume and

TABLE 1: User-oriented behavior interior dimensions.

	Dimensions	Related concepts in Table 3	Measurement in our study
Self-oriented	Activeness	Activeness, extraversion (social vitality)	Tweet amount
	Sentiment	Affect (valence), extraversion (social vitality)	Linguistic approach based on LIWC- “positive affect”
	Disturbance	Neuroticism, negative affect, sadness, anger, anxiety, etc.	Linguistic approach based on LIWC- “negative emotion,” “anxiety,” “sadness”
	Openness	Preference for novelty and variety	Hashtag adoption latency and hashtag usage variety
Peer-oriented	Popularity	Popularity	In-degree (follower count)
	Gregariousness	Gregariousness, extraversion (social vitality)	Out-degree (followee count)
	Reciprocity	Agreeableness, altruism	Friend count
	Influence	Influence	Retweet count, mention count
	Passivity	Passivity	Hashtag unadoption percentage
	Dominance	Extraversion (social dominance)	Linguistic approach based on ANEW- “dominance”
	Textual sociability	Extraversion (social vitality)	Linguistic approach based on LIWC-“social process”

content diversity, and topic hotness in Twitter is the average sum of communication count and coverage of people (see Table 2). Note that most of the measures of behavior exterior dimensions, especially the behavioral statistics, fall within this category

- (iii) Transformation. If there does not exist a developed measure from literature for a given behavior interior dimension, a new measure should be developed. The measures of content volume and content diversity fall within this category (see Tables 1 and 2)

Of these three measures, both “aggregate” and “transformation” are N -to-1 mappings between exterior dimensions and interior dimensions, while “assign” is 1-to-1 mapping.

Table 3 summarizes our surveyed results of user-oriented behavior interior dimensions from sociology, psychology, and so on. Subdimensions refer to the constituent elements found in the literature survey for the social network analysis. In this table, we note that, firstly, there is a certain degree of overlapping in the concept for primary dimensions, e.g., extraversion includes positive affect and energy level (activeness). Secondly, while there lack precise and universally agreed definition terms at the first level, there is often consensus at the subdimension levels, with more quantitative definitions that can be automatically measured or approximated from the monitored social data. The corresponding measurement is given in the “related measurement in literature” column. For example, dominance in extraversion can be approximated with affect dominance and textual dominance using the linguistic tools ANEW and LIWC [29].

Therefore, we focus on subdimensions and select the final set of user-oriented dimensions used in our study by filtering based on whether (a) they can be measured practically and (b) they are not redundant in concept. This leads to the dimensions shown in Table 1. Moreover, while Table 3 presents a traditional view from psychology and sociology, Table 1 reorganizes the dimensions from the analytic/measurement point view. That is, these

subdimensions are classified into two classes: self-oriented or peer-oriented in accordance with intrapersonal and interpersonal interdependence (as discussed in Related Work), respectively. This classification serves as a rough criterion for data preprocessing in measuring each dimension from multiple data sources, as it reflects the data coverage involved, i.e., the data sources that describe the user’s own behaviors or his peer’s behaviors as well. As for the scalability of the measurement, “activeness,” “sentiment,” “disturbance,” “dominance,” “openness,” “influence,” “passivity,” and “textual sociability” are in linear relation to the total number of tweets collected and “popularity,” “gregariousness,” and “reciprocity” are in linear relation to the number of edges in the network (i.e., follower/followee relationship).

Different from the user-oriented case which usually involves hierarchy in the concept in the related domain knowledge, the case is relatively simpler for topic interior behavior dimensions. The selection is shown in Table 2. Of these five topic dimensions, except for content richness which is a polynomial function as it compares each consecutive pair of tweets, the other four dimensions are all linear functions and the time cost of calculating sentiment and controversy is scalable to the total number of words in the tweets collected; for hotness and trend momentum, the time cost is scalable to the total number of tweets collected.

4. Behavior-Interior-Aware Preference Prediction

In this section, we will first briefly go through revisit the typical collaborative filtering (i.e., CF for short) models in Section 4.1, while introducing useful extensions by incorporating those multiple behavior interior dimensions (as given in Section 3). Both the interdependence between multiple behavior interior dimensions and temporal relations are considered in learning user preference from their neighbors

TABLE 2: Topic-oriented Behavior Interior Dimensions.

Dimensions	Subdimensions	Definition in our study	Measurement in our study
Motivation	Practical value [46, 47]	Degree to which the content provides useful information	Lack of information
	Entertaining value	Degree to which the content is interesting or surprising or awe-inspiring	
Affect	Sentiment [48]	Valence- positive or negative	$\text{Sent}(t) = \sum \text{Sent}(p_j)/n_t$
	Controversy [14]	Sentiment variation approximated opinions pro or against	$\text{Contro}(t) = \sum \text{Sent}(p_j) - \text{Sent}(p_{j-1}) /(n_{t-1})$
Content richness	Content volume [49]	Word count in a topic	$\text{CR}(t) = \alpha \text{norm}(\text{CV}(t)) + \beta \text{CD}(t)$, $\text{CV}(t)$ is word count at time t ,
	Content diversity	Nonredundancy of topic words	$\text{CD}(t)^a = \text{avgld}(p_{j1}, p_{j2})/\max(\text{len}(p_{j1}), \text{len}(p_{j2}))$, $\text{ld}(p_{j1}, p_{j2})$ is the Levenshtein distance between two tweets
Virality	Hotness [14]	Transmission times and reach of impact	$H(t) = \alpha \text{CC}(t) + \beta \text{Cov}P(t)$, $\text{CC}(t)$ and $\text{Cov}P(t)$ are communication count and coverage of people of a topic at time t
	Trend momentum [14]	Absolute hotness difference in successive time windows	$\text{TM}(t, i) = \alpha \cdot \text{TM}_{\text{CC}}(t, i) + \beta \cdot \text{TM}_{\text{Cov}P}(t, i)$, $\text{TM} * (t, i) = *(t, i) * (t, i - 1) $, $*$ = $\text{CCorCov}P$

^a $\text{CD}(t) = 0$: all the tweets are the same; $\text{CD}(t) = 1$: none of the tweets have any content that occurred in other tweets.

TABLE 3: Surveyed candidates for user-oriented behavior interior dimensions.

Dimensions		Definition	Subdimensions	Related measurement in literature
Motivation		The reasons that stimulate desire and energy in behaving in a particular way [50]	Intrinsic (e.g., an interest in the task)	Survey
			Extrinsic (e.g., a desire for reward)	Survey
Activeness		The state of being continually engaged in a particular behavior [3, 51]	—	Postcount
Affect		Observable manifestations of a subjective experienced emotion [48]	Dimensional approach: valence, arousal, dominance	ANEW, LIWC
			Categorical approach: happiness, anger, anxiety, etc.	POMC
Personality	Neuroticism	Degree of emotional stability, impulse control, and anxiety [19, 22]	—	BFI [19]
	Openness	A strong intellectual curiosity and a preference for novelty and variety [19]	Preference for novelty	Adoption variety [52]
			Preference for variety	Adoption variety [52]
	Conscientiousness	Being thorough, careful, or vigilant [19, 22, 53]	Reliability	BFI, JPI
			Responsibility	
			Achievement striving	
Extraversion	A higher degree of sociability, assertiveness, and talkativeness [54, 55]	Self-discipline	BFI, NEO PI	
		Order		
Agreeableness	Being helpful, cooperative, and sympathetic towards others [19, 22]	Social dominance	Reciprocity, survey: BFI	
		Social vitality		
		Altruism		
Social status	Influence	Tendency to influence others with prominence in certain aspect [56]	Name value	Mention count
			Content value	
	Popularity	Tendency to be widely accepted [56]	—	In-degree centrality
	Passivity	Difficulty to get influenced [16]	—	Unadopted hashtag percentage
Gregariousness	Tendency to enjoy being around others [57]	—	Out-degree centrality	

(i.e., with high similarity in behavior interior dimensions) to make future predictions.

Then in Section 4.2, to learn user preference, we expand the neighbor sets by considering the users that impact information propagation. We give a coupling mechanism that integrates the linear threshold model and neighborhood CF models in this paper.

4.1. Enhanced Collaborative Filtering Model Based on Behavior Interior Dimensions. Collaborative filtering was first introduced in the context of document recommendation in a newsgroup [31]. Since then, it is widely adopted in e-commerce. There are the two main CF models: neighborhood model and latent factor model. Here, we focus

on neighborhood models as it captures homophily through the choices of similar users; latent factors instead explore the explainability of users' choice through user/items' characteristics/dimensions.

Traditionally neighborhood models capture homophily through exterior rating/adoption times, see (1) and (2). In this sense, our method extends the neighborhood-based model by measuring the similarity s_{uv} between user u and neighbor v with multiple interior dimensions, see (3) and (4).

4.1.1. Neighborhood-Based Models. There are two types [32]: user-based and item-based. Equation (1) shows the case for user-based model. The recommendation is based on the

ratings/adoptions by similar users or given to similar items, after removing global effect and habitual rating.

$$\hat{r}_{ui} = \mu + b_u + b_i + \frac{\sum_{v \in S_{ui}^k} s_{uv}(r_{vi} - b_v - b_i)}{\sum_{v \in S_{ui}^k} s_{uv}}, \quad (1)$$

$$s_{uv} = \frac{\sum_{i \in I(r_{ui} - \bar{r}_u)(r_{vi} - \bar{r}_v)} \sqrt{\sum_{i \in I}(r_{ui} - \bar{r}_u)^2} \sqrt{\sum_{i \in I}(r_{vi} - \bar{r}_v)^2}}, \quad (2)$$

where \hat{r}_{ui} is the recommendation for user u of a certain item/hashtag i , μ is the global average, b_u and b_i denote the user- or item- specific habitual rating difference from μ , s_{uv} measures the similarity between user u and u 's neighbor v , and S_{ui}^k denotes the set of u 's k -nearest neighbors. The user- and item- based neighborhood models are dubbed as “Ngbr_{Corr}^U” and “Ngbr_{Corr}^T”, respectively. These will serve as the base model for behavior interior dimension-based improvements.

4.1.2. Enhanced Neighborhood Models. The similarity between two users (topics) in (1) is computed based on behavior interior dimensions from both static and dynamic perspectives. Static analysis measures the similarity with the Frobenius form of the difference in the empirical mean amplitude of the user interior dimensions (see (3)). For clarity's sake, this model is dubbed as “Ngbr_{EpM}^U”.

$$s_{uv} = \|\bar{d}_u - \bar{d}_v\|_F, \quad (3)$$

where $\bar{d}_u = \langle \bar{d}_{uk} \rangle$ is the empirical mean amplitude of the user interior dimensions and k is the dimension number.

Then, three dynamic patterns are extracted [33]; we dub these three user-oriented models as “Ngbr_{DFT}^U”, “Ngbr_{DWT}^U” and “Ngbr_{PCA}^U”:

- (i) The first one is DFT- (discrete Fourier transform-) based global shape feature $\theta_u^{ac} = \theta_{ukl}^{ac}$, where l indexes the largest nonzero frequency coefficients and is set to 4 as the subsequent coefficients of most topics are zero
- (ii) The second one is DWT- (discrete wavelet transform-) based local shape $\theta_u^s = \theta_{ukl}^s$, l being set to 7 (i.e., the 2nd-8th DWT coefficients, the 1st one is average amplitude), considering the 41-week coverage
- (iii) The third one is PCA- (principal component analysis-) based cooccurrence pattern, i.e., eigenvector

While the similarity between user u and v is also calculated based on the Frobenius form (similar to the previous two dynamic patterns), cooccurrence pattern based on the Eros (extended Frobenius norm) is given as follows:

$$s_{uv} = \sum_{k=1}^{n_d} w_k |\langle \mathbf{o}_k^u, \mathbf{o}_k^v \rangle| = \sum_{k=1}^{n_d} w_k |\cos \theta_k|, \quad (4)$$

where \mathbf{o} is the eigenvector of the covariance matrix for the multiple behavior interior dimensions and w is a weight vector based on the eigenvalues.

Similarly, for topic-oriented enhanced neighborhood models, we have “Ngbr_{EpM}^T”, “Ngbr_{DFT}^T”, “Ngbr_{DWT}^T”, and “Ngbr_{PCA}^T”.

4.2. Integrated Model with Preference Propagation

4.2.1. Multiple-Thread Linear Threshold Model. A typical model for impact propagation is the linear threshold model [34], see (5). In this equation, the probability of a given user to turn active is a function $p(a)$ of the number a of friends being active. The optimization goal is to maximize (6). We note that the challenges of applying this method in top- N hashtag adoption frequency prediction setting lie in the following: (a) there is a shift of focus from single item to multiple items and (b) the traditional optimization approach may produce very low prediction accuracy due to the fact that social media is a noisy and asynchronous environment for user interaction, if we take all the nonadoption event into consideration. Therefore, we come up with the following model. We dub this model as “MTLT”, see (7). As discussed above, the model is trained for each topic/hashtag. The training process aims at maximizing the likelihood of hashtag adoption prediction at each t , and t is set to weekly in our study.

$$p(a) = \frac{e^{\alpha \ln(a+1)+\beta}}{1 + e^{\alpha \ln(a+1)+\beta}}, \quad (5)$$

$$\prod_a p(a)^{Y_a} (1 - p(a))^{N_a}, \quad (6)$$

where a is friend count; α measures the impact propagation, a large value of α indicates a large degree; β is to reduce the possibility of overfitting; and Y_a and N_a denote the adoption event count and nonadoption event count.

$$\prod_i \prod_t p(u, i, t)^{Y_{i,t}}, \quad (7)$$

where $p(u, i, t)$ is the probability of user u adopting topic i at time t $tp(u, i, t) = p(\text{Stat}(u, i, t) = \text{active} | \text{Numf}(u, t - 1) = a, \text{stat}(u, i, t - 1) = \text{dormant/active}) = p(a)$; here, we assume that once a user is active, the next stage probability is proportional to the number of active friends. $Y_{i,t} = \sum_u I_{u,i,t}$; $I_{u,i,t}$ is an indicator variable denoting that user u adopts topic i at time t . The parameters are trained by moving toward the direction of the gradient.

4.2.2. Integrated Model. The collaborative filtering model predicts future user hashtag adoption times while the threshold model predicts the probability of adopting the hashtag. Note that the range of these two models is different, i.e., $\hat{r}_{ui} \in [0, \max_u \{\text{number of hashtags adopted by user } u\}]$ and $p(u, i, t | a, t - 1) \in [0, 1]$; therefore, a normalization phase is needed to integrate these two models (see (8)).

$$p(u, i) = \max \{ \gamma_1 \cdot \text{norm}(\hat{r}_{ui}) + \gamma_2 \cdot p(u, i, t | a, t - 1) \}, \quad (8)$$

where $\gamma_1, \gamma_2 \in [0, 1]$ and $\gamma_1 + \gamma_2 = 1$. Note that $\gamma_1 = 1$ implies the CF-based model and $\gamma_1 = 0$ implies the propagation model.

5. Empirical Study

In Section 5.1, we will first introduce the empirical data set used to evaluate the above-described methods and then describe the evaluation metrics to evaluate the prediction accuracy and baseline used methods for comparison. The results are reported in detail in Section 5.2.

5.1. Experimental Design

5.1.1. Empirical Data Set. We use Twitter data from 2010 01 to 2010 10, with the total size of 70 Gbytes. The behavior interior dimensions are extracted for each user and topic on a weekly basis, i.e., 41 full weeks from the 2nd~42nd week. We adopted a similar procedure as the one in [35], which is a variant of the leave-one-out holdout method. The adoption frequency prediction is evaluated on a 5-core data set S in which every user has adopted at least 5 hashtags and every hashtag has been adopted at least by 5 people. The 5-core data set S is then splitted into two sets: a training set S_{train} and a testing set S_{test} . Denote the splitting time point as t_{split} and consider we have about 10-month data set (2nd~42nd weeks of 2010); t_{split} is set at the last month, i.e. 38th week. In total, we have $|U_{\text{train}}| = 22849$ and $|T_{\text{train}}| = 32727$.

Different from the standard recommendation data set, such as MovieLens data set (<https://grouplens.org/datasets/movielens/>), where the ratings are made on a 5-star scale, with half-star increments, or KDD Cup 2011 Yahoo music recommendation data set (<http://jmlr.org/proceedings/papers/v18/>), with rating range between 1 and 5 (integral), the hashtag adoption times ranges [1, 2838] with a highly skewed distribution towards 1. Note that the case that hashtag is adopted only once takes up 71.01%. The difference between the estimated and actual adoption times fed back in parameter estimation with stochastic gradient descent that could be as large as about 2000. Furthermore, considering the highly skewed distribution, we adopted a nonlinear normalization (see (9)).

$$n_h^{\text{norm}} = 10 * (1 - 0.9^{n_h}), \quad (9)$$

where n_h and n_h^{norm} denote the actual and the normalized hashtag adoption times, respectively.

5.1.2. Evaluation Metric and Method. The prediction accuracy is measured by recall rate/hit rate of the top- N adoption frequency prediction results. A hit is deemed as occurred if the N hashtags generated for user u contain u 's most probably adopted hashtag (a.k.a. hidden hashtag/withheld hashtag) [32]. The most probably adopted hashtag is with the highest frequency. A confounding factor, 1000 random hashtags, is added for each true adoption.

The proposed methods are evaluated against two competing models that are developed based on heuristics: hashtag average adoption times and top popularity (the number of

people adopted the hashtag) [36]. The former approach recommends top- N items with the highest average adoption times. The latter adopts a similar prediction schema, recommending top- N items with the highest popularity (i.e., the greatest number of users that adopted this hashtag).

5.2. Results and Analysis

5.2.1. Prediction Accuracy. Figure 2 summarizes the recall rate of the methods proposed in Section 4. The models are trained with a learning rate 0.007, $\lambda = 0.002$. Our proposed models are marked with *. The two largest recall scores are highlighted in bold for each group. We have the following findings from Table 1.

First, we see that capturing homophily through behavior interior dimensions has better performance (i.e., the recall rate for the top 20 recommendation is 36.5% and 37.3% for $\text{Ngbr}_{\text{EPM}}^U$ and $\text{Ngbr}_{\text{EPM}}^T$) than those based purely on usage statistics (27.4% and 25% for $\text{Ngbr}_{\text{Corr}}^U$ and $\text{Ngbr}_{\text{Corr}}^T$). This supports our assumption that interior dimensions capture latent similarity between users and topics in addition to the extrinsic user-hashtag adoption frequency. Second, we observe that coupling impact propagation and similarity in user preference leads to a higher recall rate, with $\text{Intgr}_{\text{EPM}}^T$ the highest: 45.2%. The recall rate of the two coupling components, Ngbr_E^T and MTLT, is 37.3% and 33%, respectively. Hence, the complementary properties of these two factors are (a) social impact-driven propagation through followers' posts or other people's posts in the same topic and (b) similarity in interests, where hashtag prediction is supported.

5.2.2. Static vs. Dynamic. The results are summarized in Figure 2. We observe that for topic-oriented behavior interior dimensions (see left figure in Figure 2), DWT-based local shape has the best prediction accuracy, followed by the PCA-based pattern. For user-oriented behavior interior dimensions (see right figure in Figure 2), static and dynamic cases are similar in the recall rate-based prediction accuracy. The prediction accuracy curves for both types of models are in convex shape: it increases very fast for small N and then starts to level off. The turning point occurs at about 5 for user-oriented models and 10 for topic-oriented models. It indicates that the collaborative filtering models perform equally badly for small N ($N = 1$), i.e., top-1 recommendation. Thus, the collaborative filtering models perform fairly well for recommending a set of hashtags that people are most likely interested in, not a precise prediction of the exact hashtag a user may adopt.

Furthermore, the gap difference between user and topic-oriented enhanced models (e.g., $\text{Ngbr}_{\text{EPM}}^{U/T}$) and their corresponding baseline model (i.e., $\text{Ngbr}_{\text{Corr}}^{U/T}$) indicates that user and topic-oriented enhanced models have their own "best bet" range. More specifically, the gaps are large for user-oriented models but almost zero for topic-oriented models at a small range of N , whereas at a large range of N , the gaps for topic-oriented models are much larger than those of user-oriented models. Therefore, in utilizing interior dimensions for hashtag recommendation, it is better to use user-

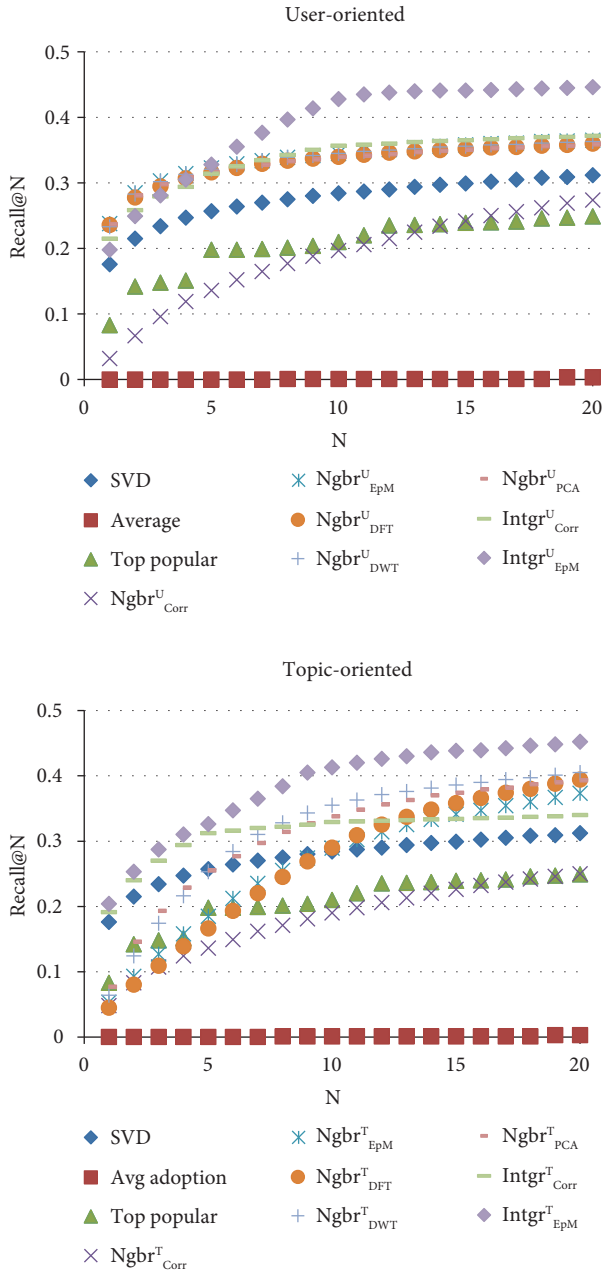


FIGURE 2: Prediction accuracy.

oriented models for small N recommendation and use topic-oriented models for large N recommendation.

5.2.3. Impact of Hashtag Popularity on Prediction Period Sensitivity. Several recent works show that hashtag popularity will affect prediction accuracy, i.e., the chance of popular hashtags got adopted is significantly higher than that of unpopular hashtags. This is due to the fact that “the inherent social component of the collaborative filtering approach makes it biased towards popularity” [36, 37]. However, its effect on the prediction period sensitivity is still unknown.

To do so, we conduct a repetitive experiment and take the mean accuracy for each prediction period by keeping the first two months for model fitting and use the following 1st to the

8th month for model evaluation. The test sets are divided into short-head (popular hashtag) test sets and long-tail (not popular hashtag) test sets in a similar way to [36]. In our data set, top 33% of hashtag adoptions involve only 1.45% of the most popular hashtags (493 short-head hashtags). Figure 3 presents the skewed distribution for hashtags with respect to their popularity shown with these 493 hashtags. Actually, it is even more long-tailed than that of the two common recommendation data sets: Movielens and Netflix [36], of which the top 33% ratings involve 1.7% and 5.5% items, respectively. The remaining 98.5% hashtags comprise the long-tail test sets.

Results in Figure 4 show that there is a significant difference in hashtag popularity on prediction period sensitivity. The recall rate-based prediction accuracy for popular (short-head) topics shows no definitive trend as the prediction period increases. The recall rate-based prediction accuracy for less popular (long-tail) topics decreases with respect to the prediction period.

6. Behavior Interior Implications

In this section, we will first explicate the improvement of interior dimension-based homophily models by zooming into the similarities and differences of the neighborhoods selected by these two approaches and develop an overlap based on Jaccard index [38]. Besides measuring homophily based on the behavior interior dimensions, in Section 6.2, we studied the explainability of interior dimensions, i.e., whether some interior dimensions are more likely to induce the user hashtag adoption behavior, through comparing traditional latent factor models [39] with explicitly modeling the “latent factor space” with behavior interior dimensions.

6.1. Exterior vs. Interior in User-/Topic-Neighbor Selection. The results in the previous section show that interior dimension-based collaborative filtering models can lead to better prediction accuracy than exterior usage-based models. The difference between exterior statistics-based CF models and interior dimension-enhanced CF model lies in the homophilous neighborhood for the prediction model to learn users’ preferences. Take the user-oriented models as an example, when predicting the preference of user u on item i , u ’s neighbors of the exterior usage-based model ($\text{Nbr}_{\text{Corr}}^U$) is limited to the k users most similar to user u that have all used item i , as denoted in $Sk(u; i)$ in (1). However, those users that have not used item i can also have similar preferences as u . It could happen that the sets of items by two users sharing similar interests are intersected for only a small part or even nonoverlapping at all, due to the multitude of items (hashtags) existing in Twitter. Thus, these user item usages can also serve as a meaningful source for the model to learn.

To compare the interior dimension-based and exterior hashtag usage frequency-based homophilous neighbors, we resort to Jaccard index [38], a statistic used for the similarity and diversity comparison of two finite sets, measured by the size of the intersection over the size of the union of the two sets. Let JI_u denote the difference between interior and

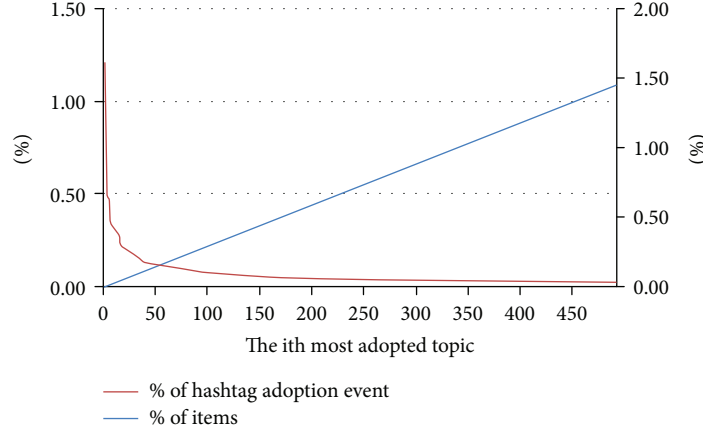


FIGURE 3: Short head and long tail of hashtag popularity distribution for top 33% hashtag adoptions in Twitter.

exterior dimension-based neighborhood selection, then we have JI_u equal to the max Jaccard index between S_u^k -user u 's neighbor sets determined through $\text{Ngbr}_{\text{EpM}}^U$ and $S_{u,i}^k$ -user u 's neighbor sets determined through $\text{Ngbr}_{\text{Corr}}^U$ over all topic i that user u has posts under (see (10)). Note that while the neighbors in $\text{Ngbr}_{\text{Corr}}^U$ differ with regard to different i , i.e., user-topic pair, they remain the same in $\text{Ngbr}_{\text{EpM}}^U$ for a given user with regard to all topics. The reason is that interior dimensions, like genomes, are more stable compared with exterior behavior manifestations.

$$JI_u = \max_i \left\{ J(S_u^k, S_{u,i}^k) = \frac{|S_u^k \cap S_{u,i}^k|}{|S_u^k \cup S_{u,i}^k|} \right\}, \quad (10)$$

where S_u^k and $S_{u,i}^k$ denote user u 's k neighbors in $\text{Ngbr}_{\text{EpM}}^U$ and $\text{Ngbr}_{\text{Corr}}^U$, respectively.

We are particularly interested in how the neighborhood difference through the interior and exterior dimension-based neighborhood selection methods varies along the population distribution for each dimension. The greater the difference is for the top p percentage with a small value of p (< 50), the more effective the interior dimension or exterior dimension is in capturing homophily. Equation (11) gives the Jaccard index of the interior and exterior dimension-based neighborhood sets for the top p percentage of users w.r.t. a specific interior dimension d . Equation (12) gives the average Jaccard index of the union of the top p percentage of users for all dimension d .

$$JI_d(p) = \text{avg} JI_u(u \in U_{d,p}), \quad (11)$$

$$JI_{\text{overall}}(p) = \text{avg} JI_u(u_d U_{d,p}), \quad (12)$$

where d indexes the interior dimensions identified in Table 1 and $U_{d,p}$ denotes the top p percentage of users w.r.t. a specific dimension d .

Similarly, for topic neighbors, the comparison is conducted between S_i^k of $\text{Ngbr}_{\text{EpM}}^U$ and $S_{i,u}^k$ of $\text{Ngbr}_{\text{Corr}}^U$, where

$S_{i,u}^k$ denotes the k items rated by u that are most similar to i and $S_{i,u}^k$ differs w.r.t. different u . The Jaccard index is then analyzed for each of and the union of the 5 topic behavior interior dimensions (see Table 2).

The results are summarized in Figure 5 for user-based and topic-based neighborhood selection, respectively. First, we can see that there exists a significant distinction in interior dimension-based and exterior behavior-based neighborhood selection for both user-based and topic-based cases: the average overlapping percentage in user neighbor selection is only 5.47% (see Figure 5), with the greatest overlapping percentage of 40% in user neighborhood; similarly, the average overlapping percentage in topic neighbor selection is only 0.69% (see Figure 5), with the greatest overlapping percentage of 40% in topic neighborhood.

Second, these interior dimensions are not independent, as the overall overlapping percentage (see the black dashed curve in Figure 5(a)) is smaller than the additive sum of each. That is, there are certain users with a high value in one interior dimension that may also have high value in another dimension. The user set sorted in decreasing order of the strengths in each interior dimension is not exclusive, i.e., $|U_{d,p}| < \sum |U_{d,p}|$.

Moreover, we can observe that generally there is a decreasing trend in the overlap of the interior-based and exterior dimension-based neighborhoods as the strength in each interior dimension decreases. On one hand, this observation is consistent with the finding in the literature about the positive correlation between content virality and activeness, sentiment [40], openness [22], and so on. On the other hand, it indicates that there is a higher probability observing exterior pattern for users/topics that are distinctively high in at least one of the interior dimensions, i.e., the left hand of each curve. More importantly, it suggests that compared with exterior dimension-based method, the power of interior dimension-based method lies in the neighborhood selection for those with low strengths in the interior dimensions (as the right hand of the curve is equal to or even smaller than average, for the latter; see the right hand of the topic hotness

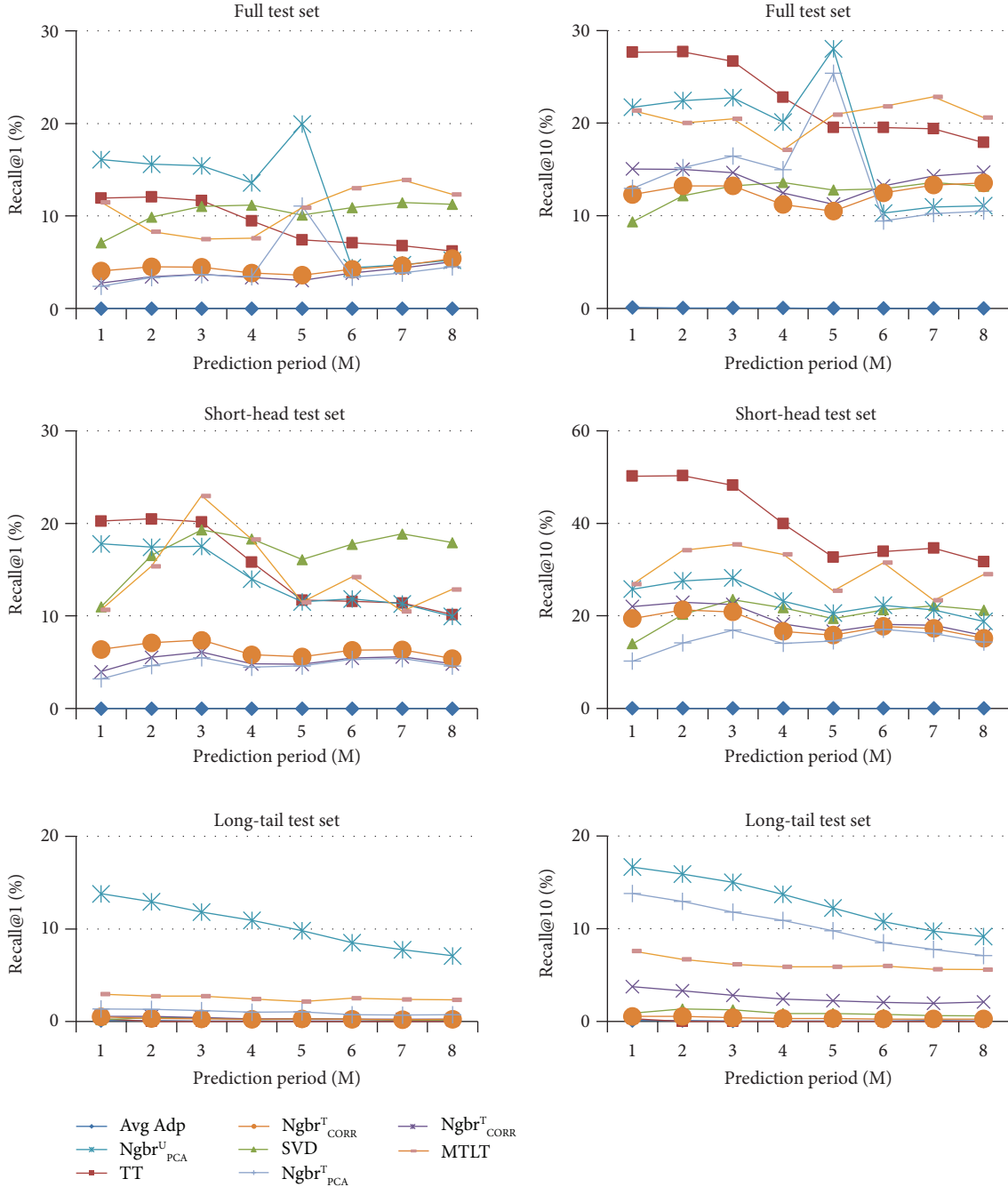


FIGURE 4: Prediction period sensitivity.

and trend momentum curves in Figure 5(b)) and is less likely to observe exterior manifestations.

6.2. Exterior vs. Interior in Explaining User-Topic Preference.

To study the explainability of interior dimensions, we resort to “latent factor models” by explicitly modeling the “latent factor space” with behavior interior dimensions. The “latent factor space” is a hidden layer that tries to characterize the common focus between each user-item pair [39]. Previous approaches such as SVD-like (see (13)) iterative estimation require imputations in order to fill in the unknown matrix entries as it involves estimation

of millions, or even billions, of parameters, and shrinkage of estimated values to account for sampling variability proves crucial to prevent overfitting [41]. Latent factor-based models transform both items and users to the same latent factor space so that they can be compared directly. A typical model associates each user u with a user-factor vector $p_u \in \mathbb{R}_f$ and each item i with an item-factor vector $q_i \in \mathbb{R}_f$. Each factor measures how much the user likes an item (e.g., movie) on the corresponding (movie) factor [39].

Among all the variants of this model, SVD is reported to have one of the best prediction accuracies [36]. This is one of

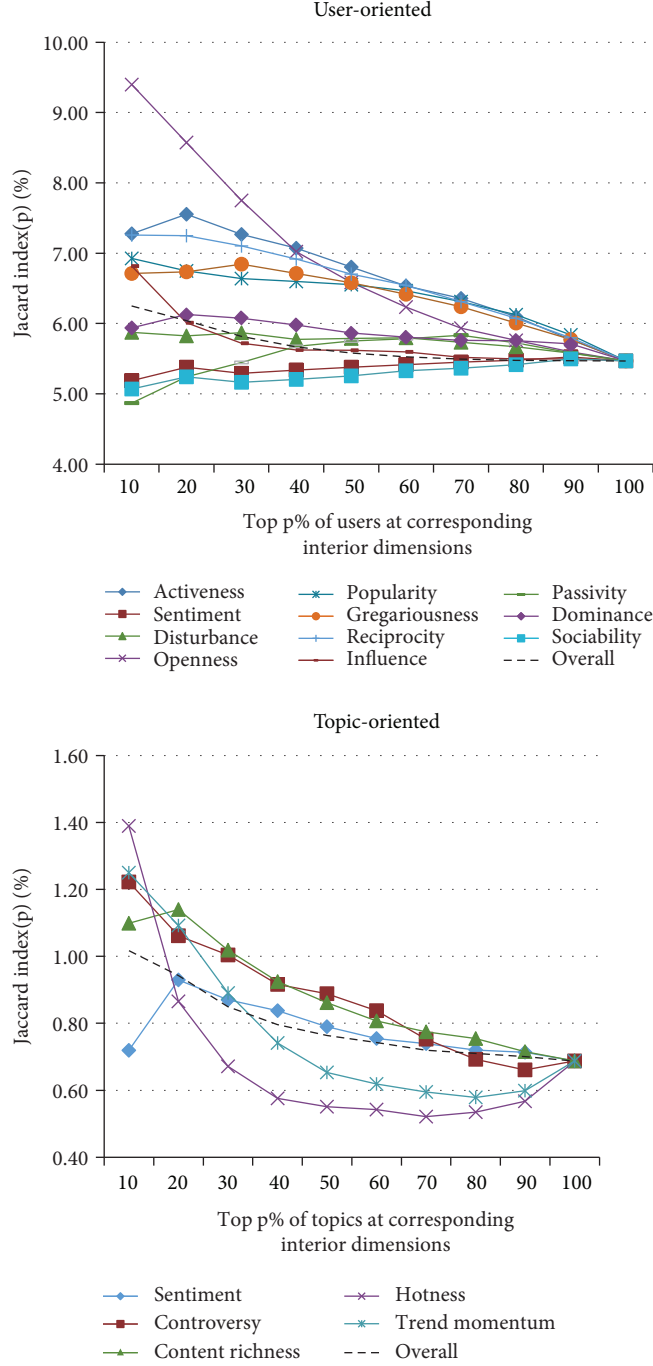


FIGURE 5: Overlapping in neighborhood selection. (a) Overlap between user interior dimension-based ($\text{Nbr}_{\text{EpM}}^U$) and exterior usage-based ($\text{Nbr}_{\text{Corr}}^U$) neighbor selection; (b) overlap between topic interior dimension-based ($\text{Nbr}_{\text{EpM}}^T$) and exterior usage-based ($\text{Nbr}_{\text{Corr}}^T$) neighbor selection).

the baseline models adopted in this paper. The parameters are estimated by using stochastic gradient descent to minimize the squared errors. For a given training case r_{ui} , we modify the parameters by moving the opposite direction of the gradient, yielding (14).

$$\hat{r}_{ui} = \mu + b_u + b_i + q_i^T p_u, \quad (13)$$

$$\min_{q_u, b_u} \sum_{(u,i) \in K} (r_{ui} - \mu - b_u - b_i - q_i^T p_u) + \lambda_2 \left(\|q_i^d\|^2 + b_u^2 + b_i^2 \right), \quad (14)$$

where μ is the global average; b_u and b_i denote the user- or item- specific habitual rating difference from μ ; $p_u \in \mathbb{R}_f$ and models the user-factor and item-factor vector, respectively;

and λ_2 denotes the extent of regularization to avoid overfitting by penalizing the magnitudes of the parameters.

Instead of modelling latent features through p_u and q_i , we model through user interior dimension explicitly with (i) empirical mean, (ii) global shape, (iii) local shape, and (iv) multidimension cooccurrence pattern, as shown in (15). Thus, we have “SVD_{EpM}^U,” “SVD_{DFT}^U,” “SVD_{DWT}^U,” and “SVD_{PCA}^U,” respectively, for user-oriented interior dimensions.

$$\begin{aligned}
 \hat{r}_{ui} &= \mu + b_u + b_i + \sum_k q_{ik} \bar{d}_{uk}, \\
 \hat{r}_{ui} &= \mu + b_u + b_i + \sum_k \sum_l q_{ikl}^{ac} \theta_{ukl}^{ac}, \\
 \hat{r}_{ui} &= \mu + b_u + b_i + \sum_k \sum_l q_{ikl}^s \theta_{ukl}^s, \\
 \hat{r}_{ui} &= \mu + b_u + b_i + \sum_k \sum_l q_{ikl}^u o_{kl}^u,
 \end{aligned} \tag{15}$$

where \bar{d}_{uk} denotes the empirical mean of each dimension (k is the dimension number) for user u , θ_{ukl}^{ac} denotes the DFT- (discrete Fourier transform-) based global shape feature with the largest l nonzero frequency coefficients, θ_{ukl}^s denotes the DWT- (discrete wavelet transform-) based local shape with l DWT coefficients (note that considering the 41-week coverage, here we use the 2nd–8th coefficients, with the 1st one being the average amplitude), and o_{kl}^u denotes PCA- (principal component analysis-) based cooccurrence pattern, with $o_k^u = \langle o_{kl}^u \rangle$ as the eigenvector of dimension k obtained from the covariance matrix for the multiple behavior interior dimensions.

Similarly, for topic-oriented interior dimensions, we have “SVD_{EpM}^T,” “SVD_{DFT}^T,” “SVD_{DWT}^T,” and “SVD_{PCA}^T.” Note that a normalization procedure is required specifically for “SVD_{DFT}^U” and “SVD_{DFT}^T” to make them converge. It is because DFT coefficients are not a constraint to the range $[0, 1]$ as other patterns do, but with the greatest possible value around 40. This is the intrinsic process of Fourier transformation of original time series into a finite combination of complex sinusoids.

While previous results show that user-oriented interior dimensions capture homophily better and lead to better prediction accuracy, topic behavior interior dimensions have better explainability than user behavior interior dimensions (see Figure 6). The accuracy starts to improve at a smaller value of N (around 2) for topic-oriented models, with the highest reaching 43% ($N = 20$), whereas there is a slight improvement for user-oriented models starting around $N = 10$, with recall@20 only 37.1%. Interestingly, we could observe that analyses focusing on topic factor explanations are dominating in the literature.

For example, in movie recommendation, some obvious factors include genre and orientation to children. Some less well-developed dimensions include “depth of character development” or “quirkiness” [39]. A plausible explanation might be that user-oriented dimensions are harder to be precisely captured than topic-oriented dimensions. Robust

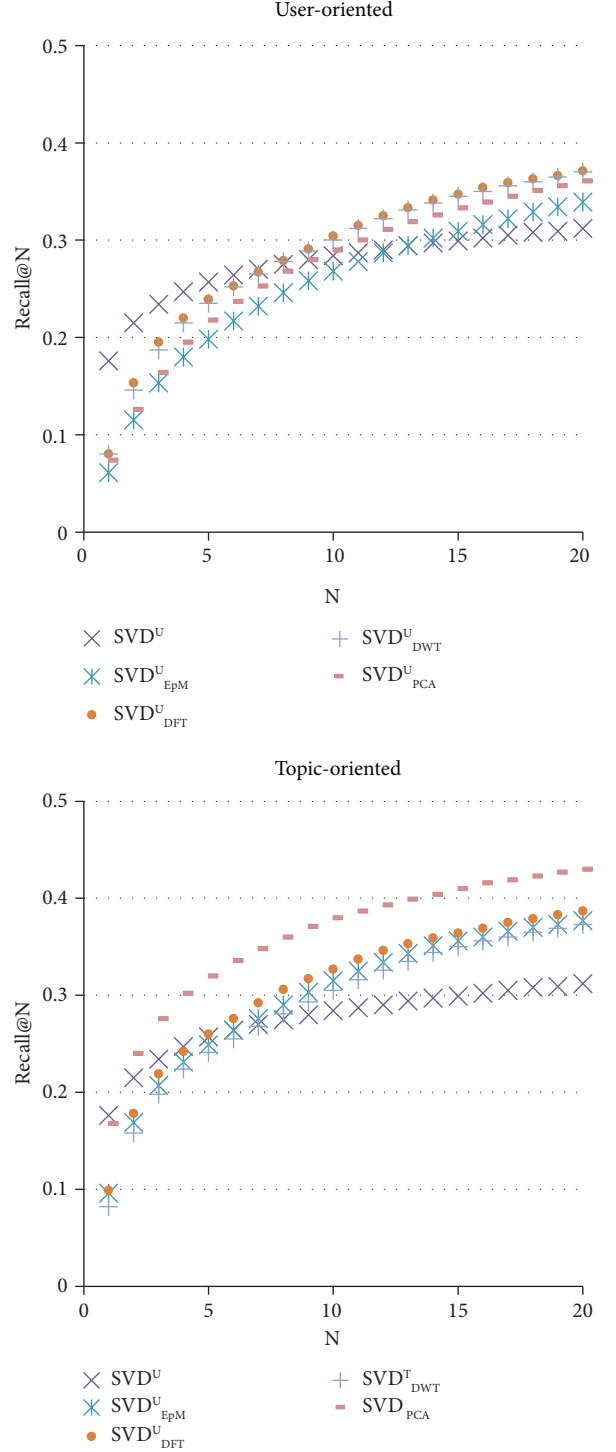


FIGURE 6: Explainability of user/topic behavior interior dimension.

answers to whether and which of these behavior interior dimensions bear a significant explainability require a more direct measurement of user behavior interior dimensions based on traditional psychometric tool, such as NEO PI or Big Five Factor inventory [19].

Based on these results, we have the following insights: first, interior dimension-based similarity in user preference and their impact propagation comprise a more crucial factor

set in the top- N hashtag recommendation than exterior usage-based similarity in user preference and their impact propagation as they provide a better support of the above two conditions. Rather than mixed together like exterior dimension-based similarity in user preference and their impact propagation, interior dimension-based neighborhood user set and the user set that impacts their decisions are almost exclusive. Thus, the linear combination of these two factors in Section 4.2 is reasonable. Besides, it gives us some insights in traditional impact propagation identification study [42, 43]: the confounding phenomenon with homophily might arise from the single exterior adoption behavior manifestation basis; approaching from interior dimension might provide a better segmentation.

7. Conclusion

In this paper, we present an integration model that emphasizes the behavior interior dimensions rather than the exterior transactional statistics in capturing user preference. We test the model on real-world Twitter data, and the results demonstrate that a higher recall rate can be achieved.

Our main contribution is to use the domain knowledge-based behavior interior dimensions to capture as much interdependence among the data as possible. The interdependence between multiple data sources is captured in two levels. Firstly, the interdependence information among raw data sources is captured as behavior interiors in Tables 1 and 2 for users and topics, respectively. Secondly, their interdependence and temporal relations are further considered.

The second contribution is that we offer a Jaccard index-based metric to clearly gauge the difference between the interior dimension-based approach and the exterior dimension-based approach in the neighbor selection by measuring the overall overlapping percentage of the neighbor sets generated through these two methods.

Another contribution is that by incorporating multiple interior dimensions in hashtag recommendation models, the explainability of hashtag recommendation is greatly enhanced. Most often, users are facing “black box” recommendations, such as the latent factor models, where the user-item rating (i.e., user-hashtag adoption times) matrix is factorized to a joint latent factor space of dimensionality (see the above analysis in Section 6), and ratings (i.e., adoption times) are modeled as the inner products in that space. In this sense, the interior dimensions make the prediction more explainable.

As for the future work, we note that in addition to the prediction task that we are dedicated to do in this paper, namely, user-hashtag recommendation, this interior dimension-based approach may be applied to other predictive tasks, such as the diffusion and retweet dynamics prediction. A second direction is to compare the effectiveness of the behavior interior dimension-based methods and those exterior statistics-based methods, e.g., some notable methods are topic feature-related diffusion prediction-based LDA (latent Dirichlet allocation) [44]. As we have mentioned above, the behavior interior dimensions can better capture the subtle differences in users’ characteristics if the data is

heterogeneous and interrelated in nature. When the diffusion pattern is homogeneous and clear-cut, such as retweet, the exterior statistics-based approach may sometimes outperform the interior dimension-based approach. Another direction is to investigate how to integrate the behavior interior dimensions with the time-dependent modeling approach in the predictive tasks to enhance the prediction accuracy. For example, TiDeH (time-dependent Hawkes process) [45] models the number of retweets as a self-exciting point process and acknowledges the differences between users by explicitly taking the behavior characteristics into consideration, even though on an exterior statistic basis. By introducing an intermediate layer of the behavior interior dimensions, it can be expected that the interpretation of the raw data in the dynamic diffusion process is to be greatly enhanced and improved.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

This work was partly supported by Griffith University’s 2018 New Researcher Grant, with Dr. Can Wang being the chief investigator.

References

- [1] M. Mathioudakis and N. Koudas, “TwitterMonitor: trend detection over the twitter stream,” in *Proceedings of the 2010 International Conference on Management of Data - SIGMOD '10*, pp. 1155–1158, Indianapolis, IN, USA, 2010.
- [2] J. Marbach, C. R. Lages, and D. Nunan, “Who are you and what do you value? Investigating the role of personality traits and customer-perceived value in online customer engagement,” *Journal of Marketing Management*, vol. 32, no. 5-6, pp. 502–525, 2016.
- [3] O. Toubia, A. T. Stephen, and A. Freud, “Viral marketing: a large-scale field experiment,” *Economics, Management, and Financial Markets*, vol. 6, no. 3, pp. 43–65, 2011.
- [4] P. Kotler and K. L. Keller, *Marketing Management*, Prentice Hall, 2011.
- [5] N. A. Diakopoulos and D. A. Shamma, “Characterizing debate performance via aggregated Twitter sentiment,” in *Proceedings of the 28th International Conference on Human Factors in Computing Systems - CHI '10*, pp. 1195–1198, Atlanta, GA, USA, 2010.
- [6] X. An, A. R. Ganguly, Y. Fang, S. B. Scyphers, A. M. Hunter, and J. G. Dy, “Tracking climate change opinions from Twitter data,” in *KDD 2014 Workshop on Data Science for Social Good*, New York, NY, USA, 2014.
- [7] J. Bollen, H. Mao, and X. Zeng, “Twitter mood predicts the stock market,” *Journal of Computational Science*, vol. 2, no. 1, pp. 1–8, 2011.
- [8] T. O. Sprenger, A. Tumasjan, P. G. Sandner, and I. M. Welp, “Tweets and trades: the information content of stock microblogs,” *European Financial Management*, vol. 20, no. 5, pp. 926–957, 2014.

- [9] D. A. Kenny, D. A. Kashy, and W. L. Cook, *Dyadic Data Analysis*, The Guilford Press, New York, NY, USA, 2006.
- [10] G. Gan, C. Ma, and J. Wu, *Data clustering: theory, algorithms, and applications*, Society for Industrial and Applied Mathematics, 2007.
- [11] T. Takahashi, R. Tomioka, and K. Yamanishi, "Discovering emerging topics in social streams via link anomaly detection," in *2011 IEEE 11th International Conference on Data Mining*, pp. 1230–1235, Vancouver, BC, Canada, 2011.
- [12] S. Aral, L. Muchnik, and A. Sundararajan, "Distinguishing influence-based contagion from homophily-driven diffusion in dynamic networks," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 106, no. 51, pp. 21544–21549, 2009.
- [13] R. Raghunathan and K. Corfman, "Is happiness shared doubled and sadness shared halved? Social influence on enjoyment of hedonic experiences," *Journal of Marketing Research*, vol. 43, no. 3, pp. 386–394, 2006.
- [14] M. Guerini, C. Strapparava, and G. Özbal, "Exploring text virality in social networks," in *Proceedings of the Fifth International Conference on Weblogs and Social Media (ICWSM 2011)*, pp. 506–509, Barcelona, Catalonia, Spain, 2011.
- [15] Y. T. Lu, S. I. Yu, T. C. Chang, and J. Y. Hsu, "A content-based method to enhance tag recommendation," in *Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI 2009)*, pp. 2064–2069, Pasadena, CA, USA, 2009.
- [16] D. M. Romero, W. Galuba, S. Asur, and B. A. Huberman, "Influence and passivity in social media," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases (ECML PKDD 2011)*, pp. 18–33, Athens, Greece, 2011.
- [17] A. Tversky, "Features of similarity," *Psychological Review*, vol. 84, no. 4, pp. 327–352, 1977.
- [18] C. X. Lin, Q. Mei, J. Han, Y. Jiang, and M. Danilevsky, "The joint inference of topic diffusion and evolution in social communities," in *2011 IEEE 11th International Conference on Data Mining*, pp. 378–387, Vancouver, BC, Canada, 2011.
- [19] O. P. John and S. Srivastava, "The big five trait taxonomy: history, measurement, and theoretical perspectives," in *Handbook of Personality: Theory and Research*, pp. 102–139, The Guilford Press, 1999.
- [20] G. Graham, "Behaviorism," in *The Stanford Encyclopedia of Philosophy*, Metaphysics Research Lab, 2017, <https://plato.stanford.edu/entries/behaviorism/>.
- [21] M. Granovetter, "Threshold models of collective behavior," *American Journal of Sociology*, vol. 83, no. 6, pp. 1420–1443, 1978.
- [22] S. Bai, T. Zhu, and L. Cheng, "Big-five personality prediction based on user behaviors at social network sites," 2012, <https://arxiv.org/abs/1204.4809>.
- [23] B. O'Connor, R. Balasubramanyan, B. R. Routledge, and N. A. Smith, "From tweets to polls: linking text sentiment to public opinion time series," in *Proceedings of the Fourth International Conference on Weblogs and Social Media (ICWSM 2010)*, pp. 122–129, Washington, DC, USA, 2010.
- [24] J. E. Chung and E. Mustafaraj, "Can collective sentiment expressed on Twitter predict political elections?," in *Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence (AAAI 2011)*, pp. 1770–1771, San Francisco, CA, USA, 2011.
- [25] S. Gaffney and P. Smyth, "Trajectory clustering with mixtures of regression models," in *Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '99*, pp. 63–72, San Diego, CA, USA, 1999.
- [26] K. Yang and C. Shahabi, "A PCA-based similarity measure for multivariate time series," in *Proceedings of the 2nd ACM International Workshop on Multimedia Databases - MMDB '04*, pp. 65–74, Washington, DC, USA, 2004.
- [27] K. Bhaduri, Q. Zhu, N. C. Oza, and A. N. Srivastava, "Fast and flexible multivariate time series subsequence search," in *2010 IEEE International Conference on Data Mining*, pp. 48–57, Sydney, NSW, Australia, 2010.
- [28] F. Mörchen, *Time Series Feature Extraction for Data Mining Using DWT and DFT*, [Ph.D. Thesis], University of Marburg, Department of Mathematics and Computer Science, 2003.
- [29] Y. R. Tausczik and J. W. Pennebaker, "The psychological meaning of words: LIWC and computerized text analysis methods," *Journal of Language and Social Psychology*, vol. 29, no. 1, pp. 24–54, 2010.
- [30] M. Bradley and P. J. Lang, "Affective norms for English words (ANEW): instruction manual and affective ratings," Tech. Rep. C-1, The Center for Research in Psychophysiology, University of Florida, Gainesville, FL, USA, 1999.
- [31] R. K. Sorde and S. N. Deshmukh, "Comparative study on approaches of recommendation system," *International Journal of Computer Applications*, vol. 118, no. 2, pp. 10–14, 2015.
- [32] M. R. McLaughlin and J. L. Herlocker, "A collaborative filtering algorithm and evaluation metric that accurately model the user experience," in *Proceedings of the 27th Annual International Conference on Research and Development in Information Retrieval - SIGIR '04*, pp. 329–336, Sheffield, UK, 2004.
- [33] F. C. Cruz, E. F. Simas Filho, M. C. S. Albuquerque, I. C. Silva, C. T. T. Farias, and L. L. Gouvêa, "Efficient feature selection for neural network based detection of flaws in steel welded joints using ultrasound testing," *Ultrasonics*, vol. 73, pp. 1–8, 2017.
- [34] A. Anagnostopoulos, R. Kumar, and M. Mahdian, "Influence and correlation in social networks," in *Proceeding of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD 08*, pp. 7–15, Las Vegas, NV, USA, 2008.
- [35] E. Diaz-Aviles, L. Drumond, L. Schmidt-Thieme, and W. Nejdl, "Real-time top-n recommendation in social streams," in *Proceedings of the Sixth ACM Conference on Recommender Systems - RecSys '12*, pp. 59–66, Dublin, Ireland, 2012.
- [36] P. Cremonesi, Y. Koren, and R. Turrin, "Performance of recommender algorithms on top-n recommendation tasks," in *Proceedings of the Fourth ACM Conference on Recommender Systems - RecSys '10*, pp. 39–46, Barcelona, Spain, 2010.
- [37] Ö. Celma and P. Cano, "From hits to niches?: or how popular artists can bias music recommendation and discovery," in *Proceedings of the 2nd KDD Workshop on Large-Scale Recommender Systems and the Netflix Prize Competition - NETFLIX '08*, p. 5, Las Vegas, NV, USA, 2008.
- [38] M. Levandowsky and D. Winter, "Distance between sets," *Nature*, vol. 234, no. 5323, pp. 34–35, 1971.
- [39] Y. Koren, "Factor in the neighbors: scalable and accurate collaborative filtering," *ACM Transactions on Knowledge Discovery from Data*, vol. 4, no. 1, pp. 1–24, 2010.

- [40] J. Berger and K. L. Milkman, "What makes online content viral?," *Journal of Marketing Research*, vol. 49, no. 2, pp. 192–205, 2012.
- [41] R. Bell, Y. Koren, and C. Volinsky, "Modeling relationships at multiple scales to improve accuracy of large recommender systems," in *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '07*, pp. 95–104, San Jose, CA, USA, 2007.
- [42] M. McPherson, L. Smith-Lovin, and J. M. Cook, "Birds of a feather: homophily in social networks," *Annual Review of Sociology*, vol. 27, no. 1, pp. 415–444, 2001.
- [43] A. R. Soetevent, "Empirics of the identification of social interactions: an evaluation of the approaches and their results," *Journal of Economic Surveys*, vol. 20, no. 2, pp. 193–228, 2006.
- [44] L. AlSumait, D. Barbará, and C. Domeniconi, "On-line LDA: adaptive topic models for mining text streams with applications to topic detection and tracking," in *2008 Eighth IEEE International Conference on Data Mining*, pp. 3–12, Pisa, Italy, 2008.
- [45] R. Kobayashi and R. Lambiotte, "TiDeH: time-dependent Hawkes process for predicting retweet dynamics," in *Proceedings of the Tenth International Conference on Web and Social Media (ICWSM 2016)*, pp. 191–200, Cologne, Germany, 2016.
- [46] J. Berger and K. L. Milkman, *Social Transmission, Emotion, and the Virality of Online Content*, Wharton Research Paper, 2010.
- [47] D. Boyd, S. Golder, and G. Lotan, "Tweet, tweet, retweet: conversational aspects of retweeting on Twitter," in *2010 43rd Hawaii International Conference on System Sciences*, pp. 1–10, Honolulu, HI, USA, 2010.
- [48] G. Miller, "Social scientists wade into the tweet stream," *Science*, vol. 333, no. 6051, pp. 1814–1815, 2011.
- [49] M. Naaman, H. Becker, and L. Gravano, "Hip and trendy: characterizing emerging trends on Twitter," *Journal of the American Society for Information Science and Technology*, vol. 62, no. 5, pp. 902–918, 2011.
- [50] R. Agrifoglio, S. Black, C. Metallo, and M. Ferrara, "Extrinsic versus intrinsic motivation in continued Twitter usage," *Journal of Computer Information Systems*, vol. 53, no. 1, pp. 33–41, 2012.
- [51] Z. Wen and C. Y. Lin, "How accurately can one's interests be inferred from friends," in *Proceedings of the 19th International Conference on World Wide Web - WWW '10*, pp. 1203–1204, Raleigh, NC, USA, 2010.
- [52] P. Bogdanov, M. Busch, J. Moehlis, A. K. Singh, and B. K. Szymanski, "The social media genome: modeling individual topic-specific behavior in social media," in *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining - ASONAM '13*, pp. 236–242, Niagara, ON, Canada, 2013.
- [53] S. V. Paunonen and D. N. Jackson, "The Jackson Personality Inventory and the five-factor model of personality," *Journal of Research in Personality*, vol. 30, no. 1, pp. 42–59, 1996.
- [54] R. Helson and V. S. Y. Kwan, "Personality development in adulthood: the broad picture and processes in one longitudinal sample," in *Advances in Personality Psychology*, pp. 77–106, Psychology Press, 2000.
- [55] B. W. Roberts, K. E. Walton, and W. Viechtbauer, "Patterns of mean-level change in personality traits across the life course: a meta-analysis of longitudinal studies," *Psychological Bulletin*, vol. 132, no. 1, pp. 1–25, 2006.
- [56] M. Cha, H. Haddadi, F. Benevenuto, and P. K. Gummadi, "Measuring user influence in Twitter: the million follower fallacy," in *Proceedings of the Fourth International Conference on Weblogs and Social Media (ICWSM 2010)*, pp. 10–17, Washington, DC, USA, 2010.
- [57] A. Suh and K. S. Shin, "Exploring the effects of online social ties on knowledge sharing: a comparative analysis of collocated vs dispersed teams," *Journal of Information Science*, vol. 36, no. 4, pp. 443–463, 2010.

Research Article

Supervised Learning for Suicidal Ideation Detection in Online User Content

Shaoxiong Ji ^{1,2}, Celina Ping Yu,³ Sai-fu Fung,⁴ Shirui Pan ⁵ and Guodong Long ⁵

¹University of Queensland, Brisbane, Australia

²University of Technology Sydney, Sydney, Australia

³Global Business College of Australia, Melbourne, Australia

⁴City University of Hong Kong, Kowloon, Hong Kong

⁵Centre for Artificial Intelligence, University of Technology Sydney, Sydney, Australia

Correspondence should be addressed to Shirui Pan; shirui.pan@uts.edu.au and Guodong Long; guodong.long@uts.edu.au

Received 1 February 2018; Revised 16 May 2018; Accepted 17 July 2018; Published 9 September 2018

Academic Editor: Gao Cong

Copyright © 2018 Shaoxiong Ji et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Early detection and treatment are regarded as the most effective ways to prevent suicidal ideation and potential suicide attempts—two critical risk factors resulting in successful suicides. Online communication channels are becoming a new way for people to express their suicidal tendencies. This paper presents an approach to understand suicidal ideation through online user-generated content with the goal of early detection via supervised learning. Analysing users' language preferences and topic descriptions reveals rich knowledge that can be used as an early warning system for detecting suicidal tendencies. Suicidal individuals express strong negative feelings, anxiety, and hopelessness. Suicidal thoughts may involve family and friends. And topics they discuss cover both personal and social issues. To detect suicidal ideation, we extract several informative sets of features, including statistical, syntactic, linguistic, word embedding, and topic features, and we compare six classifiers, including four traditional supervised classifiers and two neural network models. An experimental study demonstrates the feasibility and practicability of the approach and provides benchmarks for the suicidal ideation detection on the active online platforms: Reddit SuicideWatch and Twitter.

1. Introduction

Suicide might be considered as one of the most serious social health problems in the modern society. Many factors can lead to suicide, for example, personal issues, such as hopelessness, severe anxiety, schizophrenia, alcoholism, or impulsivity; social factors, like social isolation and overexposure to deaths; or negative life events, including traumatic events, physical illness, affective disorders, and previous suicide attempts. Thousands of people around the world fall victims to suicide every year, making suicide prevention become a critical global public health mission.

Suicidal ideation or suicidal thoughts are people's thoughts of committing suicide. It can be regarded as a risk indicator of suicide. Suicidal thoughts include fleeting thoughts, extensive thoughts, detailed planning, role playing,

and incomplete attempts. According to a WHO report [1], 788,000 people estimated worldwide committed suicide in 2015. And a large number of people, especially teenagers, were reported having suicidal ideation. Thus, one possible approach to preventing suicide effectively is early detection of suicidal ideation.

With the widespread emergence of mobile Internet technologies and online social networks, there is a growing tendency for people to talk about their suicide intentions in online communities. This online content could be helpful for detecting individuals' intentions and their suicidal ideation. Some people, especially adolescents, choose to post their suicidal thoughts in social networks, ask about how to commit suicide in online communities, and enter into online suicide pacts. The anonymity of online communication also allows people to freely express the pressures and anxiety they

suffer in the real world. This online user-generated content provides another possible angle for early suicide detection and prevention.

Previous research on suicide understanding and prevention mainly concentrates on its psychological and clinical aspects [2]. Recently, many studies have turned to natural language processing methods and classifying questionnaire results via supervised learning, which learns a mapping function from labelled training data [3]. Some of these researches have used the “International Personal Examination Screening Questionnaire,” and analysed suicide blogs and posts from social networking websites. However, these studies have their limitations. (1) From both a psychological and a clinical perspective, collecting data and/or patients is typically expensive, and some online data may help in understanding thoughts and behaviours. (2) Simple feature sets and classification models are not predictive enough to detect suicidal tendencies.

In this paper, we investigate the problem of suicidal ideation detection in online social websites, with a focus on understanding and detecting the suicidal thoughts in online user content. We perform a thorough analysis of the content, the language preferences, and the topic descriptions to understand the suicidal thoughts from a data mining perspective. Six different sets of informative features were extracted and six supervised learning algorithms were compared to detect suicidal ideation within the data. It is a novel application of automatic suicidal intention detection on social content with the combination of our proposed effective feature engineering and classification models.

This paper makes notable contributions and novelties to the literature in the following respects:

- (1) Knowledge discovery: this is a novel application of knowledge discovery and data mining to detect suicidal ideation in online user content. Previous work in this field has been conducted by psychological experts with statistical analysis; this approach reveals knowledge on suicidal ideation from a data analytic perspective. Insights from our analysis reveal that suicidal individuals often use personal pronouns to show their ego. They are more likely to use words expressing negativity, anxiety, and sadness in their dialogue. They are also more likely to choose the present tense to describe their suffering and the future tense to describe their hopelessness and plans for suicide.
- (2) Dataset and platform: this paper introduces the Reddit platform and collects a new dataset for suicidal ideation detection. Reddit’s SuicideWatch BBS is a new online channel for people with suicidal ideation to express their anxiety and pressures. Social volunteers respond in positive, supportive ways to relieve the depression and hopefully prevent potential suicides. This data source is not only useful for suicide detection but also for studying how to effectively prevent suicide through effective online communication.

- (3) Features, models, and benchmarking: rather than using basic models with simple features for suicidal ideation detection, this approach (1) identifies informative features from a number of perspectives, including statistical, syntactic, linguistic, word embedding features, and topic features; (2) compares with different classifiers from both traditional and deep learning perspectives, such as support vector machine [4], Random Forest [5], gradient boost classification tree (GBDT) [6], XGBoost [7], multilayer feed forward neural net (MLFFNN) [8], and long short-term memory (LSTM) [9]; and (3) provides benchmarks for suicidal ideation detection on SuicideWatch on Reddit, one active online forum for communication about suicide.

This paper is organised as follows. In Section 2, we review the related works on suicide analysis and detection. We introduce the datasets in Section 3 along with data exploration and knowledge discovery. Section 4 describes the classification and feature extraction methods. Section 5 is the experimental study. We conclude this paper in Section 6.

2. Related Works

Suicide detection has drawn the attention of many researchers due to an increasing suicide rate in recent years. The reasons of suicide are complicated and attributed to a complex interaction of many factors [10]. The research techniques used to examine suicide also span many fields and methods. For example, clinical methods may examine resting-state heart rate [11] and event-related instigators [12]. Classical methods also include using questionnaires to assess the potential risk of suicide and applying clinician-patient interactions [13].

The goal of text-based suicide classification is to determine whether candidates, through their posts, have suicidal ideation. Such techniques include suicide-related keyword filtering [14, 15] and phrase filtering [16].

Machine learning methods especially supervised learning and natural language processing methods have also been applied in this field. The main features consist of N -gram features, knowledge-based features, syntactic features, context features, and class-specific features [17]. Besides, word embedding [18] and sentence embedding [19] are well applied. Models for cybersuicide detection include regression analysis [20], ANN [21], and CRF [22]. Okhapkina et al. built a dictionary of terms pertaining to suicidal content and introduced term frequency-inverse document frequency (TF-IDF) matrices for messages and a singular vector decomposition for matrices [23]. Mulholland and Quinn extracted vocabulary and syntactic features to build a classifier for suicidal and nonsuicidal lyricists [24]. Huang et al. built a psychological lexicon dictionary and used an SVM classifier to detect cybersuicide [25]. Chattopadhyay [8] proposed a mathematical model using Beck’s suicide intent scale and applying multilayer feed-forward neural network to classify suicide intent. Pestian et al. [26] and Delgado-Gomez et al. [27] compared the performance of different multivariate techniques.

TABLE 1: Annotation rules and examples of social texts.

Categories	Rules	Examples
Suicide text	(i) Expressing suicidal thoughts	<i>I want to end my life tonight.</i>
	(ii) Including potential suicidal actions	<i>Yesterday, I tried to cut my wrist, but failed.</i>
Nonsuicide text	(i) Formally discussing suicide	<i>The global suicide rate is increasing.</i>
	(ii) Referring to other’s suicide	<i>I am so sad to hear that Robin Williams ended his life.</i>
	(iii) Not relevant to suicide	<i>I love this TV show and watch every week.</i>

The relevant extant research can also be viewed according to the data source.

2.1. Questionnaires. Mental disorder scale criteria such as DSM-IV (<https://www.psychiatry.org/psychiatrists/practice/dsm>) and ICD-10 (<http://apps.who.int/classifications/icd10/browse/2016/en>), and the “International Personal Disorder Examination Screening Questionnaire” (IPDE-SQ) provide good tools for evaluating an individual’s mental status and their potential for suicide. Delgado-Gomez et al. classified the results of IPDE-SQs based on “Barrat’s Impulsiveness Scale” (version 11) [28] and the “Holmes-Rahe Social Readjustment Rating Scale” to identify people likely to attempt suicide [27].

2.2. Suicide Notes. Suicide notes provide material for natural language processing. Previous approaches have examined suicide notes using content analysis [26], sentiment analysis [17, 29], and emotion detection [22]. In the age of cyberspace, suicide notes are now also written in the form of web blogs and can be identified as carrying the potential risk of suicide [14].

2.3. Online User Content. Cash et al. [30], Shepherd et al. [31], and Jashinsky et al. [16] have conducted psychology-based data analysis for content that suggests suicidal tendencies in the MySpace and Twitter social networks. Ren et al. explored accumulated emotional information from online suicide blogs [32]. O’Dea et al. developed automatic suicide detection on Twitter by applying logistic regression and SVM on TF-IDF features [33]. Reddit has also attracted much research interest. Huang and Bashir applied linguistic cues to analyse the reply bias [34]. De Choudhury et al. did many works on suicide-related topics in Reddit including the effect of celebrity suicides on suicide-related content [35] and the transition from mental health illness to suicidal ideation [36].

A questionnaire is a useful tool for collecting data, but it costs highly. Suicide notes are useful materials for training a classifier. The current dataset of suicide notes is quite small. Automatic detection on online user content will be a promising way for suicide detection and prevention. Our proposed method investigated a better solution with effective feature engineering on a bigger social dataset than the previous work. And it can adapt to real-world application with the ability of automatic detection compared with questionnaires.

3. Data and Knowledge

We collect the suicidal ideation texts from Reddit and Twitter and manually check all the posts to ensure they were

TABLE 2: Two balanced Reddit datasets.

Dataset	Subreddits
1	SuicideWatch versus others (nonsuicide)
2	SuicideWatch versus gaming
	SuicideWatch versus jokes
	SuicideWatch versus books
	SuicideWatch versus movies
	SuicideWatch versus AskReddit

correctly labelled. Our annotation rules and examples of posts appear in Table 1.

3.1. Reddit Dataset. Reddit is a registered online community that aggregates social news and online discussions. It consists of many topic categories, and each area of interest within a topic is called a subreddit.

In this dataset, online user content includes a title and a body of text. To preserve privacy, we replace personal information with a unique ID to identify each user. We collected posts with potential suicide intentions from a subreddit called “SuicideWatch”(SW) (<https://www.reddit.com/r/SuicideWatch/>). Posts without suicidal content were sourced from other popular subreddits (<https://www.reddit.com/r/all/>, <https://www.reddit.com/r/popular/>). The collection of nonsuicidal data is totally a user-generated content, and the posts of news aggregation and administrator are excluded. To facilitate the study and demonstration, we will study the balanced dataset in Reddit and study imbalanced dataset in Twitter in the following subsection.

The Reddit dataset includes 3549 suicidal ideation samples and a number of nonsuicide texts. In particular, we construct two datasets for Reddit as shown in Table 2. The first dataset includes two subreddits in which one is from SuicideWatch and another is from popular posts in Reddit. The second dataset is composed of six subreddits that include SuicideWatch and another five hot topics: gaming (<https://www.reddit.com/r/gaming/>), jokes (<https://www.reddit.com/r/Jokes/>), books (<https://www.reddit.com/r/books/>), movies (<https://www.reddit.com/r/movies/>), and AskReddit (<https://www.reddit.com/r/AskReddit/>). In the second dataset, the combination of SuicideWatch with any other subreddit will be a new balanced subdataset, for example, suicide versus gaming and suicide versus jokes. These two datasets will be studied on Subsections 5.1 and 5.2 separately.



FIGURE 1: Word cloud visualisation of suicidal texts in Reddit and Twitter.

TABLE 3: Linguistic statistical information extracted by LIWC.

Average word count	Suicide	Nonsuicide
Personal nouns	30.01	14.6
Quantifiers	3.78	3.37
Positive emotion	5.61	7.84
Negative emotion	11.12	4.89
Anxiety	1.46	0.55
Sadness	3.86	0.63
Past focus	6.78	6.27
Present focus	34.81	17.86
Future focus	4.06	1.76
Family	1.07	0.82
Friend	1.02	0.78
Female references	0.95	1.35
Male references	1.03	2.40
Work	2.50	3.92
Money	0.60	1.38
Death	4.81	0.61
Swear words	1.47	1.62

3.2. Twitter Dataset. Many online users also want to talk about the suicidal ideation in social networks. However, Twitter is quite different with Reddit as (1) each tweet’s length is limited in 140 characters (this limit is now 280 characters), (2) tweet users may have some social network friends from the real world while Reddit users are fully anonymous, and (3) the communication and interaction type are totally different between social networking websites and online forums.

The Twitter dataset is collected using a keyword filtering technique. Suicidal words and phrases include “suicide,” “die,” and “end my life.” Many of the collected tweets have the suicidal-related words, but they possibly talk about a suicide movie or advertisement which does not contain suicidal ideation. Therefore, we manually checked and labeled collected tweets according to the annotation rules in Table 1. Finally, the Twitter dataset has totally 10,288 tweets with 594 tweets (around 6%) with suicidal ideation. This dataset is an imbalanced dataset and will be studied in Section 5.3.

3.3. Data Exploration and Knowledge Discovering. To understand suicidal individuals, we analysed the words, languages, and topics in online user content.

3.3.1. Word Cloud. Word clouds were used to provide a visual understanding of the data. The users’ posts in Reddit and tweets in Twitter with potential suicide risk are showed separately in Figures 1(a) and 1(b). As we can see, suicidal posts frequently use words such as “life,” “suicide,” and “kill,” providing a direct indication of the users’ suicidal thoughts. Words expressing feelings or intentions are also frequently used, such as “feel,” “want,” and “know.” For example, some suicidal posts wrote, “I feel like I have no one left and I want to end it,” “I want to end my life,” and “I don’t know how much of it was psychological trauma.”

In addition, the dominant words in these two social platforms have different styles due to the posting rules of the platforms. The Reddit users are willing to compose their posts in a specific way. For instance, they describe their life events and their stories about their friends. While the content in Twitter is much more straightforward with expressions like “want kill,” “going kill,” and “wanna kill.” The details are usually not included in their tweets.

3.3.2. Language Preferences. Language preferences provide an overview of the statistical linguistic information of the data. The listed variables shown in Table 3 were extracted using LIWC 2015 [37]. All these categories are features based on word counts. We calculated the average value of each variable in both suicide-related texts and suicide-free posts. As shown in the table, content with or without suicidality quite differs in many items.

- (i) Users with suicidal ideation use many personal pronouns to show their ego. For example, “I want to end my life.”
- (ii) They express more negative emotions, like anxiety and sadness. For example, “I was drowning in guilt and depression for several years after.”
- (iii) As for the tense, texts with suicidal ideation tend to use the present and future tense. They tend to use the present tense to describe their suffering, pain, and depression. For example, “I’m feeling so bad.” The future tense is used to describe their hopeless feelings about the future and their suicide intentions. For example, “I’m eventually going to kill myself.”
- (iv) Both types of posts discuss family and friends and make female or male references.

TABLE 4: Topic words extracted from posts containing suicidal thoughts.

Number	Top 10 words for each suicide-related topics in SuicideWatch
1	Money, working, suicide, gun, fucked, come, yet, failed, erase, thats
2	Said, got, went, started, friend, back, father, told, mother, girl
3	Im, school, go, year, time, know, one, ive, day, got
4	Mm, dont, its, ive, cant, get, know, around, time, pain
5	Im, feel, like, want, know, friend, would, life, get, time
6	Imagine, cellophane, abandoned, anyone, medical, cheated, mr, surgery, yelling, letter
7	Im, want, life, like, get, feel, ive, know, year, even
8	Fucking, very, tomorrow, bottom, accept, sharp, n't, went, wife, attacked
9	Condition, suicide, also, hope, tx, california, chronic, jumping, crisis, age
10	Please, find, mother, car, social, live, need, accident, debt, month

(v) Unsurprisingly, more words related to death appear in texts about suicide. For example, “kill,” “die,” “end life,” and “suicide.”

(vi) Both types of posts contain a similar number of swear words.

One of the findings from Table 3 and Figure 1 is that people with suicidal thoughts tend to directly show their intentions in anonymous online communities when faced with some kinds of problem in the real world. Their posts often show negative feelings with strong ego and intention.

3.3.3. Topic Description. We extracted 10 topics from posts containing suicidal ideation using the latent Dirichlet allocation (LDA) [38] topic modelling method, as shown in Table 4. There are some Internet slangs such as “tx” (thanks) and abbreviations like “im” (I am) and “n’t” (“negatory”). In the field of standard natural language processing, personal words like “I,” “me,” and “you” are stop words and should be removed, but we kept them in this exploration because they contain important information. Thus, there are many personal pronouns included in these topic words, which are identical to the results in Table 3.

Interestingly, we observed that posts containing suicidal themes could be summarised into three categories: internal factors, external social factors, and mixed internal/external factors. Specifically, internal factors, including words like “know” (topics 3, 4, 5, and 7), “want,” “feel” and “like” (topics 5 and 7), and “hope” (topic 9) express people’s feelings, intentions, and desires, while other words such as “money” and “working” (topic 1), “friend” (topics 2 and 5), “school” (topic 3), “surgery” (topic 6), “crisis” (topic 9), and “accident” (topic 10) indicate that posts are linked to social factors. In topic 3, 5, 9, and 10, both factors are represented.

4. Methods and Technical Solutions

4.1. Feature Processing. By preprocessing and cleaning the data in advance, we extracted several features including statistics, word-based features (e.g., suicidal words and pronouns), TF-IDF, semantics, and syntactics. Additionally, we used distributed features by training neural networks to

embed word into vector representations, along with topic features extracted by LDA [38] as unsupervised features.

4.1.1. Statistical Features. User-generated posts are varied in length, and some statistical features can be extracted from texts. Some posts use short and simple sentences, while others use complex sentences and long paragraphs.

After segmentation and tokenisation, we captured statistical features as follows:

- (i) The number of words, tokens, and characters in the title
- (ii) The number of words, tokens, characters, sentences, and paragraphs in the text body

4.1.2. Syntactic Features: POS. Syntactic features are useful information in natural language processing tasks. We extracted parts of speech (POS) [39] as features for our suicidal ideation detection model to capture the similar grammatical properties in users’ posts.

Common POS tags include nouns, verbs, participles, articles, pronouns, adverbs, and conjunctions. POS subgroups were also identified to provide more detail about the grammatical properties of the posts. Each post was parsed and tagged, and the number of each category in the title and text body was simply counted.

4.1.3. Linguistic Features: LIWC. Online users’ posts usually contain emotions, relativity, and harassment words. Lexicons are widely applied for extracting these features. To analyse the linguistic and emotional features in the data, we used Linguistic Inquiry and Word Count [37] (LIWC 2015 (<http://liwc.wpengine.com/>)) which was proposed and developed by the University of Texas at Austin. This approach was used in a previous study [34]. The tool contains a powerful internally built dictionary for matching the target words in posts when parsing data. About 90 variables were output. In addition to word count-based features, it could extract features based on emotional tone, cognitive processes, perceptual processes, and many types of abusive words. Specific categories include word count, summary language, general descriptors, linguistic

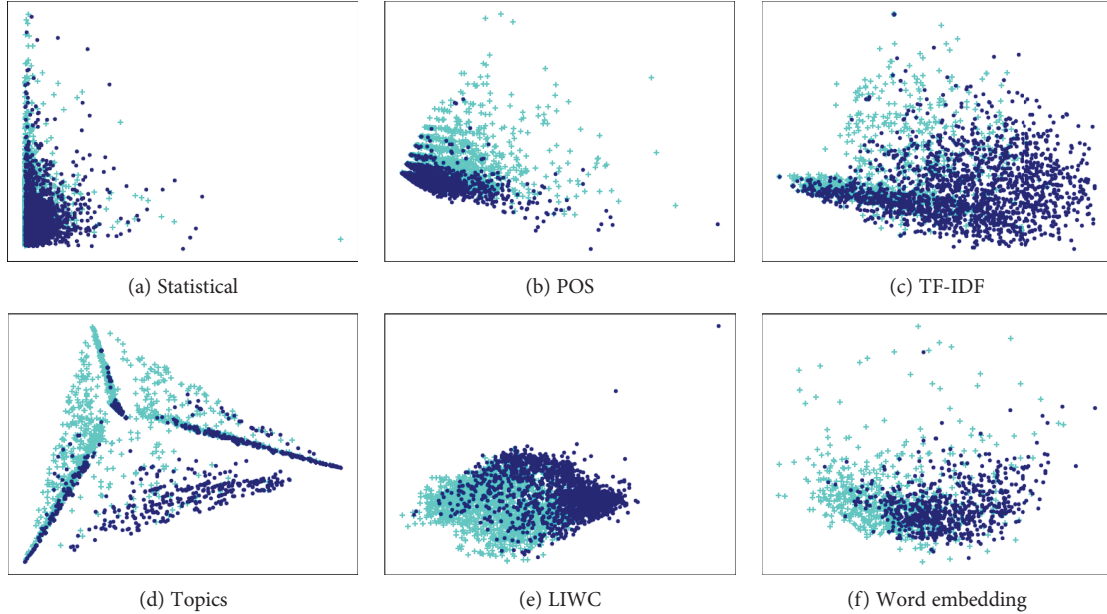


FIGURE 2: Visualisation of extracted features using PCA.

dimensions, psychological constructs, personal concern, informal language markers, and punctuation.

4.1.4. Word Frequency Features: TF-IDF. Many kinds of expression are related to suicide. We used TF-IDF to extract these features and measure the importance of various words from both suicidal posts and nonsuicidal posts. TF-IDF measures the number of times that each word occurs in the documents and adds a penalty depending on the frequency of the word in the entire corpus.

4.1.5. Word Embedding Features. The distributed representation, which is able to preserve the semantic information in texts, is popular and useful for many natural language processing tasks. It embeds words into a vector space. There are several techniques for word embedding. We employed the *word2vec* ([18], <https://code.google.com/archive/p/word2vec/>) to derive a distributed semantic representation of the words.

There are two architectures for word2vec word embedding, that is, CBOW and Skip-gram. CBOW predicts the present word based on the context, Skip-gram predicts the closest words to the current word provided.

4.1.6. Topic Features. Suicidal posts and nonsuicidal posts talk about different topics which can provide good understanding for two categories. We applied the latent Dirichlet allocation (LDA) [38] to reveal latent topics in user posts. Each topic is a mixture probability of word occurrence in the topic, and each post is a mixture probability of topics.

Given the set of documents and the number of topics, we used LDA to extract the topics from each posts, then calculate the probability that each post belonged to every generated topics. Hence, the posts are represented by their

thematic properties as probability vectors at the length of the number of topics.

(1) Feature Visualisation. To understand the informativeness of these feature sets, we visualise the features on the Reddit dataset in a 2-dimensional space by using principal component analysis (PCA) [40] in Figure 2. The results demonstrate that we indeed extract features that can largely separate the points in different classes. We will further validate the effectiveness of our feature sets in Section 5.

4.2. Classification Models. Suicidality detection in social content is a typical classification problem of supervised learning. Given a dataset $\{x_i, y_i\}_i^n$ consisting a set of texts $\{x_i\}_i^n$ with labels $\{y_i\}_i^n$, we trained a supervised classification model to learn the function from the training data pairs of input objects and supervisory signals:

$$y_i = F(x_i), \quad (1)$$

where $y_i = 1$ means that the expression x_i is “suicide text” (ST), otherwise $y_i = 0$ means “not suicide text (non-ST).” The training or learning of the classification model is to minimise the prediction error in the given training data. The prediction error is to be presented as a loss function $L(y, F(x))$ where y is the real label and $F(x)$ is the predicted label by using classification model. In summary, the goal of training algorithm is to obtain an optimal prediction model $F(x)$ by solving below optimisation task:

$$\hat{F} = \underset{F}{\operatorname{argmin}} \mathbb{E}_{x,y} [L(y, F(x))]. \quad (2)$$

Different classification methods may have different definition of loss function and predefined structure of model. We employed both classical supervised learning classification

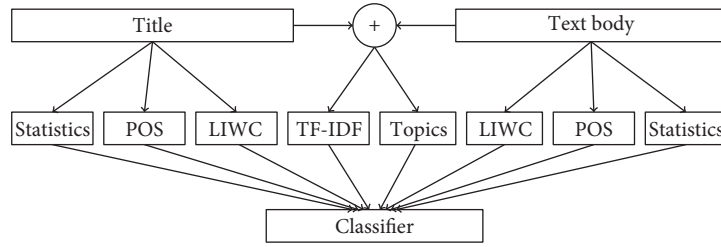


FIGURE 3: The model's structure for Reddit dataset.

methods and deep learning methods to solve the suicidal ideation classification task.

The structure of our feature extraction method is shown in Figure 3. As mentioned in Section 4.1, features comprised statistics, POS counts, LIWC features, TF-IDF vectors, and topic probability features. Among these features, we applied POS features and LIWC features to both the title and text body of user posts. We combined the title and the body into one piece of text to extract topic probability vectors and TF-IDF vectors. All extracted features were input to the classifiers.

5. Empirical Evaluation

5.1. Comparison and Analysis on Suicide versus Nonsuicide. This section compares various classification methods using different combinations of features with 10-fold cross validation (Our codes are available in <https://github.com/shaoxiongji/sw-detection>). The specific classification models include support vector machine [4], Random Forest [5], gradient boost classification tree (GBDT) [6], XGBoost [7], and multilayer feed-forward neural net (MLFFNN) [8]. SVM is able to solve problems that are not linearly separable in lower space by constructing a hyperplane in high-dimensional space. It can be adapted to many kinds of classification tasks [41, 42]. Random Forest, GBDT, and XGBoost are tree ensemble methods that use decision trees as base classifiers and produce a form of committee to gain better performance than any single base classifier. MLFFNN takes the different features as input and learns the combination of them with nonlinearity.

For comparison and to solve the problem of understanding the semantic meaning and syntactic structure of sentences, deep learning provides powerful performance [43]. We used long short-term memory (LSTM) [9] network, one state-of-the-art deep neural network. LSTM takes the title and text body of user posts with word embedding as its inputs and uses memory cell to preserve the state over long periods, capturing the long-term dependencies in long conversation detection.

As shown in Table 5, all methods' performance increases by combining more features on the whole. This observation validates the effectiveness and informativeness of our extracted features. However, the contribution each feature makes varies, which leads to fluctuations in the results of individual methods. The XGBoost had the best performance of the six methods when taking all groups of features as

inputs. Although LSTM does not require feature processing and is renowned for its state-of-the-art performance in many other natural language processing tasks, it did not perform as well as some of the other ensemble learning methods with sufficient features in this case. Random Forest, GBDT, XGBoost, and MLFFNN with proper features produced better accuracy and F1 scores than LSTM on our Reddit dataset. Admittedly, deep learning with word embedding is rather convenient and typically achieves adequate results, even without complicated feature engineering.

The AUC performance measurement in each classification is the area under the receiver operating characteristic curve with all extracted features. In the last column of Table 5, the AUC has an increasing tendency with more combined features. The XGBoost method gains the highest AUC of 0.9569 while other methods have very similar AUC value above 0.9.

5.2. Suicide versus Single Subreddit Topics. To evaluate the classification on suicide with any other specific online communities, we extended our datasets and experiments to other specific subreddits, including "gaming," "jokes," "books," "movies," and "AskReddit."

The results are shown in Figure 4. Using the features extracted with our approach was a very effective way of classifying the suicidal ideation posts from another subreddit domain. In fact, the classification results on suicidal dataset versus the subreddit dataset were better than suicidal versus nonsuicidal dataset where the nonsuicidal samples are composed of multiple popular subreddit domains. In these experiments, XGBoost produced the best results on "movies" and "AskReddit" in terms of accuracy and F1 scores. LSTM and Random Forest outperformed the other models in "gaming" and "books," respectively.

5.3. Experiments on Twitter Dataset. To evaluate the performance of our proceeded features and the classification models, we do another experiment on our Twitter dataset. Tweet text without long text body is different with Reddit text. Thus, for the experimental setting, there is a slight difference between them. We exclude the number of paragraphs in statistical features, POS, and LIWC features of text bodies. The rest of the settings are similar to our previous experiment. Considering the class imbalance in Twitter data, we adopt undersampling techniques. The results are the average metrics of each undersampled data shown in Table 6. The receiver operating characteristic curves of

TABLE 5: Comparison of different methods using different features.

Methods	Features	Acc.	Prec.	Recall	F1-score	AUC
SVM	Statistics	0.8064	0.8045	0.8189	0.8116	0.8061
	Statistics + topic	0.8609	0.881	0.8406	0.8603	0.8613
	Statistics + topic + TF-IDF	0.8571	0.8414	0.8865	0.8634	0.8565
	Statistics + topic + TF-IDF + POS	0.8674	0.8545	0.8916	0.8727	0.8670
	Statistics + topic + TF-IDF + POS + LIWC	0.9123	0.9144	0.9133	0.9138	0.9123
Random Forest	Statistics	0.7732	0.8094	0.7258	0.7653	0.7741
	Statistics + topic	0.8973	0.8922	0.9082	0.9001	0.8971
	Statistics + topic + TF-IDF	0.8915	0.8795	0.912	0.8954	0.8911
	Statistics + topic + TF-IDF + POS	0.8986	0.8801	0.9273	0.9031	0.8981
	Statistics + topic + TF-IDF + POS + LIWC	0.9357	0.9213	0.9554	0.938	0.9353
GBDT	Statistics	0.7505	0.7632	0.7398	0.7513	0.7507
	Statistics + topic	0.898	0.8856	0.9184	0.9017	0.8976
	Statistics + topics + TF-IDF	0.896	0.89	0.9082	0.899	0.8958
	Statistics + topic + TF-IDF + POS	0.8928	0.8893	0.9018	0.8955	0.8926
	Statistics + topic + TF-IDF + POS + LIWC	0.9461	0.9354	0.9605	0.9478	0.9458
XGBoost	Statistics	0.7667	0.7822	0.7513	0.7664	0.7670
	Statistics + topic	0.8999	0.8938	0.912	0.9028	0.8997
	Statistics + topic + TF-IDF	0.9019	0.8941	0.9158	0.9049	0.9016
	Statistics + topic + TF-IDF + POS	0.9103	0.8998	0.9273	0.9133	0.9100
	Statistics + topic + TF-IDF + POS + LIWC	0.9571	0.9499	0.9668	0.9583	0.9569
MLFFNN	Statistics	0.7647	0.7742	0.7742	0.7742	0.7731
	Statistics + topic	0.8821	0.8740	0.8525	0.8631	0.8961
	Statistics + topic + TF-IDF	0.8606	0.8369	0.8401	0.8385	0.8855
	Statistics + topic + TF-IDF + POS	0.9068	0.9038	0.8868	0.8952	0.9369
	Statistics + topic + TF-IDF + POS + LIWC	0.9283	0.9391	0.9205	0.9295	0.9403
LSTM	word2vec word embedding	0.9266	0.9786	0.8750	0.9239	0.9276

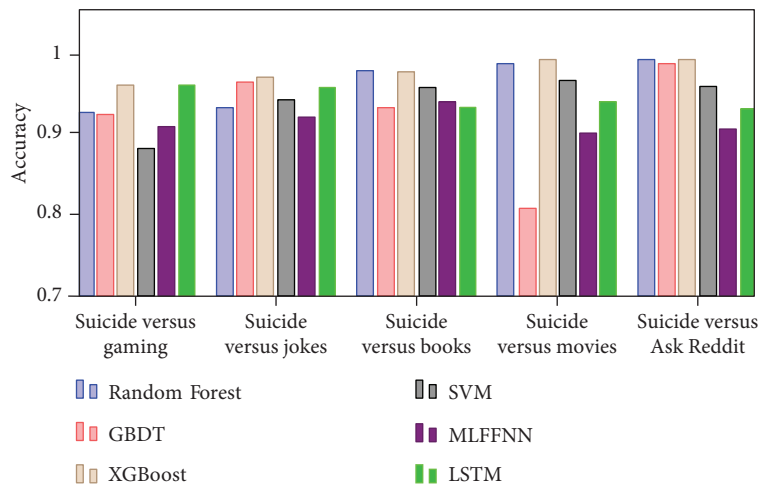


FIGURE 4: Classification for suicidal ideation of SuicideWatch versus other six subreddits.

these methods are showed in Figure 5. In these dataset, Random Forest gains better performance than most models except for the metric of precision in which the MLFFNN gains a slightly better result.

6. Conclusion

The amount of text keeps growing with the popularisation of social networking services. And suicide prevention

TABLE 6: Comparison of different models using all processed features on Twitter data.

Model	Acc.	Prec.	Recall	F1	AUC
Random Forest	0.9638	0.9638	0.9917	0.9646	0.9862
GBDT	0.9500	0.9413	0.9603	0.9503	0.9825
XGBoost	0.9591	0.9425	0.9782	0.9597	0.9843
SVM	0.9485	0.9261	0.9755	0.9497	0.9813
MLFFNN	0.9412	0.9661	0.9194	0.9421	0.9823
LSTM	0.9108	0.9399	0.8802	0.9059	0.9747

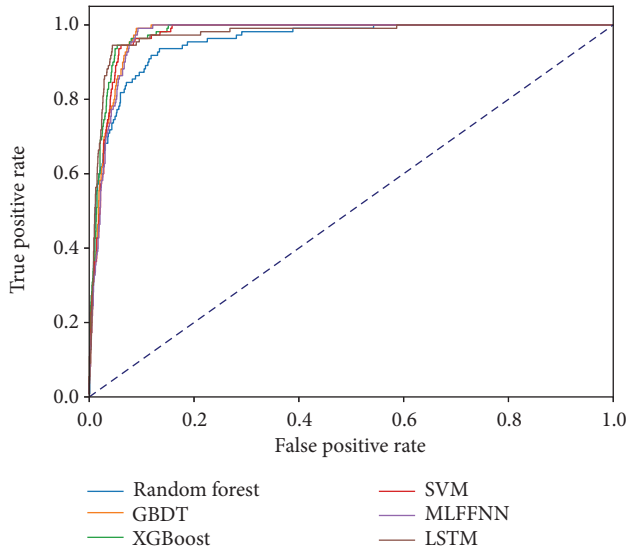


FIGURE 5: The receiver operating characteristic curve of six methods with all processed features.

remains an important task in our modern society. It is therefore essential to develop new methods to detect online texts containing suicidal ideation in the hope that suicide can be prevented.

In this paper, we investigated the problem of suicidality detection in online user-generated content. We argue that most work in this field was conducted by psychological experts with statistical analysis, which is limited by the cost and privacy issue in obtaining data. By collecting and analysing the anonymous online data from an active Reddit platform and Twitter, we provide rich knowledge that can complement the understanding of suicidal ideation and behaviour. Though applying feature processing and classification methods to our carefully built datasets, Reddit and Twitter, we evaluated, analysed, and demonstrated that our framework can achieve high performance (accuracy) in distinguishing suicidal thoughts out of normal posts in online user content.

While exploiting more effective feature sets, complex models or other factors such as temporal information may improve the detection of suicidal ideation—these will be our future directions; the contribution and impact of this paper are threefold: (1) delivering rich knowledge in understanding suicidal ideation, (2) introducing datasets for the

research community to study this significant problem, and (3) proposing informative features and effective models for suicidal ideation detection.

Data Availability

The data used to support the findings of this study are available from the first author upon request (email: shaoxiong.ji@uq.edu.au).

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.


References

- [1] "Suicide rates, Global Health Observatory (GHO) data," 2015, http://www.who.int/gho/mental_health/suicide_rates/en/.
- [2] V. Venek, S. Scherer, L. P. Morency, A. S. Rizzo, and J. Pestian, "Adolescent suicidal risk assessment in clinician-patient interaction," *IEEE Transactions on Affective Computing*, vol. 8, no. 2, pp. 204–215, 2017.
- [3] M. Mohri, A. Rostamizadeh, and A. Talwalkar, *Foundations of Machine Learning*, MIT Press, 2012.
- [4] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.
- [5] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.
- [6] J. H. Friedman, "Greedy function approximation: a gradient boosting machine," *The Annals of Statistics*, vol. 29, no. 5, pp. 1189–1232, 2001.
- [7] T. Chen and C. Guestrin, "XGBoost: a scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '16*, pp. 785–794, San Francisco, CA, USA, 2016, ACM.
- [8] S. Chattopadhyay, "A mathematical model of suicidal-intent-estimation in adults," *American Journal of Biomedical Engineering*, vol. 2, no. 6, pp. 251–262, 2012.
- [9] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [10] R. C. O'Connor and M. K. Nock, "The psychology of suicidal behaviour," *The Lancet Psychiatry*, vol. 1, no. 1, pp. 73–85, 2014.
- [11] D. Sikander, M. Arvaneh, F. Amico et al., "Predicting risk of suicide using resting state heart rate," in *2016 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, pp. 1–4, Jeju, Republic Korea, 2016, IEEE.
- [12] N. Jiang, Y. Wang, L. Sun, Y. Song, and H. Sun, "An ERP study of implicit emotion processing in depressed suicide attempters," in *2015 7th International Conference on Information Technology in Medicine and Education (ITME)*, pp. 37–40, Huangshan, China, 2015, IEEE.
- [13] W. C. Chiang, P. H. Cheng, M. J. Su, H. S. Chen, S. W. Wu, and J. K. Lin, "Socio-health with personal mental health records: suicidal-tendency observation system on Facebook for Taiwanese adolescents and young adults," in *2011 IEEE 13th International Conference on e-Health Networking, Applications and Services*, pp. 46–51, Columbia, MO, USA, 2011, IEEE.

- [14] Y. P. Huang, T. Goh, and C. L. Liew, "Hunting suicide notes in web 2.0 - preliminary findings," in *Ninth IEEE International Symposium on Multimedia Workshops (ISMW 2007)*, pp. 517–521, Beijing, China, 2007, IEEE.
- [15] K. D. Varathan and N. Talib, "Suicide detection system based on Twitter," in *2014 Science and Information Conference*, pp. 785–788, London, UK, 2014, IEEE.
- [16] J. Jashinsky, S. H. Burton, C. L. Hanson et al., "Tracking suicide risk factors through Twitter in the US," *Crisis*, vol. 35, no. 1, pp. 51–59, 2014.
- [17] W. Wang, L. Chen, M. Tan, S. Wang, and A. P. Sheth, "Discovering fine-grained sentiment in suicide notes," *Biomedical Informatics Insights*, vol. 5, Supplement 1, pp. 137–145, 2012.
- [18] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," 2013, <https://arxiv.org/abs/1301.3781>.
- [19] T. Shen, T. Zhou, G. Long, J. Jiang, S. Pan, and C. Zhang, "DiSAN: directional self-attention network for RNN/CNN-free language understanding," 2017, <https://arxiv.org/abs/1709.04696>.
- [20] S. Chattopadhyay, "A study on suicidal risk analysis," in *2007 9th International Conference on e-Health Networking, Application and Services*, pp. 74–78, Taipei, Taiwan, 2007, IEEE.
- [21] Y. M. Tai and H. W. Chiu, "Artificial neural network analysis on suicide and self-harm history of Taiwanese soldiers," in *Second International Conference on Innovative Computing, Informatio and Control (ICICIC 2007)*, pp. 363–363, Kumamoto, Japan, 2007, IEEE.
- [22] M. Liakata, J. H. Kim, S. Saha, J. Hastings, and D. Rebolz-Schuhmann, "Three hybrid classifiers for the detection of emotions in suicide notes," *Biomedical Informatics Insights*, vol. 5, Supplement 1, 2012.
- [23] E. Okhapkina, V. Okhapkin, and O. Kazarin, "Adaptation of information retrieval methods for identifying of destructive informational influence in social networks," in *2017 31st International Conference on Advanced Information Networking and Applications Workshops (WAINA)*, pp. 87–92, Taipei, Taiwan, 2017, IEEE.
- [24] M. Mulholland and J. Quinn, "Suicidal tendencies: the automatic classification of suicidal and non-suicidal lyricists using NLP," in *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pp. 680–684, Nagoya, Japan, 2013.
- [25] X. Huang, L. Zhang, D. Chiu, T. Liu, X. Li, and T. Zhu, "Detecting suicidal ideation in Chinese microblogs with psychological lexicons," in *2014 IEEE 11th Intl Conf on Ubiquitous Intelligence and Computing and 2014 IEEE 11th Intl Conf on Autonomic and Trusted Computing and 2014 IEEE 14th Intl Conf on Scalable Computing and Communications and Its Associated Workshops*, pp. 844–849, Bali, Indonesia, 2014, IEEE.
- [26] J. Pestian, H. Nasrallah, P. Matykiewicz, A. Bennett, and A. Leenaars, "Suicide note classification using natural language processing: a content analysis," *Biomedical Informatics Insights*, vol. 3, pp. 19–28, 2010.
- [27] D. Delgado-Gomez, H. Blasco-Fontecilla, F. Sukno, M. Socorro Ramos-Plasencia, and E. Baca-Garcia, "Suicide attempters classification: toward predictive models of suicidal behavior," *Neurocomputing*, vol. 92, pp. 3–8, 2012.
- [28] D. Delgado-Gomez, H. Blasco-Fontecilla, A. A. Alegria, T. Legido-Gil, A. Artes-Rodriguez, and E. Baca-Garcia, "Improving the accuracy of suicide attempter classification," *Artificial Intelligence in Medicine*, vol. 52, no. 3, pp. 165–168, 2011.
- [29] J. P. Pestian, P. Matykiewicz, M. Linn-Gust et al., "Sentiment analysis of suicide notes: a shared task," *Biomedical Informatics Insights*, vol. 5, Supplement 1, pp. 3–16, 2012.
- [30] S. J. Cash, M. Thelwall, S. N. Peck, J. Z. Ferrell, and J. A. Bridge, "Adolescent suicide statements on MySpace," *Cyberpsychology, Behavior, and Social Networking*, vol. 16, no. 3, pp. 166–174, 2013.
- [31] A. Shepherd, C. Sanders, M. Doyle, and J. Shaw, "Using social media for support and feedback by mental health service users: thematic analysis of a Twitter conversation," *BMC Psychiatry*, vol. 15, no. 1, p. 29, 2015.
- [32] F. Ren, X. Kang, and C. Quan, "Examining accumulated emotional traits in suicide blogs with an emotion topic model," *IEEE Journal of Biomedical and Health Informatics*, vol. 20, no. 5, pp. 1384–1396, 2016.
- [33] B. O'Dea, S. Wan, P. J. Batterham, A. L. Calear, C. Paris, and H. Christensen, "Detecting suicidality on Twitter," *Internet Interventions*, vol. 2, no. 2, pp. 183–188, 2015.
- [34] H. Y. Huang and M. Bashir, "Online community and suicide prevention: investigating the linguistic cues and reply bias," in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems - CHI'16*, San Jose, CA, USA, 2016.
- [35] M. Kumar, M. Dredze, G. Coppersmith, and M. De Choudhury, "Detecting changes in suicide content manifested in social media following celebrity suicides," in *Proceedings of the 26th ACM Conference on Hypertext & Social Media - HT '15*, pp. 85–94, Guzelyurt, Northern Cyprus, 2015, ACM.
- [36] M. De Choudhury, E. Kiciman, M. Dredze, G. Coppersmith, and M. Kumar, "Discovering shifts to suicidal ideation from mental health content in social media," in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems - CHI '16*, pp. 2098–2110, San Jose, California, USA, 2016, ACM.
- [37] J. W. Pennebaker, R. L. Boyd, K. Jordan, and K. Blackburn, *The Development and Psychometric Properties of LIWC2015*, University of Texas at Austin, 2015.
- [38] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation," *Journal of Machine Learning Research*, vol. 3, pp. 993–1022, 2003.
- [39] A. Voutilainen, "Part-of-speech tagging," in *The Oxford Handbook of Computational Linguistics*, pp. 219–232, Oxford University Press, 2003.
- [40] I. T. Jolliffe, "Principal component analysis and factor analysis," in *Principal Component Analysis*, pp. 115–128, Springer, 1986.
- [41] S. Pan, J. Wu, and X. Zhu, "CogBoost: boosting for fast cost-sensitive graph classification," *IEEE Transactions on Knowledge and Data Engineering*, vol. 27, no. 11, pp. 2933–2946, 2015.
- [42] S. Pan, J. Wu, X. Zhu, G. Long, and C. Zhang, "Task sensitive feature exploration and learning for multitask graph classification," *IEEE Transactions on Cybernetics*, vol. 47, no. 3, pp. 744–758, 2017.
- [43] S. Pan, R. Hu, G. Long, J. Jiang, L. Yao, and C. Zhang, "Adversarially regularized graph autoencoder," 2018, <https://arxiv.org/abs/1802.04407>.

Research Article

Weibo Attention and Stock Market Performance: Some Empirical Evidence

Minghua Dong,¹ Xiong Xiong,^{1,2} Xiao Li,³ and Dehua Shen ^{1,2}

¹College of Management and Economics, Tianjin University, Tianjin 300072, China

²China Center for Social Computing and Analytics, Tianjin University, Tianjin 300072, China

³School of Finance, Nankai University, Tianjin 300350, China

Correspondence should be addressed to Dehua Shen; dhs@tju.edu.cn

Received 29 December 2017; Revised 21 July 2018; Accepted 19 August 2018; Published 3 September 2018

Academic Editor: Shuliang Wang

Copyright © 2018 Minghua Dong et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In this paper, we employ Weibo Index as the proxy for investor attention and analyze the relationships between investor attention and stock market performance, i.e., trading volume, return, and volatility. The empirical results firstly show that Weibo attention is positively related to trading volume, intraday volatility, and return. Secondly, there exist bidirectional causal relationships between Weibo attention and stock market performance. Thirdly, we generally find that higher Weibo attention indicates higher correlation coefficients with the quantile regression analysis.

1. Introduction

The development of the Internet offers investors more channels to obtain information, discuss market performance, and communicate their forecasting. Also, the change of information environment has made the information network between investors and markets more complex and harder to analyze. Various websites have their own functions and user structures so that different websites have distinct degrees of influence on investors, e.g., Twitter, Facebook, Google, Baidu Index, and Sina Weibo. As the largest social media website in China, Sina Weibo has 340 million active registered users until the first quarter of 2017. It has exceeded Twitter and become the social platform which has most active registered users in the world. Almost all Chinese listed companies and government agencies have their own official Weibo accounts to publish information and discuss the influence by major policy changes. At the same time, many investors who are regarded as more technical and specialized than most of individual investors also use Sina Weibo to share their opinions and forecast stock market performance. In that sense, the use of Sina Weibo helps individual investors to obtain news and it also can be used to measure the change of investor attention. Therefore, the focus of this paper is on the

relationships between Sina Weibo attention and stock market performance, i.e., trading volume, volatility, and return.

In recent empirical studies, many scholars investigated the relationships between investor attention measured by open-source information and stock market performance [1–3]. Some studies have shown that investor attention measured from Twitter [4, 5], Google [6], Facebook [7], Baidu Index [8, 9], and other channels [2, 10] can be used to analyze the stock performance. In particular, Chen [11] used Google search volume to measure investor attention to analyze global stock markets. Vozlyublennaya [12] used Google search frequency to measure investor attention and found that there was a significant short-term change in index returns following an increased attention but a shock to returns leads to a long-term change in attention. Bank et al. [13] explored the relationship between Google search volume and German stock performance, and the results show that increasing search queries lead to a rise in trading activity and stock liquidity. Zhang et al. [8] used search frequency of stock name in Baidu Index as the proxy variable for investor attention to explain abnormal return as well as trading volume. Shen et al. [9] regarded Baidu news as the proxy for information flow to study the relationship between information flow and return volatility, and the empirical findings

contradicted the prediction of MDH but supported the SIAH. Other scholars used different categories of indirect proxies to measure investor attention. Sicherman et al. [14] used daily investor online account logins as the financial attention to explain the relationship between investors' personal portfolios and how attention affects trading activities. Lou [15] revealed how firm advertising attracts investor attention and influenced short-term stock returns. Fang and Peress [16] studied the relations between media coverage and expected stock returns and found that firms with more media coverage had a higher return and such influence maintains larger for small company. Ben-Rephael et al. [17] found that institutional attention responds more quickly to major news events rather than earnings announcements or analyst recommendation changes through using news searching and news reading activity for specific stocks on Bloomberg terminals and Google search activity to measure abnormal institutional investor attention.

Some scholars studied investor attention in China through different social media channels for information, such as Baidu [18, 19], Guba [20–22], and Sina Weibo [23]. But previous research on Sina Weibo often focuses on specific stock performance, particular time of duration, or account information. However, there is few research using the entire microblogs on Sina Weibo because it is hard to confirm the number of keywords in entire Sina Weibo. This paper is also in line with the abovementioned studies, but we consider Weibo attention through all appearing frequency of key words in entire Sina Weibo to reveal the relation between Weibo attention and stock performance. And we use quantile regression to analyze whether there is a difference between impacts of higher and lower attention from Sina Weibo on the stock market. At the same time, we use the performance of five major indices including Shanghai Stock 50 Index (SH50), CSI 300 Index (CSI 300), Shenzhen Index (SZ), Small and Medium-Sized Enterprise Index (SME), and China Growth Enterprise Market Index (ChiNext) to represent different stock markets rather than only focusing on a single market. We firstly consider the relations between Weibo attention and stock market performance through three contemporaneous correlations containing one linear and two nonlinear methods. In order to discriminate the bidirectional relationships, we use Granger causality test to further investigate the above relationships. Finally, we use quantile regression to analyze how different levels of attention may influence the stock markets. According to above methods, we find that there are positive relations between Weibo attention and trading volume or intraday volatility; however, the coefficients between Weibo attention and returns are different across markets. Moreover, the trading volume of SZ, SME, and ChiNext can Granger-cause Weibo attention but there is no Granger causality in other situations. Through quantile regression analysis, we also find that high attention actually indicates accurate market performance.

This paper is organized as follows. Section 2 describes the data. Section 3 introduces the methodology of empirical analysis. Section 4 performs the contemporaneous correlation, the Granger causality, and the quantile regression test. Section 5 concludes this paper.

2. Data Description

Sina Weibo as the largest Chinese microblogging website has more influence than other social media platforms, and many investors use their Sina Weibo account to share their opinion on specialized stock market performance [23]. So, the high frequency of the keywords appearing in Sina Weibo means that more investors discuss the performance of specialized stock market and they pay more attention on the stock market. So, we regard Weibo Index which represents the number of the keywords appearing in Sina Weibo as the proxy to measure Weibo attention and obtain the data from the official website (<http://www.weizhishu.com/>).

On the other hand, we use returns, trading volume, and intraday volatility to measure stock market performance. We choose Shanghai Stock 50 Index (SH50), CSI 300 Index (CSI 300), Shenzhen Index (SZ), Small and Medium-Sized Enterprise Index (SME), and China Growth Enterprise Market Index (ChiNext) as the keywords and obtain relevant Weibo Index in this paper. The market index data including returns, closing prices, opening prices, the highest prices, the lowest prices, and trading volumes from March 1, 2013 to October 31, 2017 (1137 trading days) are from CSMAR database. We consider range-based volatility including more information, and previous studies have demonstrated that range-based volatility can estimate index fluctuates more effectively than other low-frequency methods in both Chinese and foreign stock markets [24, 25]. So, we define intraday volatility of index as follows [26]:

$$V_{i,t} = \frac{1}{2} H_p L_{p_{i,t}}^2 - (2 \ln 2 - 1) O_p C_{p_{i,t}}^2, \quad (1)$$

where $H_p L_{p_{i,t}}$ is the difference in natural logarithms of the highest and lowest prices for index i on day t and $O_p C_{p_{i,t}}$ is the difference in natural logarithms of the opening and closing prices for index i on day t .

Table 1 reports the statistical property of index returns, volume, volatility, and Weibo attention in this paper, and we also give the results of Jarque-Bera statistic test and Ljung-Box statistic test in this table. And we use natural logarithm to deal with different volumes. From this table, we can obtain that most of the variables fit Gaussian distribution except the returns of SME and ChiNext. Also, the means of returns across different stock markets are almost positive except SME. Besides, the volume and attention of different markets have little difference. And we also find that the intraday volatility has the highest value in kurtosis, and skewness in four variables and the skewness of returns are all negative while others are all positive. We also observe that most of the variables fit Gaussian distribution except returns of SME and ChiNext. At the same time, all variables exist 20th-order serial correlation. Figure 1 illustrates the evolution of the all-trading-day Weibo Index of Shanghai Stock 50 Index (SH50), CSI 300 Index (CSI 300), Shenzhen Index (SZ), Small and Medium-Sized Enterprise Index (SME), and China Growth Enterprise Market Index (ChiNext) from March 1, 2013 to October 31, 2017. There are 1137 trading days, and we can find that Weibo attention of HS300, SME, and

TABLE 1: Statistical properties for the variables.

Variables	Mean	Max	Min	Median	Std.	Kurtosis	Skewness	JB	Q(20)
SH50_returns	0.04	7.84	-9.38	0.00	1.63	9.14	-0.46	110***	1825***
HS300_returns	0.05	6.71	-8.75	0.06	1.57	8.94	-0.87	113***	1814***
SZ_returns	0.03	6.45	-8.24	0.09	1.75	7.15	-0.86	64***	953***
SME_returns	-0.03	6.83	-100	0.16	3.46	614.50	-21.34	7	17801414***
ChiNext_returns	0.00	7.16	-100	0.11	3.67	487.08	-17.92	11	11162412***
SH50_volume	12.75	15.14	11.43	12.55	0.73	3.15	0.97	14755***	180***
HS300_volume	13.94	15.74	12.79	13.83	0.64	2.89	0.73	15774***	101***
SZ_volume	12.93	14.96	10.83	13.35	1.10	1.65	-0.30	20034***	103***
SME_volume	11.93	13.04	10.72	11.91	0.48	2.38	-0.03	14283***	18***
ChiNext_volume	11.42	12.57	9.81	11.44	0.55	2.32	-0.20	15749***	29***
SH50_volatility	2.01×10^{-4}	5.67×10^{-3}	2.68×10^{-6}	7.37×10^{-5}	4.62×10^{-4}	60.60	6.70	2418***	165676***
HS300_volatility	1.76×10^{-4}	4.34×10^{-3}	3.19×10^{-6}	6.64×10^{-5}	3.97×10^{-4}	50.58	6.15	3379***	114424***
SZ_volatility	1.99×10^{-4}	5.18×10^{-3}	1.93×10^{-6}	8.10×10^{-5}	4.25×10^{-4}	56.61	6.40	2337***	143913***
SME_volatility	2.02×10^{-4}	5.67×10^{-3}	0	7.99×10^{-5}	4.45×10^{-4}	66.36	6.89	2398***	199154***
ChiNext_volatility	2.95×10^{-4}	7.83×10^{-3}	0	1.30×10^{-4}	5.66×10^{-4}	56.94	6.17	2443***	145072***
SH50_attention	4.96	8.94	0.00	4.85	1.53	2.68	0.39	9462***	33***
HS300_attention	5.36	9.08	0.00	5.36	0.69	7.03	-0.09	4615***	770***
SZ_attention	2.52	6.83	0.00	2.48	1.11	3.44	0.43	2028***	44***
SME_attention	6.75	11.57	0.00	6.73	0.63	20.22	-0.01	2132***	14051***
ChiNext_attention	8.77	11.68	0.00	8.83	0.69	25.85	-1.93	5281***	25430***

This table reports the statistical properties for returns, volume, volatility, and Weibo attention. JB denotes the Jarque-Bera statistic test with the null hypothesis of Gaussian distribution. Q(20) denotes the Ljung-Box statistic test for up to 20th-order serial correlation. *** indicates significant at 1% level.

ChiNext has smaller fluctuation than SH300 and SZ. We also observe that the peaks and troughs of different evolutions happen in the same period.

3. Empirical Methodology

We firstly analyze the correlation of the evolution of Weibo attention and market variables through Pearson correlation coefficient, Spearman correlation coefficient, and Kendall correlation coefficient. And then, Granger causality test captures the bidirectional relationships between investor attention and stock performance. Finally, we use quantile regression analysis to study the further relationships among different variables.

3.1. The Contemporaneous Correlation. In order to calculate the coefficients between different stock returns, trading volume, intraday volatility, and corresponding Weibo Index, with the consideration of the evolution of Weibo attention and market variables, we, respectively, use Pearson correlation coefficient, Spearman correlation coefficient, and Kendall correlation coefficient from linear to nonlinear aspects to analyze the relations among these variables [27]. We calculate different correlation coefficients as follows:

$$\rho_p = \frac{\text{Cov}(WI, MV)}{\sigma_{WI}\sigma_{MV}}, \quad (2)$$

where ρ_p represents Pearson correlation coefficient. $\text{Cov}(WI, MV)$ represents covariance between Weibo Index and market variables, and σ_{WI} and σ_{MV} are the standard deviations of Weibo Index and market variables.

$$\rho_s = 1 - \frac{6\sum d_i^2}{n(n^2 - 1)}, \quad (3)$$

where ρ_s represents Spearman correlation coefficient. And we calculate d_i through firstly rank Weibo Index and corresponding market variables separately and get the absolute value of the difference of the ranking.

$$\begin{aligned} \rho_k &= \frac{C - D}{\sqrt{(N_3 - N_1)(N_3 - N_2)}}, \\ N_3 &= \frac{1}{2}n(n-1), \\ N_1 &= \sum_1^s \frac{1}{2}U_i(U_i - 1), \\ N_2 &= \sum_i^t \frac{1}{2}V_i(V_i - 1), \end{aligned} \quad (4)$$

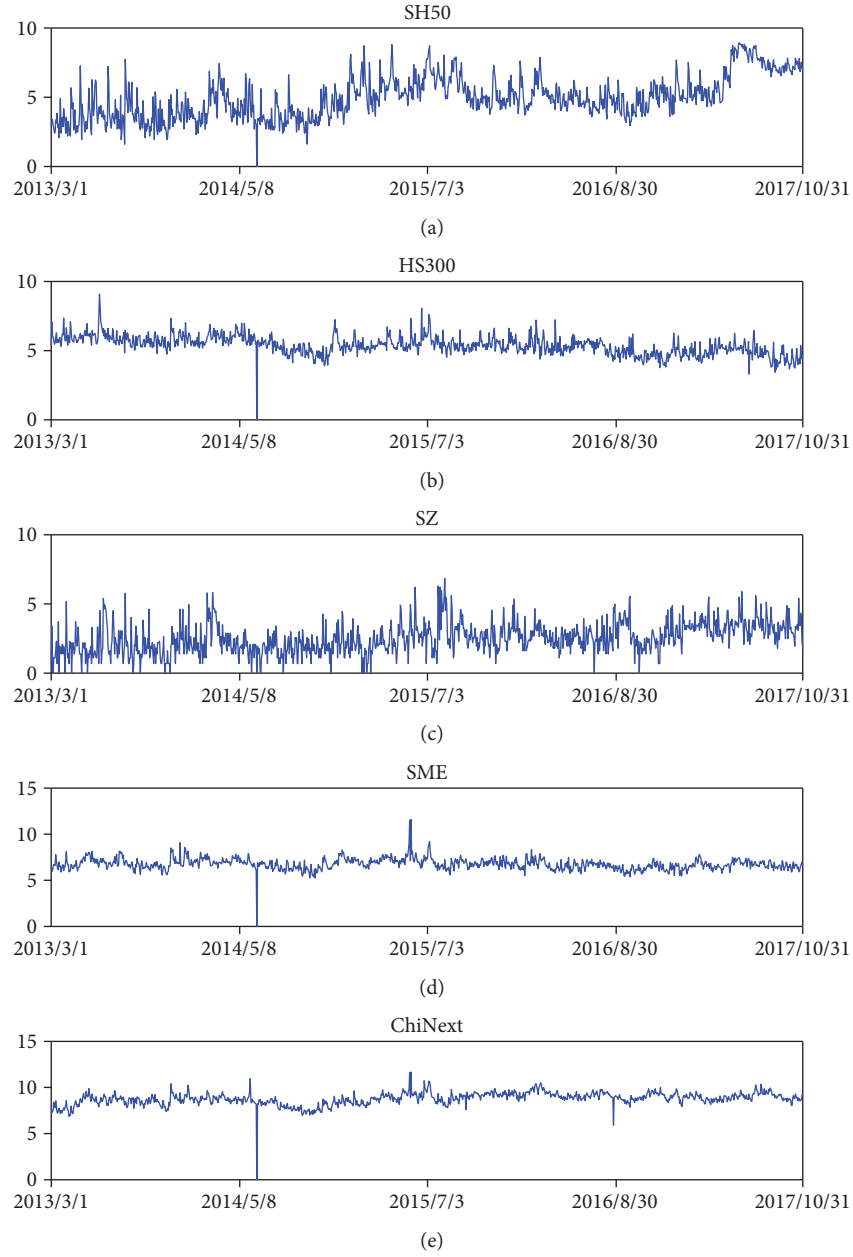


FIGURE 1: The evolution of Weibo attention. This figure shows the evolution of Weibo Index of different stock markets from March 1, 2013 to October 31, 2017. We notice that there are some zero values in the figure. After checking the data, we found that zeros fall into the nontrading days. In the following empirical analysis, these calendar days are removed from the sample.

where ρ_k denotes Kendall correlation. C is the number of consistent couples, and D is the number of inconsistent couples; U_i and V_i , respectively, mean the number of element in the i th set of Weibo Index and market variables. The correlation coefficients are all between $+1$ and -1 , and the negative coefficient indicates the adverse relationship between market variables and Weibo Index.

3.2. Granger Causality. Shen et al. [28] have proved that there exist bidirectional relationships between open information and shock market performance. Zhang et al. [8] also find that there exist bidirectional relationships between investor

attention measured by Baidu Index and stock market performance. So, we also use Granger causality test to analyze the bidirectional relationships of Weibo attention and different market variables. We construct the following regression models to test the Granger causality [26]:

$$\begin{aligned}
 WI_t &= u_{WI} + \sum_{i=1}^p \alpha_i WI_{t-i} + \sum_{j=1}^p \beta_j MV_{t-j} + \varepsilon_{t,WI}, \\
 MV_t &= u_{MV} + \sum_{i=1}^p \alpha_i MV_{t-i} + \sum_{j=1}^p \beta_j WI_{t-j} + \varepsilon_{t,MV},
 \end{aligned} \tag{5}$$

TABLE 2: Correlation coefficients between Weibo Index and market variables.

Index	Pearson	Spearman	Kendall
Panel A: returns			
SH50	0.0365 (0.2188)	0.0705 (0.0174)**	0.0466 (0.0188)**
HS300	-0.1152 (0.0001)***	-0.0584 (0.0491)**	-0.0377 (0.0571)*
SZ	-0.0902 (0.0023)***	-0.0379 (0.2017)	-0.0244 (0.2253)
SME	0.0038 (0.8970)	0.0748 (0.0117)**	0.0518 (0.0089)***
ChiNext	-0.0865 (0.0035)***	-0.0636 (0.0319)**	-0.0433 (0.0287)**
Panel B: trading volume			
SH50	0.2980 (0.0000)***	0.3661 (0.0000)***	0.2441 (0.0000)***
HS300	0.0755 (0.0109)**	-0.1971 (0.0000)***	-0.1212 (0.0000)***
SZ	0.3304 (0.0000)***	0.3955 (0.0000)***	0.2657 (0.0000)***
SME	0.2394 (0.0000)***	0.1787 (0.0000)***	0.1223 (0.0000)***
ChiNext	0.4732 (0.0000)***	0.5373 (0.0000)***	0.3686 (0.0000)***
Panel C: volatility			
SH50	0.2056 (0.0000)***	0.1416 (0.0000)***	0.0969 (0.0000)***
HS300	0.2871 (0.0000)***	0.3993 (0.0000)***	0.2716 (0.0000)***
SZ	0.0972 (0.0010)***	-0.0258 (0.3844)	-0.0160 (0.4271)
SME	0.1698 (0.0000)***	0.3093 (0.0000)***	0.2082 (0.0000)***
ChiNext	0.2071 (0.0000)***	0.2270 (0.0000)***	0.1531 (0.0000)***

This table reports different correlation coefficients between returns, trading volume, and volatility of SH50, HS300, SZ, SME, and ChiNext and Weibo attention from March 1, 2013 to October 31, 2017. *** indicates significant at 1% level; ** indicates significant at 5% level; * indicates significant at 10% level.

where p means value range, WI_t and MV_t denote the value of Weibo Index and market variables at corresponding time, α and β denote the coefficient, u_{WI} and u_{MV} denote the intercept term, and $\varepsilon_{t,WI}$ and $\varepsilon_{t,MV}$ denote regression error.

3.3. Quantile Regression Analysis. Aouadi et al. [29] show that higher attention measured by Google search volume in France decreases stock liquidity and increases volatility. And quantile regression analysis can reflect how different distributions of independent variables influence dependent variables. So, we use quantile to analyze whether different levels of Weibo attention influence the market variables.

As for a continuous random variable y , the probability of y which is equal or lesser than $y(\tau)$ is τ and we call the τ quantile is $y(\tau)$ according to Koenker and Bassett [30]. We can express it as follows:

$$\tau = P(y \leq y(\tau)) = F(y(\tau)), \quad (6)$$

where $F(y(\tau))$ is the cumulative distribution function of y . And we also have the following:

$$y(\tau) = F^{-1}(y(\tau)), \quad (7)$$

and it means the portion of y which is less than τ is $y(\tau)$. And we define check function as follows:

$$\rho_\tau(u) = \tau u I(u \geq 0) + (\tau - 1)u I(u < 0). \quad (8)$$

According to the equation, if we define u as $y - \xi$, we can get the following equation:

$$\rho_\tau(y - \xi) = \tau(y - \xi)I(y - \xi \geq 0) + (\tau - 1)(y - \xi)I(y - \xi < 0). \quad (9)$$

The quantile regression of y is to find ξ to minimum $E[\rho_\tau(y - \xi)]$.

4. Empirical Results

This section presents our results of relations between Weibo attention and stock index performances. We calculate contemporaneous correlation in Section 4.1, perform the Granger causality test in Section 4.2, and provide quantile regression analysis in Section 4.3.

4.1. The Contemporaneous Correlation. We use Pearson correlation coefficient, Spearman correlation coefficient, and Kendall correlation coefficient to study the relationship between Weibo attention and stock performance.

Table 2 reports the correlation coefficients between Weibo Index and market variables. As for the correlation coefficients between stock returns and Weibo Index, we can see that SH50 and SME have positive coefficients though they are insignificant by Pearson correlation coefficient, while other three indices are negative. The correlation coefficients of all markets are significant by Spearman correlation coefficient and Kendall correlation coefficient except SZ index. The trading volume and Weibo Index are significant by three

TABLE 3: Granger causality between Weibo Index and market variables.

X	Y	X Granger cause Y	Y Granger cause X
Panel A: returns			
Weibo Index	SH50	2.0340 (2.7100)	0.7788 (2.7100)
Weibo Index	HS300	0.9145 (2.7100)	6.6441 (2.7100)**
Weibo Index	SZ	2.7986 (2.7100)*	8.4985 (2.7100)***
Weibo Index	SME	1.1694 (2.7100)	0.1129 (2.7100)
Weibo Index	ChiNext	2.5922 (2.7100)	0.8979 (2.7100)
Panel B: trading volume			
Weibo Index	SH50	0.0972 (2.7100)	2.6370 (2.7100)
Weibo Index	HS300	0.7149 (2.7100)	0.5240 (2.7100)
Weibo Index	SZ	0.0154 (2.7100)	14.3082 (2.7100)***
Weibo Index	SME	2.4014 (2.7100)	3.4286 (2.7100)**
Weibo Index	ChiNext	0.6812 (2.7100)	13.4979 (2.7100)***
Panel C: volatility			
Weibo Index	SH50	0.6743 (2.7100)	1.2457 (2.7100)
Weibo Index	HS300	8.4079 (2.7100)***	1.8290 (2.7100)
Weibo Index	SZ	0.4745 (2.7100)	0.9255 (2.7100)
Weibo Index	SME	1.1209 (2.7100)	6.4183 (2.7100)**
Weibo Index	ChiNext	2.8049 (2.7100)*	1.6762 (2.7100)

This table reports the results for the Granger causality analysis between returns, trading volume, and volatility of SH50, HS300, SZ, SME, and ChiNext and Weibo attention. The X Granger cause Y means Weibo attention can Granger-cause the changes of market variables and Y Granger cause X means markets can Granger-cause the changes of Weibo attention. *** indicates significant at 1% level; ** indicates significant at 5% level; * indicates significant at 10% level.

kinds of correlation coefficient, and the coefficients are all positive except CSI300 with Spearman correlation coefficient and Kendall correlation coefficient. As far as intraday volatility of each index, the results suggest that all coefficients are positive and significant at 1% except for the SZ index calculated by Spearman regression and Kendall regression.

Above results show the relationship between Weibo attention and stock market performance. We can find 60% coefficients of Weibo attention and returns are adverse. But Weibo Index and trading volume or intraday volatility of index are nearly positive except for the coefficient between Weibo Index and trading volume of CSI300 through two nonlinear regressions. The results show that the correlations of Weibo attention and stock market trading volume or intraday volatility are more obvious than return, and Weibo attention has positive influence on them. It indicates that high volume of stock market will attract more investor attention and investors will also discuss more subsequent market performance in Sina Weibo. Although investors aim to make profit, the stock market returns have no obvious effect on investor attention.

4.2. Granger Causality. In order to analyze the bilateral relation between Weibo attention and stock variables, we set Granger causality test and Table 3 shows the results through above models.

The return of SZ can Granger-cause Weibo Index, and Weibo Index can also Granger-cause the return of SZ. However, except the return of HS300 can Granger-cause Weibo attention at 5% level, no Granger causality exists in Weibo Index and returns of other markets. The trading volume can Granger-cause Weibo Index in SZ, SME, and ChiNext, and no Weibo Index can Granger-cause the changes of trading volume. In terms of intraday volatility of index, the Weibo Index can Granger-cause intraday volatility of HS300 or ChiNext and SME can Granger-cause Weibo Index while no Granger causality in others. The empirical results suggest that trading volume has more influence on investor attention. When the trading volume increases, investors will discuss more about stock in Sina Weibo to exchange ideas. And the change of returns in HS300 and SZ can also lead to more discussions.

4.3. Quantile Regression Analysis. We use quantile regression analysis at 0.05, 0.2, 0.6, 0.8, and 0.95 to consider the influence on Weibo attention and distribution of different market variables in order to analyze the relationships between different levels of attention and stock market performance.

Table 4 shows the results for the quantile regression analysis. We can find that at 0.95 quantile, the coefficients between stock returns and Weibo Index are the highest and significant at 1% level. We also find that higher quantile will lead higher coefficient. But as for the trading volume, the highest coefficient happens at different quantiles. The highest coefficient of SH50 and SME happens at 0.8 quantile, and HS300, SZ, and ChiNext happen, respectively, at 0.95 quantile, 0.4 quantile, and 0.6 quantile. With regard to intraday volatility of index, we can find that most of the indices have the highest relationship at 0.95 quantile except SME.

From the above results, we observe that the highest Weibo attention always means the highest coefficients between market returns and intraday volatility. That means in this case where stock market is discussed frequently, the relations between the market return and intraday volatility and investors are larger. But the highest Weibo attention does not always mean the highest trading volume, and we consider different markets have different quantiles because of the different stock market structures so that Weibo investors have different opinions on the change of different stock markets.

5. Conclusions

This paper employs the Weibo Index as the proxy for investor attention and uses the return, trading volume, and intraday volatility of SH50, HS300, SZ, SME, as well as ChiNext to represent different stock market performances. We investigate the relations between Weibo attention and stock market performance. We firstly find that the statistical property of Weibo Index and the coefficient of Weibo Index and trading volume or intraday volatility are positive regardless of linear regression or nonlinear regression except HS300 while the relations between returns and Weibo Index are 60% adverse. Secondly, we use Granger causality test to analyze the bilateral relation between Weibo attention and market stock performance. The results show that trading volume

TABLE 4: Results for the quantile regression analysis.

Indices	Quantile regression					
	0.05	0.2	0.4	0.6	0.8	0.95
Panel A: returns						
SH50	-0.3718***	0.0252	0.0328	0.0752***	0.1059***	0.3492***
HS300	-1.2600***	-0.4090***	-0.1226**	-0.0668	0.2057***	0.9803***
SZ	-0.9174***	-0.1622***	-0.0354	-0.0121	-0.0825	0.3389***
SME	-0.8412	-0.2953**	0.0198	0.2887***	0.5507***	0.9357***
ChiNext	-1.5149***	-0.5379***	-0.2338**	-0.0720	0.2259	0.7316**
Panel B: trading volume						
SH50	0.0685***	0.0727***	0.0902***	0.1532***	0.3290***	0.3052***
HS300	-0.2416***	-0.2277***	-0.1841***	-0.0657***	0.2200***	0.3326***
SZ	0.1505***	0.4418***	0.5550***	0.3858***	0.1683***	0.0799***
SME	0.0107***	0.0383***	0.1076***	0.1982***	0.2855***	0.1804***
ChiNext	0.3937***	0.4471***	0.4436***	0.4705***	0.4288***	0.2215***
Panel C: volatility						
SH50	0.0000	0.0002	0.0001	0.0013***	0.0052***	0.0230***
HS300	0.0008***	0.0016***	0.0031***	0.0061***	0.0122***	0.0393***
SZ	-0.0002	0.0005**	0.0005	0.0010**	0.0045***	0.0163***
SME	0.0008***	0.0018***	0.0040***	0.0067***	0.0136***	0.0143
ChiNext	0.0005	0.0009	0.0035***	0.0079***	0.0189***	0.0392**

*** indicates significant at 1% level; ** indicates significant at 5% level.

can Granger-cause Weibo attention for 3 out of 5 but no Granger causality exists on return and intraday volatility. Thirdly, the results of quantile regression show that higher Weibo attention always means higher coefficient of Weibo attention and market performance, especially for market returns and intraday volatility.

These findings demonstrate that there exists a relation between Weibo attention and stock market performance. Therefore, investors can pay attention to Weibo attention to analyze the change of stock market and adjust the proportion of stocks from different stock market. However, in this paper, we do not find the underlying mechanisms behind those phenomena. We attempt to explain the reason for the phenomena from the perspectives of different investor structures and the users from different websites in further work.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this article.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (71532009 and 71701150), the Young Elite Scientists Sponsorship Program by Tianjin (TJSQNTJ-2017-09), and the Fundamental Research Funds for the Central Universities (63182064).

References

- [1] A. B. Abel, J. C. Eberly, and S. Panageas, "Optimal inattention to the stock market," *American Economic Review*, vol. 97, no. 2, pp. 244–249, 2007.
- [2] D. Andrei and M. Hasler, "Investor attention and stock market volatility," *The Review of Financial Studies*, vol. 28, no. 1, pp. 33–72, 2015.
- [3] D. Savage, X. Zhang, X. Yu, P. Chou, and Q. Wang, "Anomaly detection in online social networks," *Social Networks*, vol. 39, pp. 62–70, 2014.
- [4] Y. R. Lin, B. Keegan, D. Margolin, and D. Lazer, "Rising tides or rising stars?: dynamics of shared attention on twitter during media events," *PLoS One*, vol. 9, no. 5, article e94093, 2014.
- [5] S. Mohd Shariff, X. Zhang, and M. Sanderson, "User perception of information credibility of news on Twitter," in *Advances in Information Retrieval*, pp. 513–518, Springer, 2014.
- [6] A. Siganos, "Google attention and target price run ups," *International Review of Financial Analysis*, vol. 29, no. 5, pp. 219–226, 2013.
- [7] A. Siganos, E. Vagenas-Nanos, and P. Verwijmeren, "Facebook's daily sentiment and international stock markets," *Journal of Economic Behavior & Organization*, vol. 107, pp. 730–743, 2014.
- [8] W. Zhang, D. Shen, Y. Zhang, and X. Xiong, "Open source information, investor attention, and asset pricing," *Economic Modelling*, vol. 33, pp. 613–619, 2013.
- [9] D. Shen, X. Li, and W. Zhang, "Baidu news information flow and return volatility: evidence for the sequential information arrival hypothesis," *Economic Modelling*, vol. 69, pp. 127–133, 2018.

- [10] Y. Zhang, Z. Zhang, L. Liu, and D. Shen, "The interaction of financial news between mass media and new media: evidence from news on Chinese stock market," *Physica A: Statistical Mechanics and its Applications*, vol. 486, pp. 535–541, 2017.
- [11] T. Chen, "Investor attention and global stock returns," *Journal of Behavioral Finance*, vol. 18, no. 3, pp. 358–372, 2017.
- [12] N. Vozlyublennaia, "Investor attention, index performance, and return predictability," *Journal of Banking & Finance*, vol. 41, pp. 17–35, 2014.
- [13] M. Bank, M. Larch, and G. Peter, "Google search volume and its influence on liquidity and returns of German stocks," *Financial Markets and Portfolio Management*, vol. 25, no. 3, pp. 239–264, 2011.
- [14] N. Sichernman, G. Loewenstein, D. J. Seppi, and S. P. Utkus, "Financial attention," *The Review of Financial Studies*, vol. 29, no. 4, pp. 863–897, 2016.
- [15] D. Lou, "Attracting investor attention through advertising," *The Review of Financial Studies*, vol. 27, no. 6, pp. 1797–1829, 2014.
- [16] L. Fang and J. Peress, "Media coverage and the cross-section of stock returns," *Journal of Finance*, vol. 64, no. 5, pp. 2023–2052, 2009.
- [17] A. Ben-Rephael, Z. Da, and R. D. Israelsen, "It depends on where you search: institutional investor attention and under-reaction to news," *The Review of Financial Studies*, vol. 30, no. 9, pp. 3009–3047, 2017.
- [18] D. Shen, Y. Zhang, X. Xiong, and W. Zhang, "Baidu index and predictability of Chinese stock returns," *Financial Innovation*, vol. 3, no. 1, p. 4, 2017.
- [19] J. Chen, Y. J. Liu, L. Lu, and Y. Tang, "Investor attention and macroeconomic news announcements: evidence from stock index futures," *Journal of Futures Markets*, vol. 36, no. 3, pp. 240–266, 2016.
- [20] Y. Huang, H. Qiu, and Z. Wu, "Local bias in investor attention: evidence from China's internet stock message boards," *Journal of Empirical Finance*, vol. 38, pp. 338–354, 2016.
- [21] X. Li, D. Shen, and W. Zhang, "Do Chinese internet stock message boards convey firm-specific information?," *Pacific-Basin Finance Journal*, vol. 49, pp. 1–14, 2018.
- [22] B. Zhang and Y. Wang, "Limited attention of individual investors and stock performance: evidence from the ChiNext market," *Economic Modelling*, vol. 50, pp. 94–104, 2015.
- [23] D. Shen, X. Li, M. Xue, and W. Zhang, "Does microblogging convey firm-specific information? Evidence from China," *Physica A: Statistical Mechanics and its Applications*, vol. 482, pp. 621–626, 2017.
- [24] M. Parkinson, "The extreme value method for estimating the variance of the rate of return," *Journal of Business*, vol. 53, no. 1, pp. 61–65, 1980.
- [25] B. X. Sun and M. J. Wang, "A new class GARCH model based on price range," *Journal of Applied Statistics and Management*, vol. 32, no. 2, pp. 259–267, 2013.
- [26] C. W. J. Granger, "Investigating causal relations by econometric models and cross-spectral methods," *Econometrica*, vol. 37, no. 3, pp. 424–438, 1969.
- [27] W. Zhang, X. Li, D. Shen, and A. Tegli, "Daily happiness and stock returns: some international evidence," *Physica A: Statistical Mechanics and its Applications*, vol. 460, pp. 201–209, 2016.
- [28] D. Shen, W. Zhang, X. Xiong, X. Li, and Y. Zhang, "Trading and non-trading period internet information flow and intraday return volatility," *Physica A: Statistical Mechanics and its Applications*, vol. 451, pp. 519–524, 2016.
- [29] A. Aouadi, M. Arouri, and F. Teulon, "Investor attention and stock market activity: evidence from France," *Economic Modelling*, vol. 35, no. 3, pp. 674–681, 2013.
- [30] R. Koenker and G. Bassett Jr., "Regression quantiles," *Econometrica*, vol. 46, no. 1, pp. 33–50, 1978.

Research Article

A Trip Purpose-Based Data-Driven Alighting Station Choice Model Using Transit Smart Card Data

Kai Lu ¹, Alireza Khani,² and Baoming Han ¹

¹*School of Traffic and Transportation, Beijing Jiaotong University, Beijing 100044, China*

²*Department of Civil, Environmental and Geo-Engineering, University of Minnesota, Minneapolis, MN 55455, USA*

Correspondence should be addressed to Kai Lu; lukai_bjtu@163.com and Baoming Han; bmhan@bjtu.edu.cn

Received 18 December 2017; Revised 2 June 2018; Accepted 15 July 2018; Published 28 August 2018

Academic Editor: Shuliang Wang

Copyright © 2018 Kai Lu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Automatic fare collection (AFC) systems have been widely used all around the world which record rich data resources for researchers mining the passenger behavior and operation estimation. However, most transit systems are open systems for which only boarding information is recorded but the alighting information is missing. Because of the lack of trip information, validation of utility functions for passenger choices is difficult. To fill the research gaps, this study uses the AFC data from Beijing metro, which is a closed system and records both boarding information and alighting information. To estimate a more reasonable utility function for choice modeling, the study uses the trip chaining method to infer the actual destination of the trip. Based on the land use and passenger flow pattern, applying k-means clustering method, stations are classified into 7 categories. A trip purpose labelling process was proposed considering the station category, trip time, trip sequence, and alighting station frequency during five weekdays. We apply multinomial logit models as well as mixed logit models with independent and correlated normally distributed random coefficients to infer passengers' preferences for ticket fare, walking time, and in-vehicle time towards their alighting station choice based on different trip purposes. The results find that time is a combined key factor while the ticket price based on distance is not significant. The estimated alighting stations are validated with real choices from a separate sample to illustrate the accuracy of the station choice models.

1. Introduction

In the late 1990s, smartcard payment systems were installed in some big cities, and after more than twenty years of development, more than one hundred cities over five continents have adopted smartcard payment systems [1]. This technology has become pivotal to ticket fare collection for public transit for both bus and metro. Since its inception, the transit smart card system produced a large amount of very detailed data on on-board transactions [2]. The smart data system contains many aspects such as the hardware technology (radiofrequency identification (RFID), electromagnetic shield), system construction, and data storage [3, 4]. Meanwhile, with the rich data source data collected by smart card, a lot of researchers are interested in the applications of those data. Generally, the application can be classified into three levels: strategic level, tactical level, and operational level [5]. For the strategic level, the large amount of data from smart

card gives an opportunity for tracking and analyzing long-term individual travel behaviors in both spatial and temporal dimensions. The valuable historical data are fundamental data input for short-term or long-term transit network planning [6]. At the same time, tracking the starting and ending date for each user could obtain the life span of each transit user, which is the supplemental input for network planning [7]. Tactical level is the research related to the strategies that are trying to improve the efficiency, benefits, and energy consumption of the transit system [8]. Operational level is the most popular topic in data application. Generally, there are two branches in this research: passenger behavior analysis and service adjustments. We believe that based on travel information recorded by smart card data, the passenger behavior such as route choice and transfer station choice during their journey in the transit network can be deduced [9–11]. In order to provide better service for the passengers and save their travel time, the timetables are rescheduled

based on the variable passenger demand [12]. Meanwhile, operation agencies could estimate and evaluate the transit service performance by operational statistics such as bus run time, vehicle-kilometers, and person-kilometers [13–16].

In addition to the closed travel information loop for each transit user, in some transit systems, passengers are required to tap the card only while they enter the vehicle, which provide only boarding information [17]. In these systems, the one of the boarding or alighting information is missing. Thanks to the automated vehicle location (AVL), automated data collection (ADC), and other support data resource, merging various transit datasets makes it possible to complete the travel route information. For the past 10 years, researchers worked on finding the closed information for each individual trip. Table 1 summarizes the literatures on seeking missing information in open systems including the methodologies, pros, and future research. In Table 1, AFC is short for automatic fare collection. ADCS is short for automated data collection systems, and AVL is short for automatic vehicle location.

Trip chaining methodology is the typical methodology in these research. Here are two basic assumptions: (1) A high percentage of riders return to the destination station of their previous trip to begin their next trip, and (2) a high percentage of riders end their last trip of the day at the station where they began their first trip of the day. In addition to applying the basic assumptions, for each cardholder, there should be more than one trip in the system. Otherwise, it is impossible to infer the alighting station. For some passengers such as commuters, multiday travel information is recorded. The single trip destination could be inferred based on records from other days. If there is only a one-day trip for the cardholder and contains only one trip, the alighting station is invalid.

For passengers, when choosing the alighting station, they consider the in-vehicle time, transfer time, walking time, and ticket fare comprehensively and choose the station which has the highest utility. Sometimes, the alighting station differs based on different trip purposes because the time value could vary for different purposes. To formulate this optimization model, it is necessary to validate the weight and the coefficient for those impact parameters. Because of the missing information and lack of closed trip data, the validation of those models is seldom discussed. The early attempt to validation and sensitivity analysis is based on the on-board survey data to illustrate the feasibility of the method. However, the on-board survey is expensive and data samples are limited.

The Beijing metro system is a closed system, which contains both boarding and alighting information. With walking time, in-vehicle time, and ticket fare for each candidate alighting a station in a buffer walking time for each trip and the real alighting station from AFC data, the coefficient of each utility factor is estimated. Inspired by Tavassoli et al. [28], we relaxed the alighting station information in AFC data from the Beijing metro system to estimate the alighting station for the different trip purposes to see what choice model could illustrate passenger behavior based on different trip purposes. The choice model calibration results for the

different trips could be used for passenger behavior analysis, network planning, and policy applications.

This paper is organized as follows. In the following section, it describes the data and data preparation process. In the next section, the method for determining trip purposes, trip origins, and trip destinations is presented. In the methodology section, a multinomial logit model and mixed logit models with independent and correlated normally distributed random coefficients are proposed. We used the AFC data to calibrate the parameters in different models in the first and second parts of the empirical study. In the last part of the empirical study, a separate sample of AFC records is used to illustrate the model's accuracy and validity. Conclusions and directions for future work are presented in the last section.

2. Data: Beijing Metro Transit

2.1. Data Description. The data used in this paper are obtained from a metro transit in Beijing, China, and were excerpted from one week of data, in December 2016. At that time, there were 17 lines serving more than 10 million passengers every day with more than 8000 train services. The majority of line headways ranged from 2 to 5 min, and in the peak hour, the headway could reach 90 s. There are two kinds of payment in Beijing metro, a Yikatong card, which can be charged and used for several times, and one trip pass. The proportion of the Yikatong cardholder among all transit passengers is roughly 80%, and only the Yikatong card data can be recorded in the AFC system. In this research, the AFC data, station geometry data, and timetable data are required, and Table 2 represents the data recorded in the dataset.

The AFC dataset contains the entry and exit information for each passenger. One record represents a trip for a passenger. For example, a passenger started his trip from Xizhimen Station at 8:00 AM and alighted at Dongzhimen Station at 8:30 AM. Every station has a unique station ID and station location. For a normal station, the route ID saved only one route. For a transfer station, it serves more than one route, so the route ID contains more than one route. For example, Xizhimen Station is a transfer station for route 13, route 2, and route 4. This station only has one unique station ID, station name, and station location in the dataset. The 3 routes are saved in the route ID. The timetable dataset recorded the train arrival and departure time at each stop for each route. The passenger in-vehicle time could be inferred. In Beijing, the ticket price is based on the shortest travel distance and does not take route into consideration. For example, one passenger started his trip from Xizhimen Station to Dongzhimen Station; regardless of whether he takes route 13 or route 2, the ticket price is the same.

In the database discussed above, the AFC data provide the sample for the empirical study. Walking distances were calculated as the Euclidean distance, and the timetable was used to calculate the travel time between stations using the shortest path.

TABLE 1: Review of studies on estimating alighting stop in a tap-in transit system.

Author	Data	Assumption and constraints	Analysis/use methodology	Application	Pros	Limitations
Barry et al. (2002) [18]	AFC	Two basic assumptions	Trip chaining	New York	Easy to apply	Lack of one trip estimation
Zhao et al. [19] (2007) and Zhao (2004) [20]	ADC	Walking distance threshold	Database management systems	Chicago	(i) Integrating the AFC and AVL (ii) examining the spatial connection	The model was just focused on the bus and rail station
Trépanier et al. (2007) [21]	AFC	Walking tolerance is 2 km.	Transportation object-oriented modeling with vanishing route set	Gatineau	The model is quite suitable for regular transit users	Some passenger information such as single ticket user is missing.
Chu and Chapleau (2008) [22]	AFC	5 min temporal leeway for uncertainty	The linear interpolation and extrapolation to infer the vehicle position	Société de transport de l'Outaouais	Avoids the overestimation of the transfer.	Improves the results of trip purpose and destination inference.
Nassir et al. (2011) [17]	ADCS AFC AVL	Geographical and temporal check Transfer time threshold	OD estimation algorithm	Minneapolis-Saint Paul	Relative relaxation of the search in finding the boarding stops.	The transfer time threshold is fixed
Wang et al. (2011) [23]	ADCS AVL	Walking tolerance is 1 km or 12 min.	Trip chaining methodology based on next trip is bus or rail	London	Validates the automatic inference results against large-scale survey results	Linking system usage to home addresses; access behavior could be better understood
Munizaga and Palma (2012) [24] and Munizaga et al. (2014) [25]	AFC GPS AVL	Generalized time	Position-time alighting estimate model	Santiago	(i) Uses generalized time rather than physical distance (ii) Replaces larger on-board survey	The one trip per card destination estimation is missing.
Gordon et al. (2013) [26]	AFC AVL	Walking tolerance is 1 km and max. transfer is 30 min.	Four-step trip chaining algorithm	London	The circuitry ratios to decide the potential destination for previous journey.	Not all of the passengers alight from the stops closest to the next journey.
Alsger et al. (2015) [27]	AFC	The dynamic transfer time threshold	OD estimation algorithm	Queensland	Transfer time threshold could be increased.	Extended to compare other estimation methods.

TABLE 2: Description of each dataset.

Dataset	Description
<i>AFC data</i>	
Card ID	Unique number that could be taken as the passenger ID
O station	Boarding station ID
Entry time	Access time to the station
D station	Alighting station ID
Exit time	Exit time from the station
<i>Station geographical data</i>	
Station ID	Unique station number
Station name	Name of metro station
Station latitude	Latitude of metro station
Station longitude	Longitude of metro station
Station route ID	Route number which serves at metro station
<i>Timetable data</i>	
Service ID	Given number to every trip
Arrival time	Scheduled arrival time
Departure time	Scheduled departure time
Station ID	Given station number
Route ID	Given route number
<i>Ticket fare data</i>	
O station	Entry station ID
D station	Exit station ID
Ticket price	The price for a specific OD pair.

2.2. Data Cleaning and Preparation. It has been highlighted that the level of accuracy of AFC data may vary and the data can be affected by various types of errors. These errors may affect the accuracy of individual journeys and passenger behavior analysis. In the original AFC data, some errors are caused by system failure or passenger error. The data were filtered with some transactions excluded, such as reloaded transactions, transactions with missing information such as no boarding or alighting stops, and transactions with the same entry and exit stations.

As the study uses the trip chaining method to infer the actual destinations and potential purpose, we exclude single trip cardholders due to lack of information. Figure 1 shows the preparation process. With this data process, the destination of every trip leg of each cardholder has been saved in an individual alighting station list which will be used for the trip purpose inference.

3. Methodology

3.1. Assumptions

3.1.1. Trip Purpose for Each Passenger. Trip purpose could be inferred from their alighting and boarding station. For example, if the passenger started his trip at a residential area and

went to CBD, we could say that this trip is a work trip. Based on the land use and the daily entry flow pattern for each metro station, we processed the k -means clustering method [29, 30] and classified the stations into 7 categories, and the typical stations are marked in Figure 2.

(1) Working stations (red)

Those stations are usually in the CBD area or near the software plaza. In the morning, commuters take transit to go to work and go back home in the early evening. The morning exit passengers are much larger than that in the afternoon. The entry passenger volume in the early evening or late afternoon is much more than that in the morning. The typical stations such as Guomao Station and Zhongguancun Station are marked in red in Figure 2.

(2) Residential stations (orange)

Beijing has 6 ring roads in the city. The house price is unusually high within the 3rd ring. In order to save living expenses, a lot of citizens go to the 6th or even further place to buy or rent a house. There are some huge residential zones in Beijing such as Huilongguan, Huoying. The passenger flow pattern is the opposite. The morning incoming flow is much larger than that in the afternoon, and most passengers exit at these stations in the afternoon. The typical stations such as Huilongguan Station and Tiantongyuan Station are marked in orange in Figure 2.

(3) Working-residential stations (yellow)

Although the house price is pretty high, comparing with the travel time, some commuters prefer to rent or buy a house in the downtown area. The land use is more like the mix of CBD and the residential place such as the university campus area. The passenger flow patterns of these stations keep stable, and they do not have a flow peak during the day. The typical stations such as Wukesong Station and Gongzhufen Station are marked in yellow in Figure 2.

(4) Transit hub stations (green)

The in-coming and out-coming passenger flows, whether in the morning peak hour or in the afternoon peak hour, are always large in the transit hub. Mostly, they are the key points of the transit line such as transfer stations. The typical stations such as Dongzhimen Station, Xizhimen Station, and Songjiazhuang Station are marked in green in Figure 2.

(5) Railway stations (light blue)

Based on the land use, the railway station is a very independent station category. The in-coming and out-coming flow highly depends on the railway schedule. We have 3 railway stations in Beijing. They are Beijing railway station, Beijing south

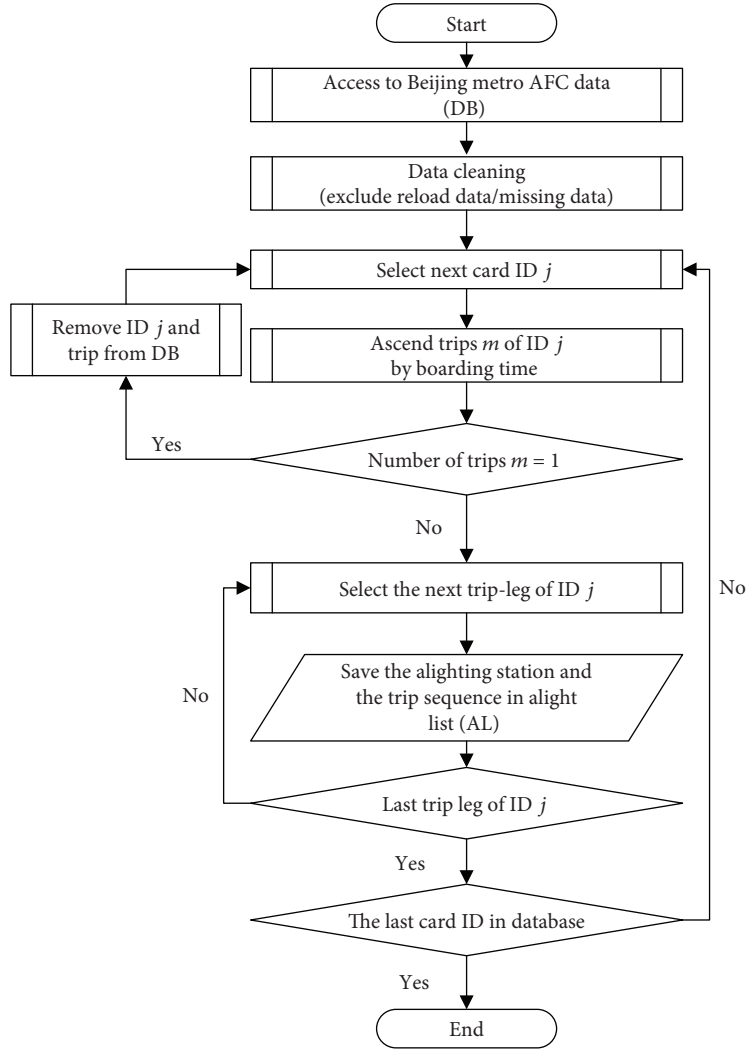


FIGURE 1: Data preparation process.

railway station, and Beijing west railway station, which are marked in blue in Figure 2.

(6) Shopping-sightseeing stations (deep blue)

There are some sightseeing and shopping sites such as The Forbidden City and Tiananmen Square, which attract a lot of tourists and visitors every day. For these stations, the total daily passenger volume during the weekends and holidays is usually higher than during workdays. The typical stations for this category, such as Tiananmen East, Tiananmen West, and Xidan stations, are marked in deep blue in Figure 2.

(7) Rural stations (purple)

The Beijing network is a huge network, and the operation distance has reached 608 km. Some rural areas also have operation lines for passengers such as Changling Line and Fangshan Line. The daily average passenger flow is much smaller in the rural

lines compared with the volume in the downtown area. The typical rural stations are marked in purple in Figure 2.

For each trip, the trip purpose could be estimated based on the station category. For example, a passenger started his trip from a residential station and finished his trip at a working station. Based on the station category, we could label this trip as a working trip. This process could efficiently determine the trip purpose during the day.

However, there is a category that the station could be a workplace or a residential place. In order to determine the trip purpose for these trips, we performed a filter process. For each passenger in Beijing AFC data, the alighting station and boarding time are recorded according to the alighting station list for a passenger during a week. If the alighting station frequency is more than three times on weekdays, we make an assumption that the passenger is a commuter in the city and this place is a workplace or a home [31]. Considering the trip sequence and boarding time for a serial

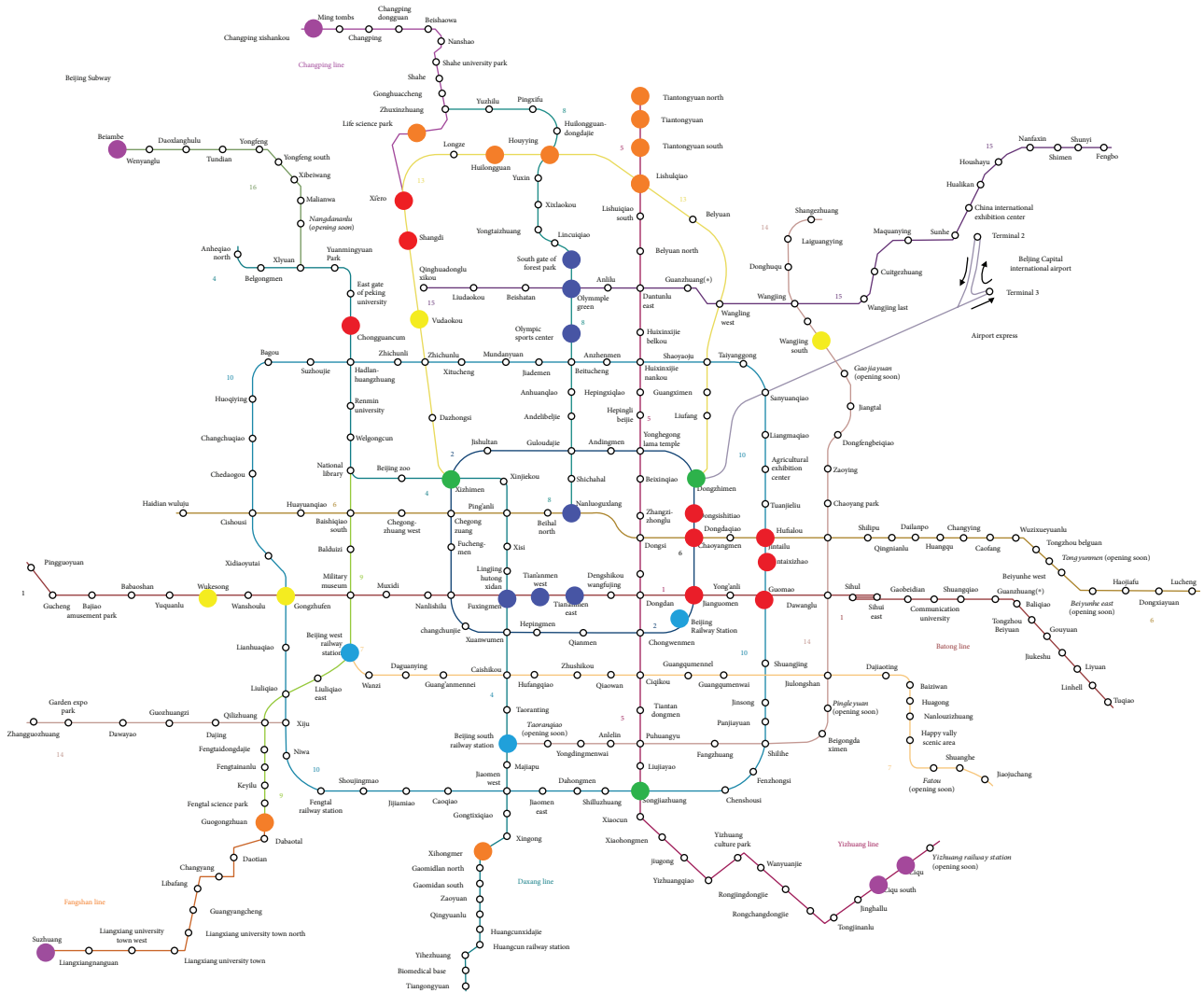


FIGURE 2: The typical stations for each category in Beijing metro.

number, if the trip happened at the early times of the day and the sequence number is one, we label this trip as a home trip. If the trip occurred later in the day or it is the last trip of the day, we label this trip as a work trip. Figure 3 shows the trip purpose labelling process.

3.1.2. Intelligent Passenger. Although the boarding and alighting information is recorded in the AFC data, the passenger trip routes are not recorded. In our study, we assume every passenger is an intelligent agent and wants to minimize the travel cost and maximize the utility of the travel. As such, the passenger will choose the shortest path from the boarding station to the alighting station. We calculate and use the shortest path travel time as in-vehicle. Also, we assume that a passenger will not detour when they go to another station by foot, so we take the Euclidean distance between the two stations as the walking distance.

3.1.3. The Actual Destination of the Trip and Walking Buffer Circle. AFC data recorded the alighting station, but the actual

destination is missing. We assume that the passenger is a smart decision-maker, so he/she would choose an alighting station which is closer to the actual destination. In this study, we assume that the actual destination is somewhere in between the two consecutive stations, the alighting station of the previous trip, and the next trip's origin, as seen in Figure 4(a). However, if the distance between the two consecutive stations is more than a walking threshold (we use 3km in the later empirical study), shown in Figure 4(b), the passenger is more likely to take other modes of transportation. In this way, the actual destination of the first trip is hard to infer, so we would exclude this trip from the analysis sample.

When the alighting stations are relaxed, in order to find some candidate alighting stations, we set a walking buffer circle. According to the previous literature, we take a 15min walk, or nearly 1km, as the walking buffer radius. The stations which are included in the buffer circle are candidate alighting stations, shown as yellow circles in Figure 4(a).

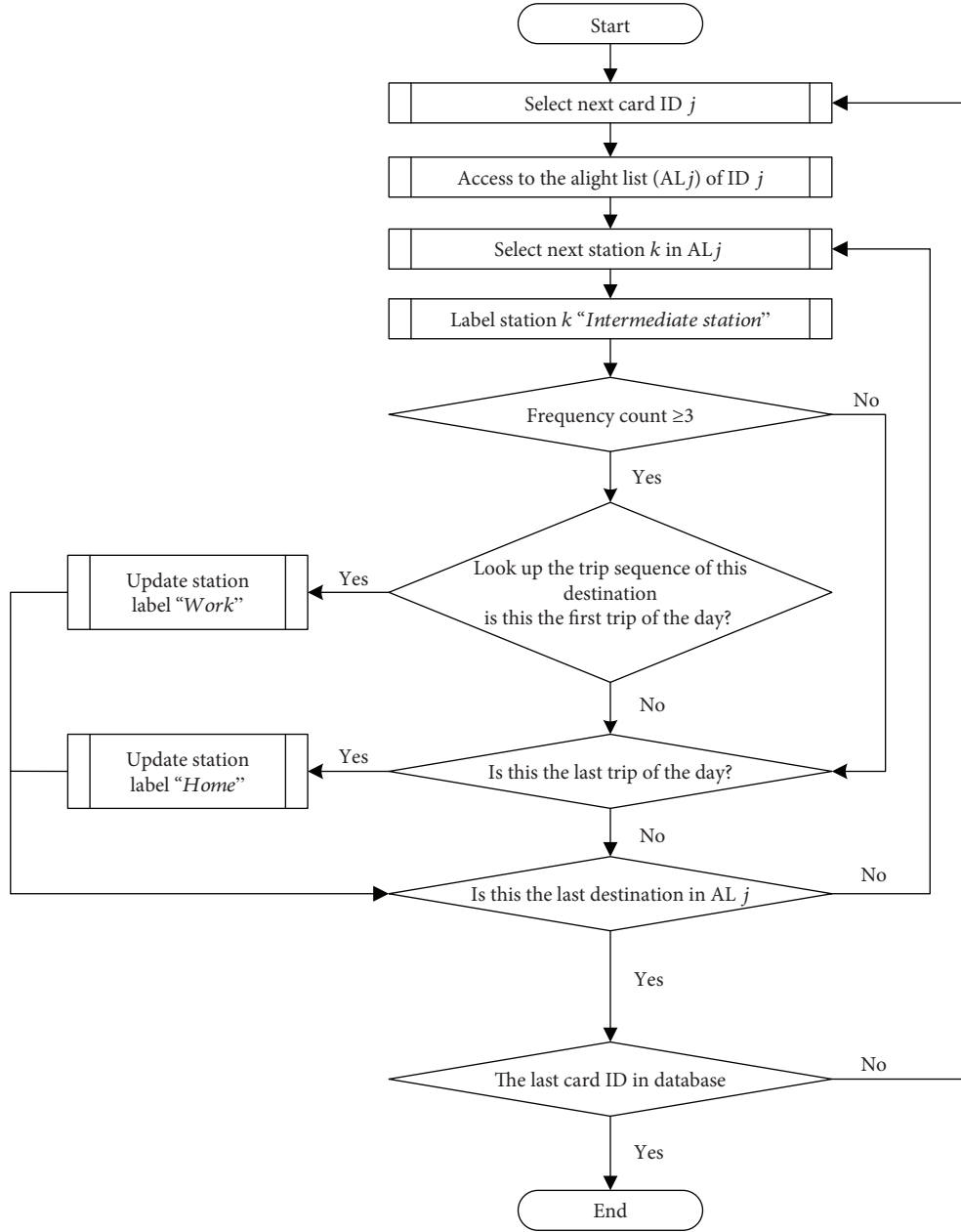


FIGURE 3: Trip purpose labelling process for work-residential trips.

3.2. The Choice Model Specification. The following notation corresponding to the choice model is used:

U_n :	Utility function of passenger n
$\alpha_{IVT}, \alpha_{WT}, \gamma_{TF}$:	Coefficients for in-vehicle time, walking time and ticket fare, respectively
$T_{IVT}^n, T_{WT}^n, T_{TF}^n$:	Value of in-vehicle time, walking time, and ticket for passenger n , respectively
$\eta_{IVT}, \eta_{WT}, \eta_{TF}$:	Takes the value one if the corresponding parameter is significant in the utility function
ε :	Random error term
J :	Choice set for each passenger

T :	Factor set
A :	Coefficient set
Γ :	Trip purpose set. 1, 2, and 3 represent work, home, and others.

3.2.1. Multinomial Logit Model (MNL). The MNL model is the prime model in transportation research which calculated the probability of each choice in a choice set. In Beijing metro, the ticket fare is distance-based, which means that passengers could walk a long distance to save money. When a passenger chooses an alighting station, there are three factors which impact the utility, in-vehicle travel time, walking time, and ticket fare. For each passenger, the utility function can be written as (1), (2), and (3).

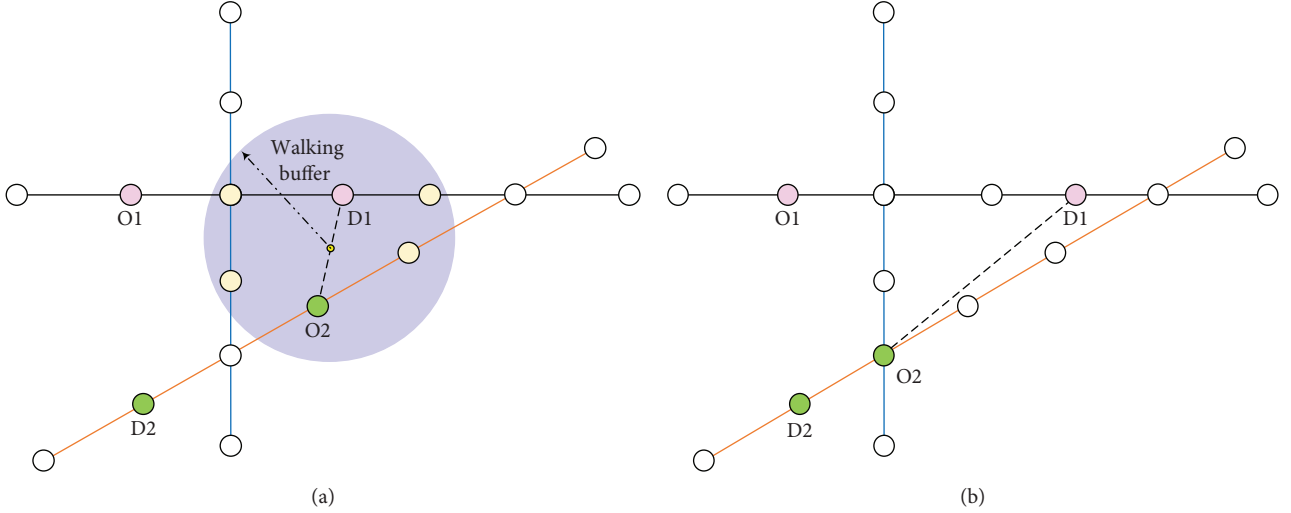


FIGURE 4: The assumption for actual destination and potential alighting station choices. Pink circles are the boarding and alighting stations of the first trip. Green circles are the boarding and alighting stations of the next trip. Yellow circles are candidate alighting stations.

$$U_n = \alpha_{IVT} \eta_{IVT} T_{IVT}^n + \alpha_{WT} \eta_{WT} T_{WT}^n + \alpha_{TF} \eta_{TF} T_{TF}^n + \varepsilon, \quad (1)$$

$$y_{nj} = \begin{cases} 1 & \text{if } U_{nj} \geq U_{nj'} \text{ for } j' \in \{1, \dots, J\}, \\ 0 & \text{otherwise} \end{cases}, \quad (2)$$

$$\eta_{IVT}, \eta_{WT}, \eta_{TF} = \begin{cases} 1 & \text{if } T_{IVT}^n, T_{WT}^n, \text{ and } T_{TF}^n \text{ are significant.} \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

The choice of alternative j by passenger n may be derived from (2) to yield the following functional form of the multinomial logit

$$P(y_{nj} = 1 \mid \alpha_{IVT}, \alpha_{WT}, \alpha_{TF}, T) = \frac{\exp(U_{nj})}{\sum_{j'=1}^J \exp(U_{nj'})}. \quad (4)$$

3.2.2. Mixed Logit Model with Independent Normally Distributed Random Coefficients. In the standard logit model, the coefficients for the same factors share the same ‘‘preference.’’ However, a different passenger could have a different preference for the same factor. Mixed logit models can be derived from a variety of different behavioral specifications, and each derivation provides a particular interpretation. The mixed logit model is defined on the basis of the functional form for its choice probabilities. The utility function in the mixed logit model and the coefficient in (1) are statistical distributions instead of a constant number, which means for each passenger n , α_n , β_n , γ_n follow distributions, and the coefficients vary over people.

$$\alpha_{IVTn} \sim f(\alpha_{IVT} \mid \theta), \quad \alpha_{WTn} \sim f(\alpha_{WT} \mid \theta), \quad \alpha_{TFn} \sim f(\alpha_{TF} \mid \theta), \quad (5)$$

where θ is the parameter of the distribution over the population, such as the mean and variance of α_n . Conditional on α_n , and assuming the unobserved term ε is iid extreme value, the

probability that passenger n chooses alternative j is the standard logit formula.

$$\begin{aligned} L_{nj}(A_n, T) &= \frac{\exp(\alpha_{IVT} T_{IVT}^{nj} + \alpha_{WT} T_{WT}^{nj} + \alpha_{TF} T_{TF}^{nj})}{\sum_{j'=1}^J \exp(\alpha_{IVT} T_{IVT}^{nj'} + \alpha_{WT} T_{WT}^{nj'} + \alpha_{TF} T_{TF}^{nj'})} \\ &= \frac{\exp(A_{nj} T_{nj})}{\sum_{j'=1}^J \exp(A_{nj'} T_{nj'})}. \end{aligned} \quad (6)$$

Different elements in A may follow different distributions (including some being fixed). Because α_n is random and unknown, with the continuous f , the probability should be the integral of the standard logit over the density of A_n .

$$P(y_{nj} = 1 \mid A_n, T) = \int L_{nj}(A, T) f(A \mid \theta) dA. \quad (7)$$

3.2.3. Mixed Logit Model with Correlated Normally Distributed Random Coefficients. As in some cases, the different elements in A may be correlated with other elements. For instance, the ticket fare in Beijing metro is distance-based, and the fare distribution could have the correlation with the distribution of in-vehicle time and coming from a joint distribution with respective means and covariance matrix.

$$\sum \alpha_{TF-IVT} = \begin{bmatrix} \text{var}(\alpha_{TF}) & \text{cov}(\alpha_{TF}, \alpha_{IVT}) \\ \text{cov}(\alpha_{TF}, \alpha_{IVT}) & \text{var}(\alpha_{IVT}) \end{bmatrix}. \quad (8)$$

We assume that the in-vehicle time and ticket fare follow a multivariate normal distribution.

$$\begin{pmatrix} \alpha_{TF} \\ \alpha_{IVT} \end{pmatrix} \sim \text{MVN} \left(\begin{pmatrix} \bar{\alpha}_{TF} \\ \bar{\alpha}_{IVT} \end{pmatrix}, \begin{bmatrix} \text{var}(\alpha_{TF}) & \text{cov}(\alpha_{TF}, \alpha_{IVT}) \\ \text{cov}(\alpha_{TF}, \alpha_{IVT}) & \text{var}(\alpha_{IVT}) \end{bmatrix} \right). \quad (9)$$

Using the Cholesky factorization [32, 33], the vector $(\alpha_{IVT}, \alpha_{TF})^T$ can be replaced by

$$\begin{pmatrix} \alpha_{TF} \\ \alpha_{IVT} \end{pmatrix} = \begin{pmatrix} \bar{\alpha}_{TF} \\ \bar{\alpha}_{IVT} \end{pmatrix} + \begin{bmatrix} p_{11} & 0 \\ p_{12} & p_{22} \end{bmatrix} \begin{bmatrix} \xi_1 \\ \xi_2 \end{bmatrix} = \bar{A} + P\xi, \quad (10)$$

where ξ_1, ξ_2 are iid standard normal variables and $PP^T = \sum \alpha_{TF-IVT}$. We applied the three kinds of logit model to the Beijing network to test which one better explains the different perceptions of users.

4. Empirical Study

From the one-week AFC dataset, there were 5.05 million transactions each workday in the Beijing metro system. For a commuter, if he takes the metro to go to work and come back home, he would make at least 2 transactions in the dataset. Averagely, these transactions are made by 2.9 million cardholders, based on the static theory and sample size calculator [34]. The cardholders' sample size is 9573 when the confidence level is 95% and the confidence interval is 1. The sample cardholders made a total of 72,645 trips. For some cardholders, they have some same routine every workday. The repeated routines have the same parameters for each candidate alighting station, so the repeated routine will not affect the coefficient of the logit model. To save the calculation time, in this study, the repeated routines are counted once. After the data cleaning and trip chaining, there are 15,057 distinct trips with inferred destination for the study.

In this study, we choose 1 km as the walking buffer distance [35, 36]. As shown in Figure 3, the candidate alighting stations can be calculated based on the final destination and the location of metro stations. Sometimes, the candidate alighting stations contain more than one category. In order to improve the estimation, we filter the stations by trip purpose. For example, within the walking buffer distance, there are 5 candidate alighting stations from A to E. We already know that this trip is a work trip. If 5 stations all belong to working stations, the five stations are all candidate alighting stations. If station B is a home station, we will keep the other 4 stations as the candidate alighting stations.

For some OD pairs, the distance between real alighting station and alternative alighting station is more than 1 km, and these OD pairs did not have candidate alighting stations, which means the passenger could only egress at that station. The logit model could not be estimated in these no-candidate alighting stations or only one alighting station case. Therefore, these records are excluded, after which 13,180 trips remained.

After applying the trip purpose labelling process, 6027 trips are labeled as work trips, 2339 trips are home trips, and the remaining 4814 trips have other purposes. We used Biogeme [37] to estimate the model coefficients.

4.1. MNL Results. For the utility function, we made the assumption that the passenger choice may be influenced by

in-vehicle time, walking distance, and ticket fare. To make sure which of these factors significantly impact the utility, we tried every factor and their combination in the model to determine which ones are mostly considered in the choice process. Table 3 excludes the results with a p value over 0.05 and shows the results of the combination of different factors for the different trip purposes.

Firstly, we consider the only single impact factor in the utility function. We found out that a single factor could not explain the passenger behavior very well, especially for the ticket fare, which did not influence the passenger choice. The walking time is more influential among three factors. The coefficient for in-vehicle time is almost the same for four types of the trips, but the coefficient for walking time differs based on different trip purposes.

For the two-factor combinations, in-vehicle time and walking time explained the user behavior as the best among the three possible combinations. This combination could illustrate every trip purpose well. Regardless of the trip purpose, there is higher disutility associated with walking time compared with in-vehicle time. On average, the walking and in-vehicle time coefficient ratio α_{WT}/α_{IVT} is 1.462. However, the sensitivity for walking time is different based on the trip purpose. Work trips have the highest penalty for walking, and the coefficient ratio is 1.635 while the coefficient ratio for home trips and other trips is 1.212 and 1.149, respectively.

As for the final log likelihood, the chi-square test was used to analyze the passenger behavior based on different trip purposes rather than overall. In this case, we use $\alpha = 0.05$ as the confidence interval. After checking the χ^2 distribution table, $\chi_{0.05,3}^2 = 7.815$, compared with $|\sum_{i=1}^3 FLL - FLL_{total}| = 55.14 > 7.815$, which indicates it is more appropriate to analyze the passenger behavior based on different trip purposes rather than overall analysis.

When we only consider the rho square, the model which has three factors in the utility function performs a little better than the two-factor combinations. But in the three-factor combination model, the coefficient for ticket fare is positive. In the Beijing metro system, the ticket fare is distance-based with a potentially high correlation with in-vehicle time. So, we could consider the positive coefficient as an adjustment for overestimation of the in-vehicle time coefficient. To be more objective, in the next step, the walking and in-vehicle time model will be as the test model for home, work, other, and total trips, and the three-factor model will be the candidate model for work, other, and total trips.

4.2. Mixed Logit Model Results. We considered the three-factor and two-factor models in the mixed logit model for utility function estimation. For each utility function, similar to the MNL analysis, we test the factors with different combinations such as single-factor or two-factor with independent or correlated distributions.

4.2.1. Three Factors in Utility Function. In-vehicle time, walking time, and ticket fare are all considered in the three-factor utility function. For each trip purpose, fourteen combinations of the mixed logit model were tested. Because of the

TABLE 3: Results of the different factor combination of the MNL model.

	Purpose	RhS	ILL	FLL	TF_Coff		IVT_Coff		WT_Coff	
					MV	PV	MV	PV	MV	PV
Single factor IVT	W	0.065	-7213.23	-6812.13	—	—	-7.37	0.00	—	—
	H	0.065	-2847.97	-2670.73	—	—	-7.12	0.00	—	—
	O	0.064	-6065.24	-5725.43	—	—	-7.74	0.00	—	—
	T	0.065	-16071.30	-15277.80	—	—	-7.52	0.00	—	—
Single factor WT	W	0.412	-7213.23	-4466.97	—	—	—	—	-18.2	0.00
	H	0.131	-2847.97	-2472.62	—	—	—	—	-9.87	0.00
	O	0.283	-6065.24	-4435.62	—	—	—	—	-13.2	0.00
	T	0.376	-16071.30	-10801.90	—	—	—	—	-16.4	0.00
Two factors WT and IVT	W	0.475	-7213.23	-4015.23	—	—	-11.2	0.00	-18.3	0.00
	H	0.21	-2847.97	-2442.40	—	—	-7.4	0.00	-9.01	0.00
	O	0.353	-6065.24	-3979.53	—	—	-13.2	0.00	-15.2	0.00
	T	0.414	-16071.30	-9729.70	—	—	-11.2	0.00	-16.4	0.00
Three factors	W	0.477	-7213.23	-4008.45	0.570	0.02	-11.4	0.00	-19.8	0.00
	O	0.354	-6065.24	-3960.00	0.665	0.00	-13.7	0.00	-15.3	0.00
	T	0.412	-16071.30	-9705.12	0.598	0.00	-13.1	0.00	-16.6	0.00

RhS = rho square; ILL = init log likelihood; FLL = final log likelihood; PV = p value; MV = mean value; W = work purpose; H = home purpose; O = other purpose; T = total trip, did not distinguish trip purpose.

computational complexity of mixed logit model estimation, only some cases could reach convergence, such as the two independent distributions for fare and in-vehicle time. However, for some combinations, even when the estimation is converged, the coefficients in the model did not pass the p value test so the model did not provide a good interpretation of the passenger behavior. Based on the convergence and p value test, only two models passed. The first one is the single walking time distribution model, which explained every trip purpose except home trips. The second one is a two-independent distribution (walking time and ticket fare) model, which only explains the total sample. No model among fourteen combinations passed for home purpose trips.

Among the passed models, the penalty for walking time is much higher than that for in-vehicle time, where the home trip has the highest coefficient ratio. Meanwhile, from other mixed logit models, we learned that the ticket fare standard deviation and in-vehicle time standard deviation are not significant for the utility function, which means that different passengers could share the same coefficient for ticket fare and in-vehicle time.

4.2.2. Two Factors in Utility Function. From the previous tests, we learned that walking time and in-vehicle time are more important factors compared with ticket fare. In this case, we only consider the walking and in-vehicle times in the utility function to see which mixed logit combinations could explain the passenger behavior well. From the results, similar to the three-factor utility condition, the single walking time distribution model also passed the p value and convergence test this time, which also explained every trip purpose except home trips. The second passed model is the independent distribution combination for

walking time and in-vehicle time, which performed well for other trip purposes.

Above all, for the work trips, other trips, and total trips, some mixed logit models could illustrate passenger behavior well and based on the rho square, mixed logit models performed a better estimation result than MNL models did. Comparing the models with rho square and p value for each coefficient, the three-factor models performed better than the two-factor models did. The selected mixed logit model for alighting station choice estimation is shown in Table 4. The single walking time distribution utility function, which is a three-factor model, is selected for work trips and other purpose trips, and the two-independent distribution (walking time and ticket fare) utility function which is a three-factor mixed logit model is selected for the total trip estimation.

4.3. Alighting Station Estimation. According to the research above, we selected the best model that could illustrate every trip purpose. This time, we randomly select another 9573 cardholders and did the same prework such as data cleaning, trip purpose labelling, and candidate station selection as presented in the first part of the empirical study. For each trip purpose, 70% of the data is used as the sample to estimate the coefficient for each model and the remaining data is used for alighting station estimation simulation by Biosim [37]. The percentage of records for which the alighting station could be estimated correctly compared with the AFC records is shown in Table 5.

From Table 5, in general, regardless of the trip purpose, approximately 71.9% of the alighting stations could be estimated correctly by the MNL model and approximately 78.6% by the mixed logit model, which performed better when estimating the alighting stations. For the different trip

TABLE 4: The selected combination of mixed models for different trip purposes.

Pur	Mixed logit model	RhS	ILL	FILL	TF_Coefficient		IVT_Coefficient		WT_Coefficient		TF_Stad		WT_Stad	
					MV	PV	MV	PV	MV	PV	MV	PV	MV	PV
W	three-factor utility function, WT distribution	0.627	-7213.23	-3202.92	0.81	0.08	-18.1	0.00	-249.00	0.00	—	—	-156.00	0.02
O	Three-factor utility function, WT distribution	0.512	-6065.24	-3212.77	1.32	0.02	-21.1	0.00	-312.00	0.08	—	—	-266.00	0.08
T	Three-factor utility function, independent distributions WT and TF	0.593	-16071.3	-8010.65	1.03	0.00	-17.3	0.00	-302.00	0.00	5.62	0.04	-213.00	0.00

Stad = standard deviation.

TABLE 5: Results for alighting station estimation based on selected MNL and mixed logit models.

Trip purpose	MNL		Mixed logit	
	Model	Percentage	Model	Percentage
Home	Two factors (WT and IVT)	66.30%	—	—
Work	Two factors (WT and IVT)	78.27%	Three-factor utility function, WT distribution	81.31%
Others	Two factors (WT and IVT)	70.74%	Three-factor utility function, independent distributions WT and TF	75.35%
Total	Two factors (WT and IVT)	72.59%		79.23%

purposes, the simulation for home trips did not perform very well and only 66.30% of the alighting stations were estimated correctly. When we map the errors on the Beijing network, we found out that the incorrect estimations are mostly around the big residential zones which are surrounded by a lot of metro stations. Because of the low penalty for walking for home trips, the alighting stations for home trips could be more flexible. This potentially could affect the estimation results for home trips. For the work trips, the MNL logit model and mixed logit model both worked best among other trip purpose simulations, likely because work trips are more predictable due to their regular patterns. For the other trips, the mixed logit model performed better than the MNL model did because the mixed logit model could illustrate passengers' deviation more properly than the MNL model could.

5. Conclusions

This study is focused on the utility function calibration for alighting station estimation for different trip purposes. The main conclusions of this paper are fivefold:

- (1) We provided a two-step trip purpose labelling process to infer the trip purpose. Based on the land use and passenger flow pattern, k -means clustering was applied to classify the stations into 7 categories. For the working-residential stations, we use the trip time and alighting station frequency to infer the trip purpose.
- (2) The walking buffer radius was applied to infer the real destination. With three assumptions and the trip chaining method, the actual destination and candidate alighting stations of the trips were inferred.
- (3) The MNL mixed logit models were proposed to illustrate passenger behavior. In order to estimate alighting stations, MNL and mixed logit models with different combinations of independent variables were discussed to illustrate passenger behavior for different trip purposes.
- (4) The influence factors for alighting station choice were tested. In the empirical study, passengers were found to have a different penalty for walking time and in-vehicle time based on trip purpose, and in general, walking time has a higher disutility. Ticket fare was

not found significant compared with walking time and in-vehicle time.

- (5) The validation test represents the feasibility of the methodology proposed in this paper. Using a validation test, the model could successfully estimate 75% of the alighting stations. The work purpose trips have higher accuracy compared with other purpose trips. This coefficient calibration helps planners understand passenger behavior better and could be used in planning and policy applications.

This research, with the real AFC alighting station data, provided a new method to infer the alighting station and could validate the passenger behavior. Comparing with the on-board survey, this one is much cheaper and more convenient. Meanwhile, this work considers the passenger alighting behavior with different trip purposes, which is a new aspect of alighting behavior analysis.

Some aspects of this study could be improved in future research. The trip purpose labelling process is based on land use, passenger flow pattern, trip time, and alighting station frequency. We can define the trip purpose as a latent variable and apply the latent logit model to capture the trip purpose based on alighting station frequency, trip sequence, and boarding time automatically. Moreover, we will apply the model to a bigger data sample in order to make a more accurate estimation of complex models such as mixed logit. Finally, if possible, passengers' sociodemographic characteristics could be incorporated in the choice model to make the choice more interesting and analyze passenger behavior in a different way.

Conflicts of Interest

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work was supported by the Foundation of China Scholarship Council, the National Natural Science Foundation Project of P.R. China under Grant no. U1434207, and the Beijing Municipal Natural Science Foundation under Reference no. 8162033. The authors thank the coworkers in CECE transit lab at University of Minnesota for their

constructive suggestions and comments that have led to a significant improvement in this paper.

References

- [1] “Wikipedia List of smart card,” https://en.wikipedia.org/wiki/List_of_smart_cards.
- [2] X. Ma, Y.-J. Wu, Y. Wang, F. Chen, and J. Liu, “Mining smart card data for transit riders’ travel patterns,” *Transportation Research Part C: Emerging Technologies*, vol. 36, pp. 1–12, 2013.
- [3] J. A. Petsinger, “Electromagnetic shield to prevent surreptitious access to contactless smartcards,” US Patent 6,121,544, 2000.
- [4] Smart Card Alliance, *RF-Enabled Applications and Technology: Comparing and Contrasting RFID and RF-Enabled Smart Cards*, Smart Card Alliance Identity Council, 2007.
- [5] M.-P. Pelletier, M. Trépanier, and C. Morency, “Smart card data use in public transit: a literature review,” *Transportation Research Part C: Emerging Technologies*, vol. 19, no. 4, pp. 557–568, 2011.
- [6] J. Y. Park, D.-J. Kim, and Y. Lim, “Use of smart card data to define public transit use in Seoul, South Korea,” *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2063, no. 1, pp. 3–9, 2008.
- [7] M. Trépanier and C. Morency, “Assessing transit loyalty with smart card data,” in *12th World Conference on Transport Research*, pp. 11–15, Lisbon, Portugal, July 2010.
- [8] P. White, M. Bagchi, H. Bataille, and S. M. East, “The role of smartcard data in public transport,” in *Proceedings of the 12th World Conference on Transport Research*, pp. 1–16, Lisbon, Portugal, 2010.
- [9] B. Agard, C. Morency, and M. Trépanier, “Mining public transport user behaviour from smart card data,” *IFAC Proceedings Volumes*, vol. 39, no. 3, pp. 399–404, 2006.
- [10] M. Bagchi and P. R. White, “The potential of public transport smart card data,” *Transport Policy*, vol. 12, no. 5, pp. 464–474, 2005.
- [11] M. Hofmann, S. P. Wilson, and P. White, “Automated identification of linked trips at trip level using electronic fare collection data,” in *Transportation Research Board 88th Annual Meeting. No. 09-2417*, pp. 1–18, Transportation Research Board Meeting, 2009.
- [12] M. Utsunomiya, J. Attanucci, and N. Wilson, “Potential uses of transit smart card registration and transaction data to improve transit planning,” *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1971, pp. 119–126, 2006.
- [13] M. Trépanier, C. Morency, and C. Blanchette, “Enhancing household travel surveys using smart card data?,” in *88th Annual Meeting of the Transportation Research Board, Washington*, pp. 85–96, Transportation Research Board Meeting, 2009.
- [14] M. Trépanier, C. Morency, and B. Agard, “Calculation of transit performance measures using smartcard data,” *Journal of Public Transportation*, vol. 12, no. 1, pp. 79–96, 2009.
- [15] M. Trepanier and F. Vassiviere, “Democratized smartcard data for transit operator,” in *15th World Congress on Intelligent Transport Systems and ITS America’s 2008 Annual Meeting ITS America ERTICOITS Japan Trans Core*, pp. 1838–1849, World Congress on Intelligent Transport Systems and its Americas meeting, 2008.
- [16] Y. Sun, M. Hrušovský, C. Zhang, and M. Lang, “A time-dependent fuzzy programming approach for the green multimodal routing problem with rail service capacity uncertainty and road traffic congestion,” *Complexity*, vol. 2018, Article ID 8645793, 22 pages, 2018.
- [17] N. Nassir, A. Khani, S. G. Lee, H. Noh, and M. Hickman, “Transit stop-level origin–destination estimation through use of transit schedule and automated data collection system,” *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2263, no. 1, pp. 140–150, 2011.
- [18] J. Barry, R. Newhouser, A. Rahbee, and S. Sayeda, “Origin and destination estimation in New York City with automated fare system data,” *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1817, pp. 183–187, 2002.
- [19] J. Zhao, A. Rahbee, and N. H. M. Wilson, “Estimating a rail passenger trip origin–destination matrix using automatic data collection systems,” *Computer-Aided Civil and Infrastructure Engineering*, vol. 22, no. 5, pp. 376–387, 2007.
- [20] J. Zhao, *The Planning and Analysis Implications of Automated Data Collection Systems: Rail Transit OD Matrix Inference and Path Choice Modeling Examples*, [Ph.D. Thesis], Massachusetts Institute of Technology, 2004.
- [21] M. Trépanier, N. Tranchant, and R. Chapleau, “Individual trip destination estimation in a transit smart card automated fare collection system,” *Journal of Intelligent Transportation Systems*, vol. 11, no. 1, pp. 1–14, 2007.
- [22] K. K. A. Chu and R. Chapleau, “Enriching archived smart card transaction data for transit demand modeling,” *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2063, no. 1, pp. 63–72, 2008.
- [23] W. Wang, J. Attanucci, and N. Wilson, “Bus passenger origin–destination estimation and related analyses using automated data collection systems,” *Journal of Public Transportation*, vol. 14, no. 4, pp. 131–150, 2011.
- [24] M. A. Munizaga and C. Palma, “Estimation of a disaggregate multimodal public transport origin–destination matrix from passive smartcard data from Santiago, Chile,” *Transportation Research Part C: Emerging Technologies*, vol. 24, pp. 9–18, 2012.
- [25] M. Munizaga, F. Devillaine, C. Navarrete, and D. Silva, “Validating travel behavior estimated from smartcard data,” *Transportation Research Part C: Emerging Technologies*, vol. 44, pp. 70–79, 2014.
- [26] J. B. Gordon, H. N. Koutsopoulos, N. H. M. Wilson, and J. P. Attanucci, “Automated inference of linked transit journeys in London using fare–transaction and vehicle location data,” *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2343, no. 1, pp. 17–24, 2013.
- [27] A. A. Alsgar, M. Mesbah, L. Ferreira, and H. Safi, “Use of smart card fare data to estimate public transport origin–destination matrix,” *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2535, pp. 88–96, 2015.
- [28] A. Tavassoli, A. Alsgar, M. Hickman, and M. Mesbah, “How close the models are to the reality? Comparison of transit origin–destination estimates with automatic fare collection data,” in *Australasian Transport Research Forum 2016 Proceedings*, pp. 1–15, Melbourne, VIC, Australia, 2016.
- [29] B. Gao, Y. Qin, X. M. Xiao, and L. X. Zhu, “K-means clustering analysis of key nodes and edges in Beijing subway network,” *Journal of Transportation Systems Engineering and Information Technology*, vol. 14, no. 3, pp. 207–213, 2014.

- [30] Z. Yue, F. Chen, Z. Wang, J. Huang, and B. Wang, "Classifications of metro stations by clustering smart card data using the Gaussian mixture model," *Urban Rapid Rail Transit*, vol. 107, no. 2, pp. 48–51, 2017.
- [31] S. Shizhao, *Operation Performance Assessment of Urban Rail Transit Based on Travel Time Delay*, Beijing Jiaotong University, Beijing, China, 2016.
- [32] C. N. Haddad, *Cholesky factorization*, Springer, 2001.
- [33] R. B. Schnabel and E. Eskow, "A new modified Cholesky factorization," *SIAM Journal on Scientific and Statistical Computing*, vol. 11, no. 6, pp. 1136–1158, 1990.
- [34] "Sample Size Calculator," <http://www.calculator.net/sample-size-calculator.html>.
- [35] S. O'Sullivan and J. Morrall, "Walking distances to and from light-rail transit stations," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1538, pp. 19–26, 1996.
- [36] K. Lu, B. Han, and X. Zhou, "Smart urban transit systems: from integrated framework to interdisciplinary perspective," *Urban Rail Transit*, vol. 4, no. 2, pp. 49–67, 2018.
- [37] "Biogeme and Biosim," <http://biogeme.epfl.ch/>.

Research Article

A Methodology for Evaluating Algorithms That Calculate Social Influence in Complex Social Networks

Vanja Smailovic ^{1,2}, Vedran Podobnik ^{2,3} and Ignac Lovrek ^{2,3}

¹Sandvik Machining Solutions AB, Stockholm, Sweden

²Social Networking and Computing Laboratory (socialLAB), Faculty of Electrical Engineering and Computing, University of Zagreb, Zagreb, Croatia

³Department of Telecommunications, Faculty of Electrical Engineering and Computing, University of Zagreb, Zagreb, Croatia

Correspondence should be addressed to Vedran Podobnik; vedran.podobnik@fer.hr

Received 22 December 2017; Revised 8 June 2018; Accepted 19 June 2018; Published 8 August 2018

Academic Editor: Xiuzhen Zhang

Copyright © 2018 Vanja Smailovic et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Online social networks are complex systems often involving millions or even billions of users. Understanding the dynamics of a social network requires analysing characteristics of the network (in its entirety) and the users (as individuals). This paper focuses on calculating user's social influence, which depends on (i) the user's positioning in the social network and (ii) interactions between the user and all other users in the social network. Given that data on all users in the social network is required to calculate social influence, something not applicable for today's social networks, alternative approaches relying on a limited set of data on users are necessary. However, these approaches introduce uncertainty in calculating (i.e., predicting) the value of social influence. Hence, a methodology is proposed for evaluating algorithms that calculate social influence in complex social networks; this is done by identifying the most accurate and precise algorithm. The proposed methodology extends the traditional ground truth approach, often used in descriptive statistics and machine learning. Use of the proposed methodology is demonstrated using a case study incorporating four algorithms for calculating a user's social influence.

1. Introduction

In 2017, more than 2.5 billion people participated in online social networking, with more than two billion of them using Facebook as one of the largest online social networking platforms [1]. In a broader sense, social networks are not just structures of interconnected humans based on their participation in such platforms. Social networks can also be built around other digital products such as telecommunication network operator services (e.g., mobile phone calls and text messaging) or even nonhuman users such as networked objects and smart devices (i.e., forming the so-called Social Internet of Things) [2]. Finally, overarching social networks can be built by combining membership and activities in multiple social networks, thus creating even more complex social networks characterised by not only millions or billions of (human and nonhuman) users but also a very rich set of possible relationships between social network users.

Importantly, understanding the dynamics within a social network requires calculating different properties of complex networks. This paper will focus on properties that describe social networks at the level of the individual user. Though two types of network properties from the aspect of the individual user can be calculated—*key actors* and *key relationships*—they differ significantly in the approach to calculating them. The property *key actors* (such as influence [3–6]) represents *global user properties* as it depends on (i) the *global positioning* of the user within the entire social network and (ii) interactions between the user and *all other users* in the social network (i.e., the property $1:N$, where N is the size of the social network). On the other hand, the property *key relationships* (such as trust [7, 8]) represents *local user properties*, given that they depend on local dynamics between pairs of individual users (i.e., $1:1$ property).

Today, there are algorithms for calculating both global and local user properties in social networks [9]. Nevertheless, evaluating the algorithms varies significantly. In evaluating

local user properties, the *ground truth approach* can be applied, which is a traditional approach often used in statistics and machine learning. The basic idea behind the ground truth approach is to collect proper objective data on the modelled property and compare the result obtained from the evaluated algorithm with the result found in ground truth data. For example, when modelling the trust relationship between social network users, ground truth data can be collected using a questionnaire where the number of social network users determines the level of trust between them and other social network users [10, 11]. Given that social trust is a 1:1 user property, surveyed users may answer questions about their level of trust towards other social network users, and consequently, this provides the ground truth data. However, the same approach for evaluating global user properties is not applicable as those properties are 1:N user properties, and only users who have full knowledge of all other social network members are able to answer the ground truth questions. Considering that today's online social networks are quite sparse [12, 13] and only social network platform operators have comprehensive data on its respective users [14], new methods are obviously needed for evaluating the modelling of global user properties in complex social networks.

This paper is a contribution to existing literature in that it proposes a novel methodology for evaluating algorithms that calculate social influence in complex social networks. The proposed methodology (i) compares algorithms that rely solely on available ego-user data for calculating ego-user social influence and (ii) identifies the most accurate and precise algorithm for predicting social influence. To the best of our knowledge, there are no other methodologies for evaluating algorithms that calculate social influence in complex social networks which are in addition able to identify the most accurate and precise calculation algorithm. The paper demonstrates different phases of the proposed methodology using a case study to calculate social influence by evaluating accuracy and precision of four different algorithms that calculate social influence.

The paper follows a specific structure. Section 2 presents the concept of *social influence* in online social networks and related work in the respective field, including the use of SmartSocial Influence algorithms. In Section 3, a methodology for evaluating the method of calculating social influence in complex social networks is introduced, and its use is demonstrated in Section 4. Next, Section 5 discusses the impact of the proposed evaluation methodology and elaborates on possible implications of identifying the best-performing social influence algorithm. Section 6 provides a conclusion, focusing on constraints of the proposed approach as well as further work in the field. The questionnaires used in method for evaluating social influence are provided in the appendix to this paper.

2. Background on Previous Work

Looking back on previous work, the paper first explains the concept of social influence in online social networks and provides examples of the main services stemming from

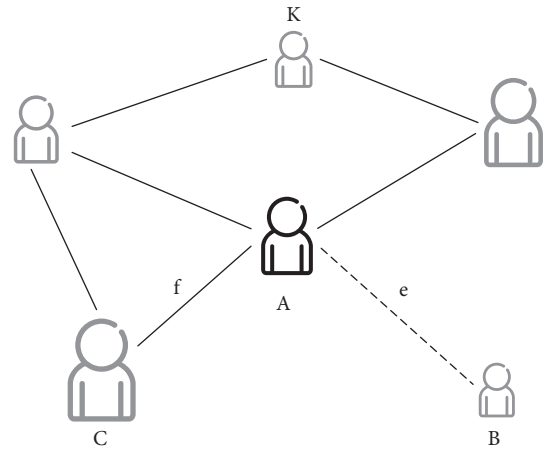


FIGURE 1: Graphical illustration of influence in a social network.

social influence. The second part in this section introduces SmartSocial Influence algorithms, a specific class of algorithms for calculating social influence.

2.1. Social Influence in Online Social Networks. Social influence is “a measure of how people, directly or indirectly, affect the thoughts, feelings and actions of others” [15]. It is a topic of interest in both sociology and social psychology, and more recently in information and communication technology (ICT), computer science, and related fields. Social influence in online social networks has seen a great rise with services such as Klout [16], Kred [17], PeerIndex [18], or Tellagence [19], all of which have demonstrated the central role of empowered users in everyday lives of ordinary people [20]. With over 620 million users scored and serving over 200 thousand business partners, Klout is an important service that is aimed at bringing influencers and brands together. Klout defines influence as “the ability to drive action” and measures it on a scale from 1 to 100, based on data from more than ten of the most popular social networking services (SNSs). As of 2017, the two most influential Klout users are Barack Obama and Justin Bieber with Klout scores of 99 and 92, respectively [21]. Figure 1 illustrates the concept of social influence using an example of six users interconnected in a social network through two types of connections. Ego-user *User A* has a greater social influence than *User B*, but less than *User C*, as denoted by the size of graphical symbols representing them. Users in the network are connected through different types of connections (e.g., *User A* and *User C* are Facebook friends, while *User A* and *User B* communicate using a text messaging service).

Numerous studies, tests, experiments, and research over a period of more than 50 years have led to various approaches in elaborating social influence [22–27]. Although rooted in social psychology and sociology, the topic of social influence has independently spread to modern online social networks with the rise of the Internet era [28].

2.2. SmartSocial Influence Algorithms. The paper compares the prediction accuracy and precision of four social influence algorithms—SLOF, SAOF, SMOF, and LRA—which all

TABLE 1: Comparison of Klout, Kred, and the *SmartSocial Influence*.

	Klout	Kred	SmartSocial Influence model
Scope of observation	“Big Brother”	“Big Brother”	Ego-network
Scale	1 to 100	1 to 1000	0 to 100
Support for multiple SNSs	✓	✗	✓
Openly published algorithm	✗	✓	✓
Telco network data-source	✗	✗	✓

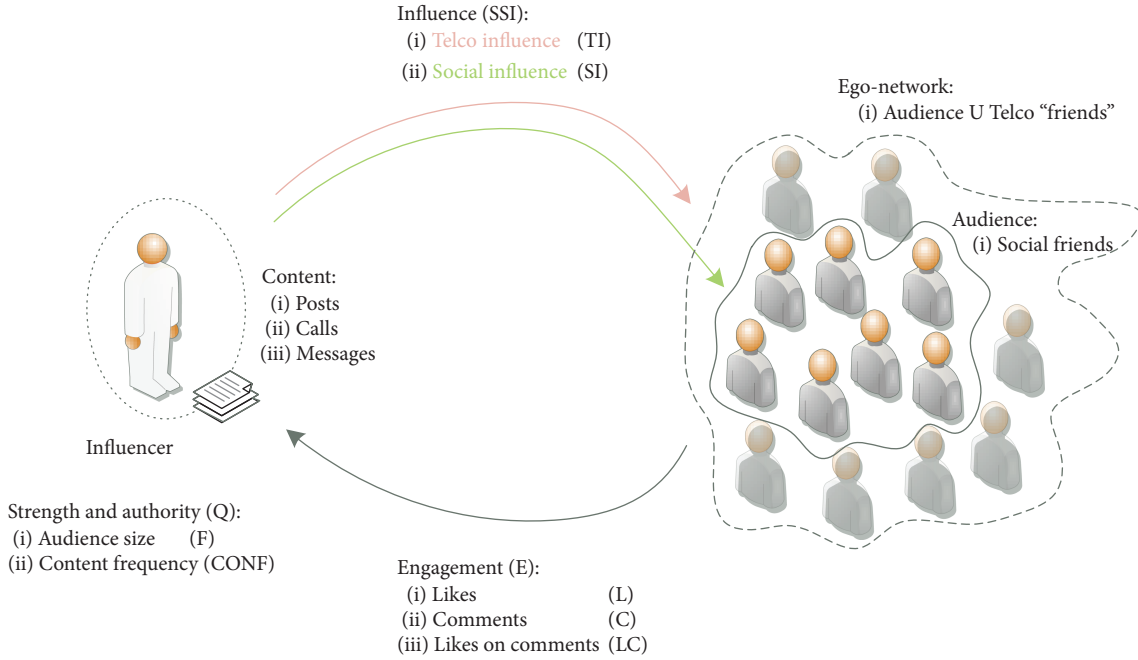


FIGURE 2: The SmartSocial Influence model.

belong to the *SmartSocial Influence* class of algorithms. SmartSocial Influence [4] is an approach to social influence modelling which takes into account the following goals: (i) inferring social influence of users based on their data retrieved from multiple, heterogeneous data-sources, namely, data on social networking services combined with data from telecommunication operators, and (ii) a multidisciplinary approach rooted in previous approaches to social influence modelling in the fields of social psychology and sociology, as well as ICT. The important difference to common approaches in social influence modelling (e.g., Klout and Kred) is the scope of observation. Unlike the SmartSocial Influence approach, the approach common to both Klout and Kred is their “Big Brother” scope of observation—they endeavor to collect vast amounts of user data to model influence that may expand beyond activities in a user’s first-degree ego-network (Table 1). Moreover, the SmartSocial Influence approach operates on smaller datasets as its scope of observation is limited to the user’s ego-network alone (Figure 1—*User B* and *User C* are in the ego-network of *User A*, but the same is not true for *User K*).

Furthermore, SmartSocial Influence explores social influence in social networks both from the *structural*

(*Structural models* analyse network structure using metrics such as degree, betweenness, and closeness centrality [29, 30], as well as eigenvector centrality [31]) and *behavioural* (*Behavioural models* analyse interaction among users, e.g., how connected users propagate or repost content, how many of them like or comment on it, or the way they engage in conversations [32, 33]) perspective—by analysing node degree (i.e., audience size), content type (i.e., quality), and content frequency (i.e., time-based longitudinal quantity) of interactions between users. Figure 2 illustrates this by identifying the main SmartSocial entities:

- (i) *Influencer*—the ego-user exerting the influence
- (ii) *Content*—items (SNS posts, calls, or messages) created by the Influencer in the SNS or telecom network
- (iii) *Ego-network*—all users who communicate with the Influencer
- (iv) *Audience*—users of a SNS who observe and *engage* with the Influencer’s *content*, a subset of the Influencer’s *Ego-network*

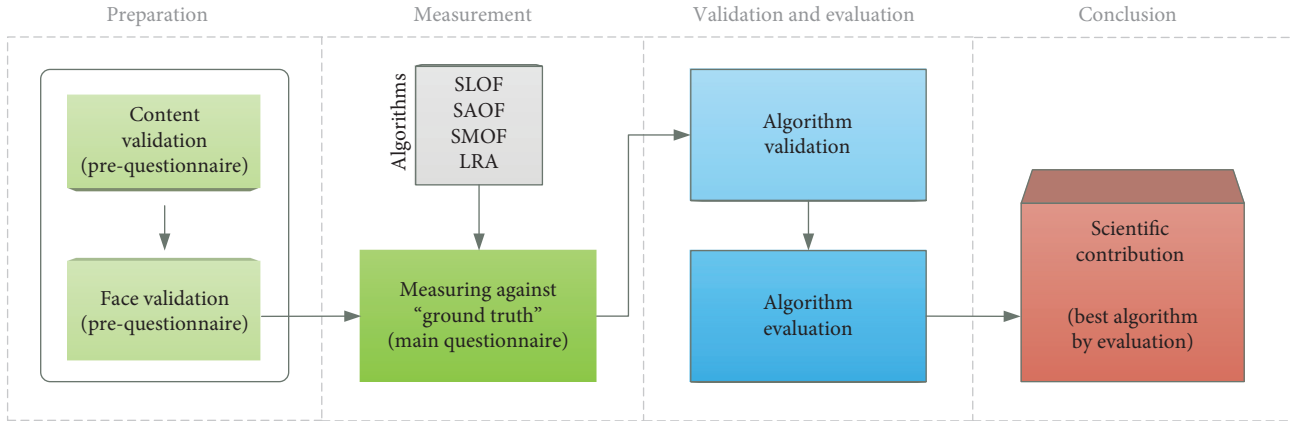


FIGURE 3: Proposed methodology for evaluating algorithms that calculate social influence in complex social networks with its four distinct phases.

An important feature is the difference between the Influencer’s *Ego-network* and the *Audience*. The Audience comprises users *connected to* the Influencer through the same SNS. The Influencer may have multiple Audiences, but for a single SNS there is only one. On the other hand, a user’s *Ego-network* comprises all users with whom the Influencer has *communicated* in the combined telecom network and SNSs. Hence, the Audience is a subset of the Influencer’s *Ego-network*. Definitions of the *relationships* between Smart-Social entities are as follows (Figure 2):

- (i) *Influence (SSI)*—*SmartSocial Influence* comprised of *TI* and *SI*
- (ii) *Telco Influence (TI)*—Influencer’s effect on the respective *Ego-network*
- (iii) *Social Influence (SI)*—Influencer’s effect on the respective *Audience*
- (iv) *Engagement*—the action taken towards the Influencer’s *content* by the *Audience* through the SNS (in the form of *likes, comments, or likes on comments*)

In short, the purpose of SmartSocial Influence algorithms is to quantify the number of engagements or interactions for a user’s publication or post (e.g., likes) with respect to the size of the audience (i.e., number of friends). In other words, a highly influential user of a SNS will have numerous posts and will be massively engaged by a large share of the respective audience.

Let us further explain the SmartSocial Influence concept on the social graph shown in Figure 1. The Influencer (or ego-user) is *User A*, connected to other users in the respective *Ego-network* (e.g., to *User B* and *User C*). *User C* is part of *User A*’s Audience; *User B* is not. Therefore, *User B* is not able to “perceive” *A*’s influence—but merely contributes to it. *User A*’s influence is defined as a property of node *A*, exerting influence on all other users in the respective Audience (part of the *Ego-network*) and described as a 1 : N relationship. This means that *User K* (not part of the Audience or *Ego-network*) is not able to “perceive” *A*’s influence.

If *User A* and *User K* were connected through the same SNS, this would then be possible. Influence is graphically represented through the size of the graphical symbol, with *User C* being the most influential in *User A*’s *Ego-network* (and Audience). In other words, Influencer’s influence is “perceivable” only by members of the Audience, whereas for the entire *Ego-network* it is “a result of contribution.” Nonaudience users of the *Ego-network* cannot “perceive” influence since they do not possess the means to do so.

More details on calculating SmartSocial Influence, along with pseudocodes of algorithms SLOF, SAOF, SMOF, and LRA, are available in [4].

3. Proposed Methodology

As previously mentioned, the SLOF, SAOF, SMOF, and LRA algorithms produce meaningful and usable results regarding one’s social influence [4, 34, 35]. However, to prove that the results hold true, they have to be *validated*.

Validity is the degree to which *evidence* supports interpretations of test scores [36]. In other words, validation reveals whether the respective algorithm produces *correct* results (that hold evidence of being truthful in the largest amount of cases) for social influence. Subsequently, evaluation leads to discovery of the best algorithm, that is, the most *accurate* and *precise* algorithm. Differences between these two terms are explained in detail in Section 3.3. In short, the methodology for evaluating algorithms provides insights into identifying the best social influence algorithm.

The proposed methodology takes place in *four phases* (Figure 3): (i) the first phase is a *preparatory* step; (ii) the second phase involves taking *measurements* of the performances of algorithms with respect to “ground truth”; (iii) the third phase is *validatory* and *evaluatory* regarding the algorithms; and (iv) the last phase is *conclusive*.

Namely, the first phase involves *pre-questionnaires*, essential to forming the *main questionnaire* in a scientifically valid manner in the second phase. The third phase uses the *main questionnaire* to validate the algorithms, and the fourth phase provides a conclusion by identifying the *best* algorithm.

The four phases of the proposed methodology for evaluating algorithms that calculate social influence in complex social networks are described in more detail further on.

3.1. Evaluation of Social Influence Calculation: Preparatory Phase. The *main questionnaire* (MQ) is employed to validate social influence results produced by each of the algorithms. Just as any other questionnaire, the MQ is a test given to respondents in form of *questions*. Each question represents a test *item* (Q_i). To make sure the (MQ) measures what it is *supposed to measure*, two different facets of validity have to be satisfied for each of the selected *items* (questions). First is *content validity* and second is *face validity*. These are tools incorporating a rigorous scientific method—validation of an artefact, in this case, the *main questionnaire* (MQ).

3.1.1. Content Validity Test. Content validity [37], also known as *logical validity*, indicates to what degree each of the test *items* measures what it should be measuring (i.e., test content). A test created by a single author may or may not be content valid, given that an author may be *biased* and create a test that does not measure what it is supposed to.

Therefore, as the content validity test, a number of individuals who are sociology/psychology researchers were asked to validate questions *directly*, after being provided with definitions of social influence.

3.1.2. Face Validity Test. Items that pass the content validity process are advanced into the *face validity* process. In contrast to content validity, face validity does not show how good the test measures what it is supposed to measure, but what it *actually appears to measure*. In other words, despite the scientific rigour of content validity, it is face validity that ensures correctness of the *interpretation* of questions and their *relevance* of the participants' answers. Some researchers argue that *face validity* is somewhat unscientific [38]; nonetheless, the test is face-valid if it *seems* valid and meaningful to the participants taking the test, decreasing its overall bias levels [39].

For that purpose, after establishing content validity with the *content validity pre-questionnaire* (PQ_{CV}), an additional pre-questionnaire should be used for establishing face validity (PQ_{FV}) of items Q_{CV_i} . The basic principle remains the same as with content validity test, but the implementation is somewhat different. Since those who are not sociology/psychology researchers are not familiar with definitions and concepts of social influence, asking them to validate questions *directly* is inappropriate. Providing them with *definitions of social influence* beforehand, as is the case with the sociology/psychology researchers, may *distort* the responses and undermine face validity. (This design approach, to the best of its ability, endeavors to mitigate the *Hawthorne* (or *Reactivity*) effect [40], the *Observer-expectancy effect* [41], and to the greatest extent the bias resulting from the *Demand characteristics* [42].) Therefore, as the face validity pre-questionnaire tests a number of nonexpert individuals who are not sociology/psychology researchers, they were asked to validate questions *indirectly*, without being provided

with definitions of social influence beforehand in order to avoid bias.

3.2. Evaluation of Social Influence Calculation: Measurement Phase. The results of the content-validity and face-validity tests are the basis for compiling the *main questionnaire* (MQ). The MQ serves as the ground truth or the “golden standard”—its purpose is to validate and evaluate algorithms SLOF, SAOF, SMOF, and LRA. Each question Q_i in the MQ requires the participant to read an “imaginary Facebook post” and choose between Facebook friends who exert a greater personal influence (either on *emotions*, *actions*, or *behaviours* as described in the question).

What each question Q_i (the total number of questions in the questionnaire MQ is denoted as $|MQ|$) explores is, in fact, the *greater social influencer* among two Facebook friends in each pair. All of the questions pose the same question *indirectly*—which of the two Facebook friends has greater social influence? A total of $|Pair|$ Facebook-friend pairs are offered as answers to each question. These pairs are permuted between questions, to avoid participant boredom and fatigue. All $|Pair|$ friend pairs in $|MQ|$ questions equal $|Pair| \times |MQ|$ observations *per participant*. Combined with $|PAR|$ participants, there are a total of $|Pair| \times |MQ| \times |PAR|$ observations *per algorithm*. Observations were carried out in the manner described below.

First, consider a single participant, denoted as PAR_j . For each Pair offered as answers to questions, there are two Facebook friends—left FB friend and right FB friend. Each Facebook friend has four social influence scores attached to it, as calculated per respective algorithm ALGO— SI_{SLOF} , SI_{SAOF} , SI_{SMOF} , and SI_{LRA} . Calculating the difference between social influence scores (SI) of the left and right Facebook friends yields a new measure defined as

$$\begin{aligned} \Delta_p(\text{Pair}, \text{ALGO}) &= SI_{\text{ALGO}}(\text{left FB friend}) \\ &\quad - SI_{\text{ALGO}}(\text{right FB friend}), \end{aligned} \quad (1)$$

$$p \in \{1, 2, \dots, |Pair| \times |MQ| \times |PAR|\},$$

where

$$\begin{aligned} \text{ALGO} &\in \{\text{SLOF}, \text{SAOF}, \text{SMOF}, \text{LRA}\}, \\ (\text{left FB friend}, \text{right FB friend}) &\in \text{Pair}, \end{aligned} \quad (2)$$

$$\text{Pair} \subseteq \text{FB Friends}(PAR_j), j \in \{1, 2, \dots, |PAR|\}.$$

Since social influence scores (SI) attain values between 0 and 100, Δ_p attains values between -100 and 100 . The value Δ_p in fact represents “measurement of certainty” with which the respective algorithm determines that the left FB friend has *greater* social influence than the right FB friend has, or vice versa. For example, $\Delta_p = -42$ means that “the right FB friend is more influential than the left FB friend by 42.”

An algorithm that *correctly* measures a *more influential* Facebook friend in a Pair (with respect to the participant's answer) gets *rewarded*, whereas the algorithm that *incorrectly* measures it gets *punished*. This means that Δ_p is a *single measurement*.

How do algorithms get *rewarded* or *punished* with respect to a *correct* or *incorrect* measurement? Let us define the measurement score of a Pair as

$$ms_p = \varepsilon_p \cdot \frac{\Delta_p}{100}, \quad (3)$$

where for each Pair of Facebook friends found in the “ground truth,”

$$\varepsilon_p = \begin{cases} -1, & \text{if more influential is the friend on the right} \\ 1, & \text{if more influential is the friend on the left.} \end{cases} \quad (4)$$

This simply means that for *correctly* measuring the more influential Facebook friend in a given Pair, an algorithm receives a measurement score of $ms_p = +|\Delta_p|/100$. In contrast, it receives $ms_p = -|\Delta_p|/100$ for an *incorrect* measurement. (One might argue whether this approach is justified. Replace the “algorithm” with a Geiger instrument for measuring radioactivity and consider the logic of “measurement confidence” as follows. If the Geiger instrument is correct, it should be rewarded. If not, it should be punished. Now, imagine an instrument that measured $\Delta_p = 100$ between two people, determining the person on the left +100 more radioactive than the person on the right. If incorrect, the algorithm should be severely punished—for potentially endangering the person on the right. If correct, it should be maximally rewarded for saving the life of the person on the left. The same holds true for smaller measurements (e.g., moderate punishment/reward for $\Delta_p = 5$) and all other variations.)

3.3. Evaluation of Social Influence Calculation: Validation and Evaluation Phase. In the phase that follows, it is important to distinguish between the two constructs—validation and evaluation of algorithms. Validation yields *proof* that the algorithm produces *sound* and *truthful* social influence scores with respect to participants’ answers, which are taken as the “ground truth.”

The single criterion for validating an algorithm is as follows:

- VI. The overall amount of *correct* measurements (from the measurement phase) is greater than half (50%) with respect to participants’ answers.

In other words, the ALGO algorithm is valid if its *average* measurement score ms_p is greater than zero by a statistically significant margin. Statistically speaking, this shows that the algorithm did not *bet* and correctly determined the greater social influencers by *sheer chance* alone, but by being aligned with the ground truth found in the participants’ answers. Since validation is a binary variable, an algorithm can either be *valid* or *invalid*. There is no comparison between the algorithms in terms of their validity; one cannot be *more valid* than the other.

Evaluation, on the other hand, enables ranking of the algorithms. As can be seen, the algorithm with the *greatest* amount of both correct and “confident measurements” (utilising greater $|\Delta_p|$) is declared the *most truthful*.

Averaging over all of the Facebook friend pairs, the *most truthful* algorithm can be identified using the evaluation criteria prioritized as follows:

- E1. The greatest *average* measurement score ms_p
- E2. The smallest *spread* (also known (in statistics) as variability, scatter, or dispersion) of measurement scores ms_p in the distribution

To paraphrase using statistics vocabulary, the criteria for the *most truthful* algorithm would be as follows:

- E1. The algorithm with the greatest *accuracy*
- E2. The algorithm with the greatest *precision*

The first criterion assumes the *average* to be *true* as a point-estimation through a sufficient amount of data points (in our case, exactly 1,152 measurement scores per algorithm (12 Facebook friend pairs in 6 questions given to 16 participants)) Let us be clear that each algorithm is completely precise with respect to repeating a *single* measurement; that is, repeating the measurement of the same Pair will always return an *identical* value. Precision is *not* used in the sense of an internally intrinsic measure, but in comparing against the ground truth. It is a question of how precise an algorithm is when put up against participants’ answers in the real world.

3.4. Evaluation of Social Influence Calculation: Conclusion Phase. Importantly, the underlying research problem should be evident—to correctly determine the *more influential* of the two Facebook users, with the ultimate goal of ranking them according to their social influence score SI. Knowing a certain SI score is inadequate per se unless comparable to another SI score. In the most general sense, this approach to evaluating relates to *maxDiff* and *best-worst choice* methodologies [43, 44] and is used to establish which of the algorithms produces the best results in a *relative* (ranked), not *absolute* (nonranked) manner.

4. Methodology in Practice—Evaluating the SmartSocial Algorithms

In the previous section, four phases of the proposed methodology for evaluating calculation of global user properties in complex social networks were explained. In this section, use of the proposed methodology will be demonstrated using a case study of calculating social influence by evaluating the accuracy and precision of four social influence algorithms—SLOF, SAOF, SMOF, and LRA—which all belong to the SmartSocial Influence class of algorithms.

4.1. Evaluation of the SmartSocial Algorithms: Preparation Phase

4.1.1. Content Validity Test. To avoid bias in selecting questions for the MQ, a *content validity pre-questionnaire* (PQ_{CV}) has to be employed. In our case, the PQ_{CV} was given to a group of 22 experts (All the experts were graduates from the Faculty of Humanities and Social Sciences, University of Zagreb, familiar with the field of social influence through (social) psychology and sociology classes and research.) (EXP_k) on the subject of *social influence*. Before answering questions, experts were shown important

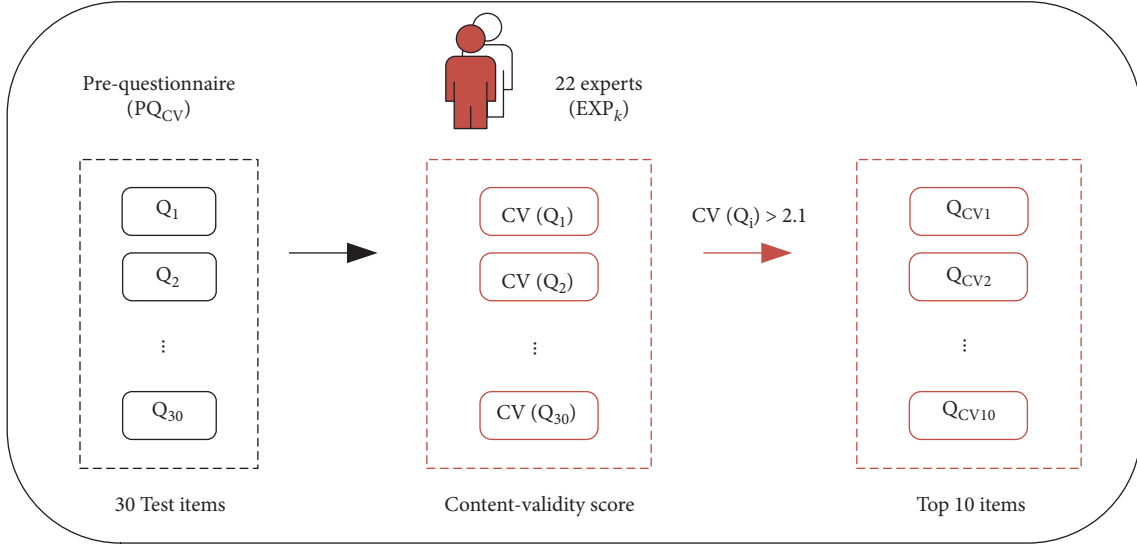


FIGURE 4: The process of content validation for questions Q_i .

definitions of social influence, which ensured that all of them utilize the same underlying concept.

The content validation process is shown in Figure 4. Of the 30 questions (Q_i) from the total in PQ_{CV} , only the *top best-rated 10* passed through to the next step of validation. An expert was given the opportunity to score each question Q_i on a scale of 1 to 5, depending on how well it explored social influence in line with the given definitions. After PQ_{CV} was finished, each question score was averaged across all experts. This produced the content-validity score for a particular item, denoted as $CV(Q_i)$.

According to [37], for a group of 22 experts, each item has to be rated above 0.42 out of a maximum of 1 in order to pass as *valid for content*. On a scale of 1 to 5, this equates to 2.1, which is the threshold for selecting a question Q_i as content-valid. In other words, the statement $CV(Q_i) > 2.1$ must hold true for each of the questions Q_i to be content-valid.

All questions Q_i , as well as their respective $CV(Q_i)$ scores, can be found in Appendix B. *Pre-questionnaire (content validity)*. Of the 30 questions in the PQ_{CV} , 29 questions passed the content validity test and the top 10 with the highest $CV(Q_i)$ scores were selected for the next phase—the face validity test (PQ_{FV}).

4.1.2. Face Validity Test. In this phase, a pre-questionnaire of top 10 questions that passed PQ_{CV} was given to 22 individuals who were not experts (NEX_x) on the subject of *social influence*. As is evident in Appendix C, these questions do not address *social influence* per se in any shape or form but ask the nonexpert to read an “imaginary Facebook post,” and each time a different one. The “post” is followed by a description regarding the effect either on personal *emotions*, *actions*, or *behaviours* with respect to a given imaginary Facebook post. Next, the nonexpert is instructed to choose which Facebook friend would cause a greater effect either on *emotions*, *actions*, or *behaviours* as described in the question. Facebook friends are presented in pairs, with each question

holding the *identical* four Facebook-friend pairs as answers. The face validation process is shown in Figure 5.

A note here is that pairs themselves are not important in this phase; the point of PQ_{FV} lies in a “hidden” 11th question which reveals itself to the nonexpert once PQ_{FV} is finished. This last question provides the necessary *definitions of social influence* and then asks the nonexpert to choose—in accordance with the provided definitions—the *more influential* friend among the *same* four Facebook-friend pairs used beforehand. In essence, it provides a filter of “correct answers” for all of the previous 10 questions. Details about the face validity test and face validity scores $FV(Q_{CVi})$ with respect to the 10 questions in PQ_{FV} can be found in Appendix C.

Exactly *four* Facebook-friend pairs are offered as answers in each Q_{CVi} because questions can have anything between 0 and 4 “correct answers,” based on “criteria” in the 11th question. Upon shifting the scale by +1, this yields a scale from 1 to 5, which corresponds directly to the previously used scale in PQ_{CV} , which is important for equal treatment of both *content-* and *face-validity*. Again, each question is given a score $FV(Q_{CVi})$ as an average across all scores of the 22 nonexperts.

Finally, the top 5 questions were chosen for MQ, with an additional Q_{MQ6} . This additional question was important for MQ as it involved a topic referring to the *mobile telecommunication operator*. In fact, it is both content- and face-valid (see Appendix B and Appendix C).

4.2. Evaluation of the SmartSocial Algorithms: Measurement Phase. To avoid fatigue [38], participants in the *main questionnaire* MQ were asked 6 questions, leading to $|MQ| = 6$. The highest scored questions that passed *content validity* as well as *face validity* pre-questionnaires were chosen to be part of the MQ, as described in the previous subsection. A total of 16 participants participated in the MQ, leading to $|PAR| = 16$. A total of 12 Facebook-friend pairs were offered as answers to

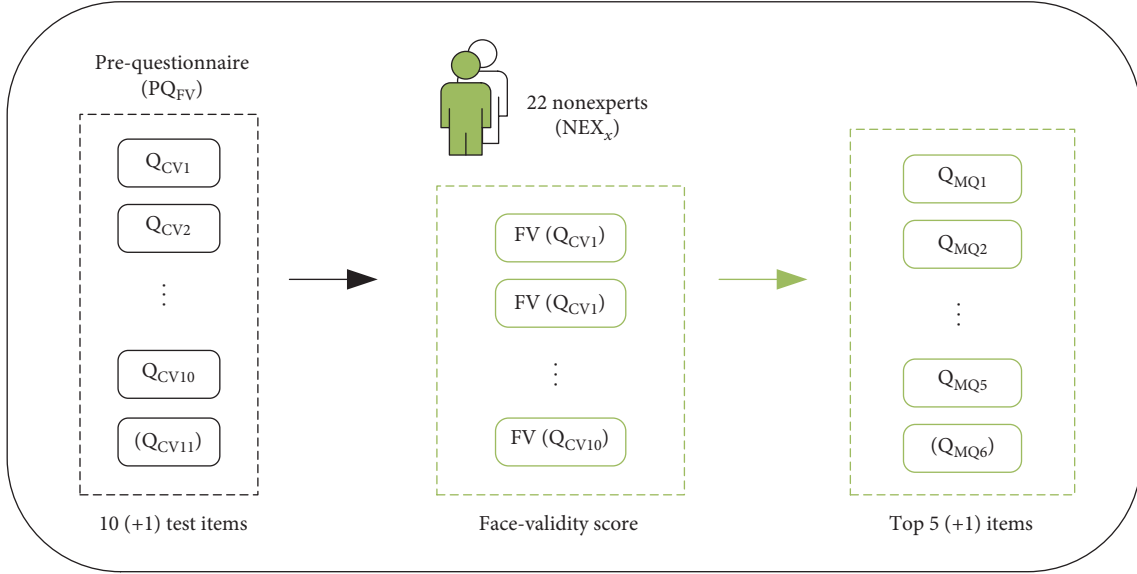


FIGURE 5: The process of face validation for questions $Q_{FV,i}$.

each question, leading to $|\text{Pair}| = 12$. All 12 friend pairs in 6 questions equal 72 observations *per participant*. Combined with 16 participants, there are a total of 1152 observations *per algorithm*.

More details about the specific questions which were part of the MQ are given in Appendix D, while more details about the metrics used in the measurement process are given in Section 3.2.

4.3. Evaluation of the SmartSocial Algorithms: Validation and Evaluation Phase

4.3.1. Validation Using Measurement Scores. Figure 6 shows the distribution of final measurement scores ms_p for the SLOF algorithm. Individual measurement scores are retrieved for *each pair* of Facebook friends and can attain values in the range $[-1, 1]$ (i.e., $+\Delta_p/100$ or $-\Delta_p/100$ for a certain pair). Given that there are 6 questions with 12 pairs across 16 participants, the distribution shows a *total of 1152 measurement scores*.

At the given resolution, it becomes evident that the SLOF ms_p distribution is multimodal, having five *modes*. This observation holds true for other (SAOF, SMOF, and LRA) ms_p distributions as well. The reason lies in the somewhat nonrandom method of selecting Pairs and their respective differences in SI, which produces a nonnormally distributed Δ_p that sometimes overlaps or repeats, producing several *modes*. (Although desirable, it was not feasible to select truly random values of Δ_p due to the fact that the SI score distributions from SmartSocial Influence algorithms are not normal. Particularly in the case of the SLOF algorithm, a high-kurtosis distribution of SI scores exists, resulting in the measurement score ms_p distribution displaying “groups” based on similar Δ_p .)

It becomes evident that the majority of *measurement scores* (ms_p) are greater than zero. To be exact, 58% of them

are positive. This means that SLOF correctly determined the greater influencer in 668 out of 1152 pairs. Validity is similar to SLOF for SAOF (Figure 7), SMOF (Figure 8), and LRA (Figure 9) as well. They correctly determined 61%, 61%, and 64% of greater influencers in pairs, respectively.

To prove the validity of each algorithm, let us formally use statistical hypothesis testing in the following manner. Consider the statement “SLOF algorithm works by sheer guessing of the correct measurements” as the *null hypothesis* (H_0) being tested. The *test statistic* is “the number of correct measurements.” Let us set the *significance level* (α) at 0.01. The *observation* is “668 correct measurements out of 1,152.”

Therefore, p value is the probability of observing between 668 and 1152 *correct* measurements with the null hypothesis being true. Calculation of p value is as follows [45]:

$$p \text{ value} = \left(\frac{1}{2}\right)^{1152} \cdot \sum_{d=668}^{1152} \binom{1152}{d}, \quad (5)$$

which equals approximately $3.28 \cdot 10^{-8}$. In other words, guessing more than 58% out of 1152 measurements correctly (p value) is statistically very improbable. Since p value $\ll \alpha$, the null hypothesis is strongly rejected.

Therefore, the *logical complement* of the null hypothesis ($\neg H_0$) can be accepted, stating that “the SLOF algorithm does *not* work by the sheer guessing of correct measurements,” which validates the algorithm. Considering that the other algorithms (SAOF, SMOF, and LRA) have even greater *test statistics*, the null hypothesis can be safely rejected for them as well. The summary is shown in Table 2.

To summarise, all of the algorithms were *successfully validated* by satisfying the single criterion for validation (V1). Note that the percentages of correct measurements are *not* comparable across the algorithms—which may be 58% percent of “correct pairs” for SLOF, and is not comparable with 64% of “correct pairs” for LRA, given that pairs are associated with different “weights” (Δ_p) to them. This is the reason, for

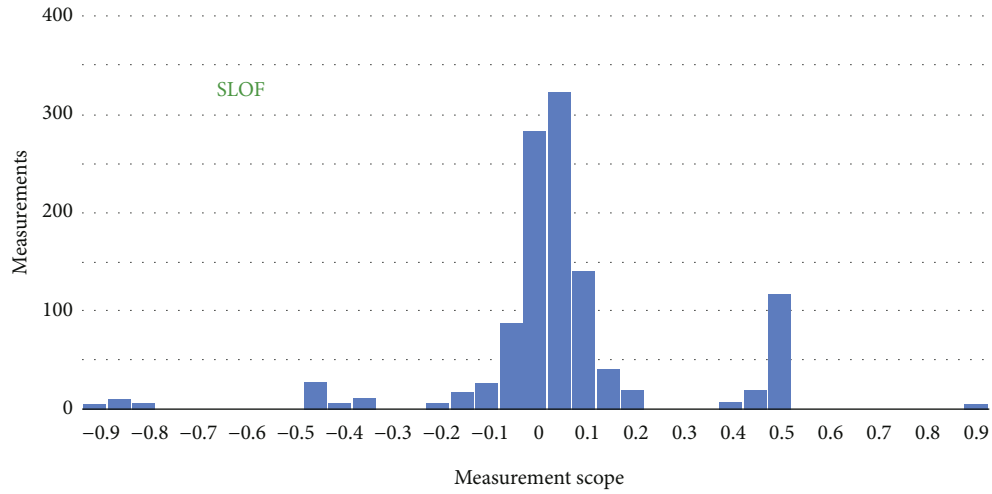


FIGURE 6: Distribution of measurement scores for the SLOF algorithm. Extreme values and outliers are not shown; the distribution shows 936 (of 1152) or 81% of all measurement scores for SLOF.

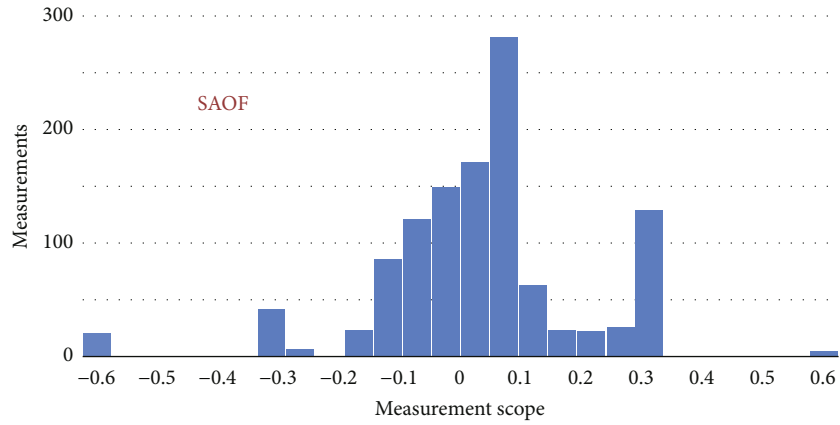


FIGURE 7: Measurement score distribution for the SAOF algorithm.

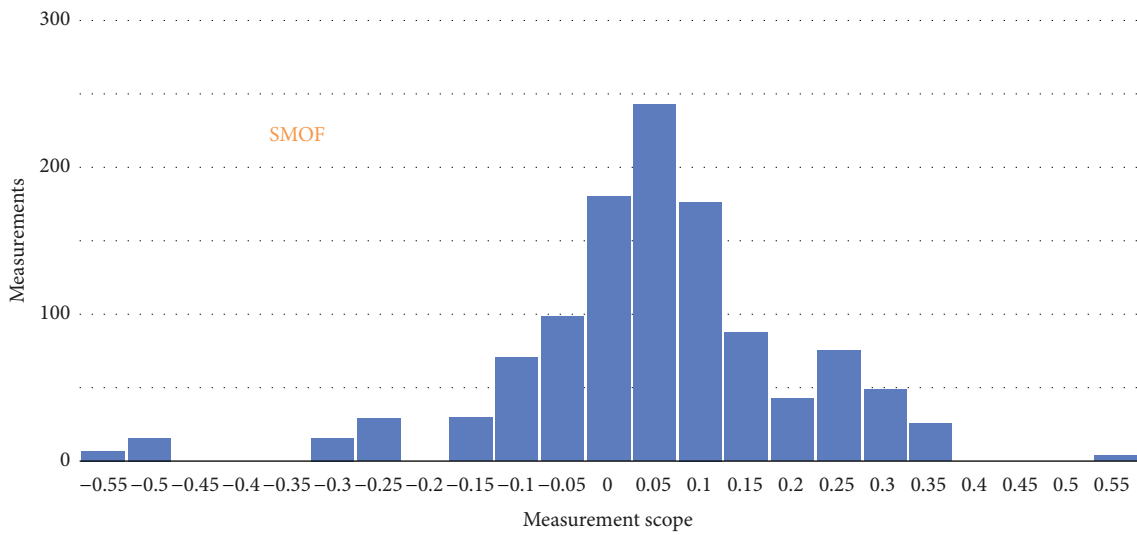


FIGURE 8: Measurement score distribution for the SMOF algorithm.

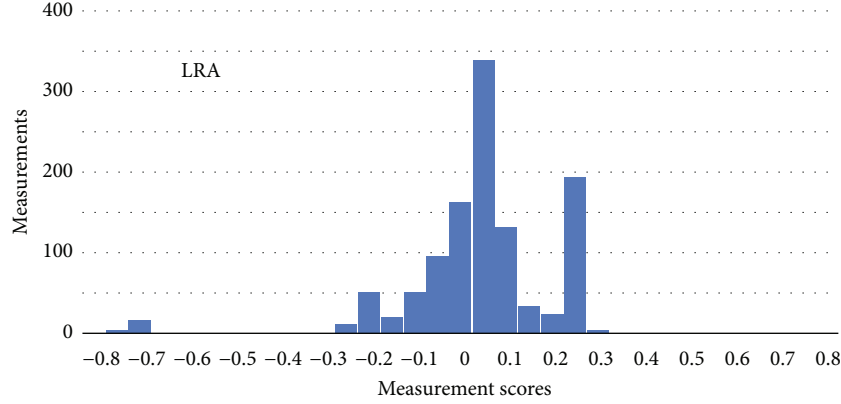


FIGURE 9: Measurement score distribution for the LRA algorithm.

TABLE 2: Summary of statistical hypothesis testing with the goal of social influence algorithm validation.

Testing parameters	SLOF	SAOF	SMOF	LRA
H_0 (null hypothesis)	Algorithm being tested works by the sheer guessing of correct measurements			
Test statistic	Number of correct measurements			
Significance level (α)	0.01			
Observation (number of correct measurements)	668 in 1152	706 in 1152	706 in 1152	733 in 1152
p value	$3.28 \cdot 10^{-8}$	$9.08 \cdot 10^{-15}$	$9.08 \cdot 10^{-15}$	$8.52 \cdot 10^{-21}$
Conclusion ($\neg H_0$)	Algorithm being tested does <i>not</i> work by the sheer guessing of correct measurements.			
Validation successful	✓	✓	✓	✓

example, that LRA is not *more valid* than SLOF. The mentioned challenge of ranking is a task for *evaluation*, not validation, as will be explained in detail in the following subsection.

4.3.2. Evaluation by Comparison. Figure 10 shows a boxplot of *measurement scores* ms_p for each algorithm. Although all four algorithms belong to the same SmartSocial Influence class of algorithms, LRA is denoted with a different color (light blue) since it is the only solely literature-based algorithm (i.e., the benchmark algorithm) and the predecessor to SLOF, SAOF, and SMOF (which are the upgraded versions [4]). The measurement scores are retrieved *per pair*, as either correct ($+\Delta_p/100$) or incorrect ($-\Delta_p/100$). A summary of the boxplot is given in Table 3.

Let us first consider the *first criterion* for evaluation (E1)—the greatest *average* measurement score (\overline{ms}_p), denoted with a “+” symbol in Figure 10. The greatest \overline{ms}_p is found in SLOF and equals 0.0358. The smallest \overline{ms}_p is found in SMOF and equals 0.0240. In between are SAOF with 0.0271 and LRA with 0.0250 \overline{ms}_p , respectively. Observing the averages, SLOF and SAOF are evaluated as more truthful, while SMOF as less truthful than their predecessor LRA—showing a +43.4%, +8.7%, and −3.8% difference in \overline{ms}_p , respectively. Based on the *first criterion* used for evaluation (E1), the two algorithms—SLOF and SAOF—demonstrated and clearly showed significant improvements

over their predecessor, the LRA algorithm, and provided a scientific contribution. In other words, this means that, on average, SLOF and SAOF surpass LRA (accuracy) in correctly determining the *greater* influencer between the two—while considering the *differences* in their respective SI scores.

Let us now consider the *second criterion* for evaluation (E2)—the smallest *spread* of *measurement scores*. Statistically speaking, there are various estimators that estimate the *spread* of values across a distribution. They are called *estimators of scale*, in contrast to *estimators of location* (i.e., such as mean or median) [46–48]. The view is that the first criterion used for evaluation utilized the sample mean (average) as an *estimator of location* to rank the algorithms.

When dealing with a large amount of data or variable measurements, *outliers* and *extreme values* are common, along with certain departures from parametric distributions. To be “resistant” to outliers or underlying *parameters* of a distribution (namely nonnormality, asymmetry, skewness, and kurtosis), robust estimators of scale have to be employed [49]. In such situations, performance of robust estimators tends to be *greater* than their nonrobust counterparts (such as *standard deviation* or *variance*) [50].

On the other hand, statistical efficiency (In (descriptive) statistics, *efficiency* of an estimator is its performance with regards to the (minimum) necessary number of observations. A more efficient estimator needs fewer observations; given that the amount of observations is not an issue with measurement scores, lower efficiency is not problematic.) of robust

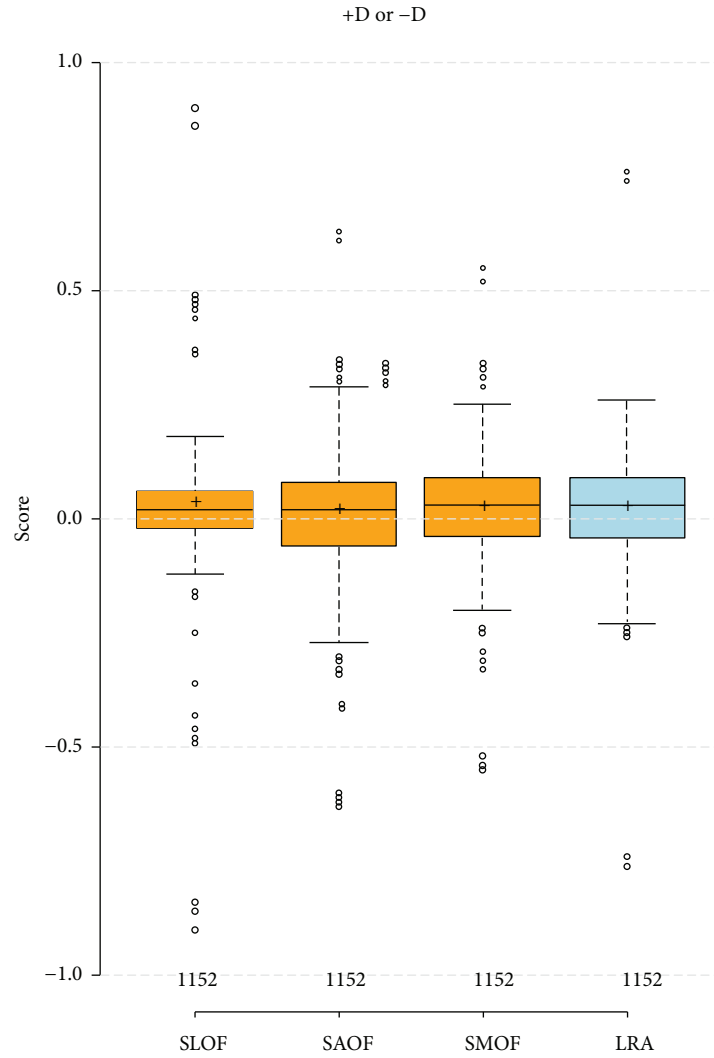


FIGURE 10: Boxplot of measurement scores for each algorithm. Boxplot uses values that are less than a 1.5x interquartile range from the 1st and/or 3rd quartile for the lower and upper whiskers (as defined by Tukey [51]); box lower-bound is the 25th percentile, middle-bound is the median, and upper-bound is the 75th percentile, and “+” denotes an average (mean) value. Plotted using BoxPlotR [52].

TABLE 3: Summary of measurement scores as boxplot statistics.

Boxplot statistic	SLOF	SAOF	SMOF	LRA
Number of measurement scores	1152	1152	1152	1152
Maximum value	0.90	0.63	0.55	0.76
Upper whisker	0.18	0.29	0.25	0.26
75th percentile	0.06	0.08	0.09	0.09
Average (mean)	0.0358	0.0271	0.0240	0.0250
Median	0.02	0.03	0.03	0.03
25th percentile	-0.02	-0.06	-0.04	-0.04
Lower whisker	-0.12	-0.27	-0.20	-0.23
Minimum value	-0.90	-0.63	-0.55	-0.76

estimators tends to be smaller. Caution should be used when seeking “resistance” to outliers—sometimes, they carry *very important* information, such as the early onset of *ozone holes* which were initially rejected as outliers [53]. Since

measurement scores are a large amount of nonparametrically distributed data containing outliers, utilization of *robust estimators of scale* is mandatory.

A thorough description of all estimators is beyond the scope of this paper; instead, only appropriate estimators are selected together with an explanation for selecting them. The estimator needs to be *appropriate* for comparing *spread* between *measurement score* distributions. The appropriate estimator successfully avoids all the “pitfalls” of the characteristics in *measurement score* distributions and additionally [48, 49, 54]

- (i) is applicable to variables using *interval* scale and not just ratio scale (*Ratio scales* (e.g., Kelvin temperature, mass, or length) have a nonarbitrary, meaningful, and unique zero value. *Interval scales* (e.g., Celsius temperature) explain the degree of difference, but not the ratio between the values. A measurement score of 0.4 is greater than that

of -0.1 , but not proportionally so. Additionally, a measurement score of 0.0 does not indicate “no determination.” Hence, measurement scores use an interval scale.)

- (ii) is applicable to variables containing both negative and positive values
- (iii) is insensitive to mean (average) value close to or approaching zero
- (iv) is insensitive to variables of which the mean (average) value can be zero
- (v) is invariant (robust) to underlying distribution of the variable (i.e., nonparametric)
- (vi) is invariant (robust) to a small number of outliers
- (vii) is invariant (robust) to asymmetry of the distribution and location estimate (or choice of central tendency, e.g., mean or median)
- (viii) has the best possible breakdown point (The breakdown point of an estimator is the proportion of incorrect observations an estimator can handle before producing incorrect results [55]. For example, consider the median; its breakdown point is 50% because that is the amount of incorrect observations introduced for it to have an incorrect median. The maximum achievable breakdown point is 50% , since that is the threshold at which it becomes impossible to discern correct from incorrect data. IQR has a breakdown point of 25% ; Rousseeuw-Croux S_n and Q_n achieve 50% . The higher the breakdown point of an estimator, the greater its robustness.)

The interquartile range (IQR) is the difference between the upper and lower quartiles; also, it is the “height” of the box in a boxplot [56]. The coefficient of quartile variation (CQV) equals IQR divided by the sum of lower and upper quartiles [47]. Although IQR does not satisfy the criterion (viii), it is an *appropriate* statistic because it satisfies all of the other (more important) criteria; the *breakdown point* of the IQR is not critically low and equals 25% , together with the CQV for which the same reasoning of *appropriateness* applies. Furthermore, Rousseeuw-Croux estimators S_n and Q_n [57] offer breakdown points of 50% , do not assume distribution *symmetry*, and work independently of the choice of central tendency (*mean* or *median*)—all highly favourable traits. Notably, the median absolute deviation (MAD), as a robust measure of spread, was considered a serious contender due to its clear benefits, for example, over standard deviation as defined and elaborated in [50]. However, an important drawback of classical MAD with regard to criterion (vii) is its sensitivity to distribution asymmetry, a behaviour measurement score distribution definitely evident as shown in Figure 10. Therefore, IQR, CQV, S_n , and Q_n form a group of selected, *appropriate estimators of scale*.

To conclude evaluation of the algorithms, a summary of boxplot parameters (*measurements scores*) and appropriate

estimators is given in Table 4. Next to each estimator is the criterion which the estimator is attached to; criterion (E1) bears *one* and criterion (E2) bears *four* estimators altogether.

All of the *appropriate estimators* gave their output in the form of a single number (i.e., values in brackets); these numbers were compared, and algorithms *ranked* accordingly (for the criterion (E1), greater values are better (more is better); for the criterion (E2), the opposite is true—smaller values are better). Ranks reflect *true positions* with respect to each estimator’s output, respectively. Some ranks exhibit a “tie” (e.g., as with S_n), where three algorithms came in 2nd, and only one came in 1st.

4.4. Evaluation of the SmartSocial Algorithms: Conclusion Phase. The *last row* (evaluation rank) in Table 4 declares the final, total rankings of algorithms with respect to evaluation. The final rank was produced as an arithmetic mean of the ranking of evaluation criteria (E1 and E2), the ranks of which were produced as arithmetic means of the respective evaluators. SLOF is compared to LRA in bold. As with criterion (E1), SLOF reigns supreme over the other algorithms along with criterion (E2) as well. In other words, SLOF is the most accurate and precise algorithm of the four analysed SmartSocial Influence algorithms. Evaluation clearly demonstrates that SLOF exhibits significant improvements over its predecessor, the LRA, and provides an original scientific contribution.

SAOF shows a minor improvement, whereas SMOF shows no improvement in the overall rankings, while SAOF is more *accurate* and SMOF is more *precise* than LRA. An interesting notice is that they are ranked (throughout the criteria) very closely to LRA, lacking the demonstrative power of improvement as exhibited by SLOF.

It seems that SMOF would greatly benefit from increasing its *accuracy*, as its precision is already on par with that of LRA. Likewise, SAOF would greatly benefit from increasing its *precision*, as it is already more accurate than LRA. Nonetheless, future research and additional work are necessary to uncover as to why the algorithms rank as they do—and motivation in answering this question lies in further experimentation and auxiliary analysis which may very well shed some additional light on a potentially decisive answer.

5. Discussion

This section discusses the impact of the proposed methodology and possible implications of SLOF as the best-evaluated algorithm. But first, to avoid any misconceptions, let us explain what validation and evaluation *are*, and what they are *not*—in terms of their respective goals.

Validation proves that all of the four SmartSocial influence algorithms do *not* work by the sheer *guessing* of correct measurements. The alternative hypotheses may be either true, or false—one cannot reason as to *how much* the algorithms produce “correct, meaningful and truthful” results; only that they do not produce random results (as is the case with guessing), when compared against the ground truth or “golden standard.” Validity is proven by ignoring the “pair weights” (Δ_p) associated with each measurement and looking

TABLE 4: Summary of criteria ranks and evaluation conclusion.

Statistic or estimator	Criterion	SLOF	SAOF	SMOF	LRA
Average (mean) rank	Accuracy (E1)	1st (0.0358)	2nd (0.0271)	4th (0.0240)	3rd (0.0250)
IQR rank	Precision (E2)	1st (0.08)	3rd (0.14)	2nd (0.13)	2nd (0.13)
CQV rank	Precision (E2)	1st (2.0)	3rd (7.0)	2nd (2.6)	2nd (2.6)
Rousseeuw-Croux S_n rank	Precision (E2)	1st (0.0835)	2nd (0.1073)	2nd (0.1073)	2nd (0.1073)
Rousseeuw-Croux Q_n rank	Precision (E2)	1st (0.0885)	2nd (0.1106)	2nd (0.1106)	2nd (0.1106)
Evaluation rank	(E1&E2)	1st	2nd	4th	3rd

at the percentage of correct measurements, as opposed to incorrect measurements.

Evaluation proves that SLOF is the best-ranked algorithm according to a pre-given set of criteria—namely, *accuracy* and *precision*. For each algorithm, accuracy is calculated using the mean (average) measurement score (as an *estimator of location*), and precision is calculated using measurement score spread (or dispersion, using robust *estimators of scale*). The algorithm with the greatest accuracy and precision emerges as the winner.

Additionally, evaluation does not enable any kind of statistical inference—the goal of validation and evaluation is not generalizability. The experiment, by its very design, did not (representatively) sample a predetermined population (One might define the population as mostly those between 20 and 30 years of age, predominantly highly educated (mostly from Zagreb, Croatia), with university degrees in information technology, medicine, psychology, or sociology.); doing so would greatly lower the amount of Facebook friendships in a sample graph, making the job of comparing algorithms all the more difficult—which is exactly what the purpose of the evaluation was in the first place.

The definition of *social influence* has been from social psychology, which is reflected to a certain degree in the design of the algorithms. On the other hand, there is no guarantee as to how much *social influence* measured by the algorithms fits *social influence* as measured by social psychologists. In other words, social influence in the “digital” realm may or may not correspond to (or be associated with) with that in the “physical, real world”—it is solely a best-effort model of it [4, 34, 35].

An analysis was conducted on the age and number of Facebook friends totalling 361 SmartSocial Influence experiment participants (The SmartSocial influence experiment was conducted in the period from September 2014 until May 2015. A total of 465 user profiles were created. Of these, 104 contained only telecommunication data, as these users did not provide their Facebook data. Consequently, the SmartSocial real-world sample comprised the remaining 361 profiles with complete, personal multisource data necessary for SmartSocial algorithms to run—both Facebook and telecommunication personal data.) (these are not the same participants who participated in the evaluation questionnaire (The SmartSocial Influence evaluation questionnaire was conducted in the period from 21st February 2016 until 14th March 2016. The first phase (pre-questionnaire) had 22 experts and 22 nonexperts as the participants. The second

phase (main questionnaire) had 16 participants.) although some may overlap). Analysis of age draws some interesting conclusions (Figure 11). Up until SI of 61, there is a slowly rising trend of age with respect to the social influence scores of participants. However, as SI approaches (60, 70], there is a sharp increase in the age of the participants, as there is a much greater representation of 30-year-olds in the sample. More interestingly, highly influential participants (SI > 80) were all 25 years of age and younger, with the most influential ones (SI > 90) being below 21.5 years of age. According to SLOF, the youth is more socially influential.

What is most surprising is the results from analysing the number of friends (Figure 12). Once more, a group of participants with SI = (60, 70] shows specific characteristics. As observed with *age*, this group predominantly comprises those older than 30 years of age; they have the average number of friends that strongly correlated to *age*. The number of friends in all other groups of influencers equals a constant 475 to 575, while the 30-year-olds, of whom 50% are female, average 160 Facebook friends.

What follows are certain specifics of SLOF, the most truthful algorithm, with regard to the sample of experiment participants described in [4]. It is important to keep in mind that SI score groups do not hold an equal number of participants—this is easily observed in the SLOF distribution of SI scores [4]. A group of SI = (0, 10] contains as much as 65% of the participants; SI = 0 holds 11% and SI = (10, 20] holds 15% of the participants. The remaining 9% of participants altogether form a great minority with SI > 20. As is expected of a score such as SI, it follows a power law with a minority of participants being responsible for the majority of social influence. Therefore, no definitive conclusions regarding *gender*, *age*, or *number of friends* with respect to social influence on Facebook can be drawn; instead, a larger, more diverse *real-world sample of participants* is needed.

Comparing the specifics of SLOF to the state-of-the-art influence algorithm Klout would be noteworthy, but impossible as Klout has been a “black box” ever since official launch in 2008, meaning its proprietary method and processing details have been unknown and remain a secret. Only recently has Klout received attention from the scientific community with their paper outlining the principles and basic mechanism of calculating social influence combined with nine other SNSs [58]. The paper does not enable direct comparison of the Klout algorithm to SmartSocial Influence algorithms because (i) validation of Klout scores in the paper

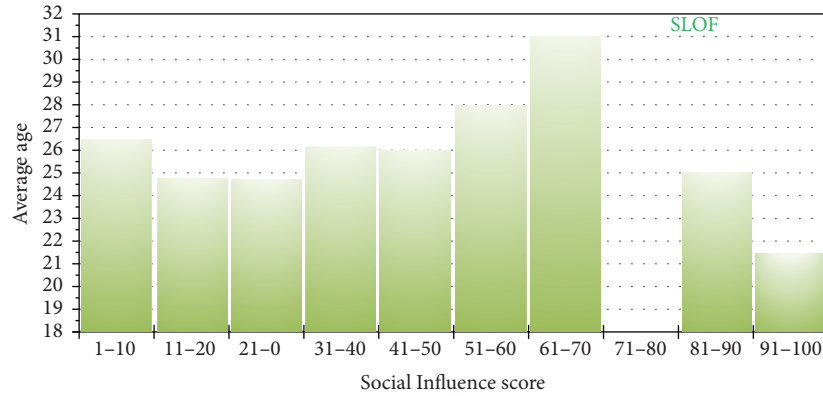


FIGURE 11: Average age with respect to SI scores of SLOF.

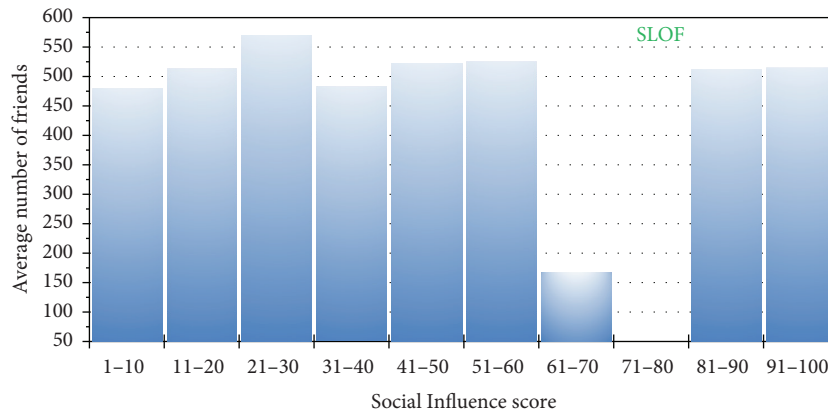


FIGURE 12: Average number of friends with respect to SI scores of SLOF.

is not as formal as the validation provided in this paper; (ii) validated scores include the top twenty people in specific categories (i.e., best ATP Tennis Players and Forbes Most Powerful Women); and (iii) it would be difficult to collect Klout scores of all 361 participants, since the Klout API as of 2017 does not yet enable fetching of Klout scores programmatically in a streamlined fashion. Klout's previous publications of Klout score distributions are obsolete due to several (major) revisions of the algorithm in the meantime. When taking everything into consideration, Klout is an impressive SNS for calculating social influence, but more transparency regarding the Klout algorithm is needed for a fair and direct comparison with alternative approaches.

6. Conclusion

This paper contributes to existing literature by proposing a new methodology for evaluating algorithms that calculate social influence in complex social networks. The paper has demonstrated the use of the proposed methodology using a case study in evaluating the accuracy and precision of four

social influence calculation algorithms from the class of SmartSocial Influence algorithms. The concept and details of SmartSocial Influence algorithms have already been presented in [4, 34, 35]; the proposed methodology validates all of them and has determined that the SmartSocial Influence algorithm (SLOF) is the most accurate and precise among them. This paper also contributes to existing literature by identifying the social influence calculation algorithm that offers higher accuracy and precision as benchmarked against the state-of-the-art LRA algorithm.

More broadly, the paper deals with a novel approach to social network user profiling with the goal of utilising multi-source, heterogeneous user data in order to infer new knowledge about users in terms of their social influence. By doing so, the paper addresses an ongoing research challenge in utilising such vast amounts of multisource, heterogeneous user data with the goal of identifying key, socially influential actors in the process of provisioning information and communication services. These actors are users equipped with smartphones, which reveals new information in regard to their social influence. This new information about a mobile

smartphone user has not only scientific but also industrial applications. For example, the best-evaluated novel algorithm for calculating a user's social influence (i.e., SLOF) can be used by telecommunication operators for churn prevention and prioritizing customer care, or by social networking services for digital advertising and marketing campaigns.

Some constraints in the proposed approach do exist. First, while the proposed methodology evaluates social influence algorithms, the question remains as to how to evaluate the very proposed methodology in return. To the authors' best knowledge, this approach is the first methodology to compare algorithms when calculating social influence based solely on available ego-user data rather than complete data on all social network users. That said, the authors of this paper will pursue encouragement of other similar research groups to develop alternative methodologies for evaluating algorithms that calculate social influence or more general global user properties, in online social networks. Second, the proposed methodology in this paper was applied on four algorithms from the SmartSocial Influence algorithm class. One of those—LRA—is a state-of-the-art benchmarking algorithm, while the other three—SLOF, SAOF, and SMOF—were previously developed by the authors of this paper. A more robust demonstration of the proposed methodology would include applying it on algorithms other than SmartSocial Influence class algorithms. This was not possible in this paper as the authors did not have access to (pseudo) code, test data, and ground truth data for other algorithms that solely use ego-user data for calculating ego-user social influence. However, they do hope that other research groups

developing such algorithms will apply the proposed methodology, presented in this paper, for benchmarking their algorithms against the SmartSocial Influence class of algorithms.

For future work, the authors plan to demonstrate applicability of the proposed evaluation methodology to other global user properties in complex social networks extending beyond social influence. Furthermore, they plan to adapt the methodology such that it is directly applicable to other social networks other than Facebook and other types of social network users beyond humans, such as networked objects and smart devices forming the Social Internet of Things.

Appendix

A. Questionnaires

The following questionnaires were developed and carried out using *Google Forms* (<https://docs.google.com/forms>). The content of the questionnaires below has been translated into English, as originally the questionnaires were given to participants in their native Croatian language.

B. Pre-Questionnaire (Content Validity)

This pre-questionnaire was given to 22 *experts* in the form of 30 questions (items); each item is scored between [1.0, 5.0], with the threshold for *passing* content validity >2.1 . Next to each question Q_i is its score $CV(Q_i)$. Questions marked as *chosen* are used for the next step (face validity pre-questionnaire).

TABLE 5: Pre-questionnaire (content validity) given to experts.

Q_i	Question text	Available answers	$CV(Q_i)$	Passed	Chosen
Q_1	You've noticed a post "Dangerous levels of chlorine detected in our hot water used for showering." A greater impression on you would leave a post by your Facebook friend: _____ or _____.	1 2 3 4 5	2.95	✓	
Q_2	You've noticed a post "If every one of us recycled, we would have CO2 emissions and receive state/country stimulus for it." You would recycle more frequently if it were posted by your Facebook friend: _____ or _____.	1 2 3 4 5	3.36	✓	
Q_3	You've noticed a post "Disaster has struck, the Nepalese are left with no food, water and electricity. I've donated money, here are instructions for you to do the same." You would donate a greater amount if it were posted by your Facebook friend: _____ or _____.	1 2 3 4 5	3.05	✓	

TABLE 5: Continued.

Q ₄	You are dissatisfied with your mobile operator. You've noticed a post "I've moved to my new telco X, I think they are better." You would more likely change your mobile operator if it were posted by your Facebook friend: _____ or _____	1 2 3 4 5	3.59	✓	✓✓
Q ₅	You've noticed a post "Gas station X has the best fuel." You would more likely refuel more frequently at the mentioned gas station if it were posted by your Facebook friend: _____ or _____	1 2 3 4 5	3.36	✓	
Q ₆	You've noticed a post "Smoking while pregnant greatly increases chances of health issues in a child." You would more likely spread this information if it were posted by your Facebook friend: _____ or _____	1 2 3 4 5	2.55	✓	
Q ₇	You've noticed a post "Electricity bills will soon go up." You would search for more details if it were posted by your Facebook friend: _____ or _____	1 2 3 4 5	3.50	✓	
Q ₈	You've noticed a post "Video out showing New Zealand's prime minister slipping on a banana." You would more likely watch the video if it were posted by your Facebook friend: _____ or _____	1 2 3 4 5	2.82	✓	
Q ₉	You've noticed a post "Hidden camera caught them in adultery." You would watch the video to a greater length if it were posted by your Facebook friend: _____ or _____	1 2 3 4 5	2.09	✗	
Q ₁₀	You've noticed a post "World leaders at their last meeting decided to increase nuclear armament." You would search for more details if it were posted by your Facebook friend: _____ or _____	1 2 3 4 5	3.68	✓	✓✓
Q ₁₁	You've noticed a post "Toothpaste X discounted in all shopping malls." You more likely go and buy the toothpaste if it were posted by your Facebook friend: _____ or _____	1 2 3 4 5	2.41	✓	
Q ₁₂	You've noticed a post "Travel agency X offers phenomenal discounts for Asia." You would search for more details if it were posted by your Facebook friend: _____ or _____	1 2 3 4 5	2.86	✓	
Q ₁₃	You've noticed a post "Concert tickets to see X selling out soon." You would search for more details if it were posted by your Facebook friend: _____ or _____	1 2 3 4 5	2.91	✓	
Q ₁₄	You've noticed a post "I'm calling everyone to join a public protest against getting rid of future generation pensions." You would more likely join this protest if it were posted by your Facebook friend: _____ or _____	1 2 3 4 5	4.0	✓	✓✓
Q ₁₅	You've noticed a post that explains the proven downside of your preferred political party. You would more likely change your vote if it were posted by your Facebook friend: _____ or _____	1 2 3 4 5	4.27	✓	✓✓
Q ₁₆	You've noticed a post "People killed in a terrorist attack in Ireland." You would search for more details if it were posted by your Facebook friend: _____ or _____	1 2 3 4 5	3.23	✓	
Q ₁₇	You've noticed a post "Home visits by National TV bill collectors more frequent in the following month." You would less likely open your doors to strangers if it were posted by your Facebook friend: _____ or _____	1 2 3 4 5	3.14	✓	
Q ₁₈	You've noticed a post "Quickly pay your monthly bills, otherwise fines follow within 24 hours according to latest news." Reading the post would to a greater extent cause restlessness if it were posted by your Facebook friend: _____ or _____	1 2 3 4 5	3.77	✓	✓✓
Q ₁₉	You've noticed a motivational post with thoughts about a brighter future, more jobs and possibilities, and greater salaries where you live. Reading the post would more likely cause peacefulness in you if it were posted by your Facebook friend: _____ or _____	1 2 3 4 5	4.0	✓	✓✓
Q ₂₀	You've noticed a post "Tensions between Balkan EU members might lead to war." Reading the post would more likely cause restlessness if it were posted by your Facebook friend: _____ or _____	1 2 3 4 5	3.64	✓	✓✓
Q ₂₁	You've noticed a post "Immigrants in EU constantly on the rise." Reading the post would more likely spark an interest if it were posted by your Facebook friend: _____ or _____	1 2 3 4 5	3.91	✓	✓✓
Q ₂₂	You've noticed a post "Parliament representatives physically confront each other at the morning sitting." Reading the post would more likely surprise you if it were posted by your Facebook friend: _____ or _____	1 2 3 4 5	2.68	✓	
Q ₂₃	You've noticed a motivational post about exercise, more physical activity, and health benefits. Reading the post would more likely motivate you if it were posted by your Facebook friend: _____ or _____	1 2 3 4 5	3.95	✓	✓✓
Q ₂₄	You are planning on seeing the movie X. You've noticed a post "I've seen X, it's horrible." Reading the post would more likely dissuade you from watching the film if it were posted by your Facebook friend: _____ or _____	1 2 3 4 5	4.41	✓	✓✓

TABLE 5: Continued.

Q ₂₅	You are planning a trip to a neighboring country/state X, it's snowing outside. You've noticed a post "X's police officers fine drivers without winter tires." Reading the post would more likely persuade you to buy winter tires if it were posted by your Facebook friend: _____ or _____	1 2 3 4 5	3.23	✓
Q ₂₆	You are driving on X section of road. You've noticed a post "Police radar-tracking speed at section X." Reading the post would more likely cause you to obey the speed limit if it were posted by your Facebook friend: _____ or _____"	1 2 3 4 5	3.0	✓
Q ₂₇	You are driving on the road section X. You've noticed a post "Terrible traffic accident at section X." Reading the post would more likely disturb you if it were posted by your Facebook friend: _____ or _____"	1 2 3 4 5	2.64	✓
Q ₂₈	You are planning a trip to city X. You've noticed a post "City X caught in bad weather." Reading the post would more likely worry/disappoint you if it were posted by your Facebook friend: _____ or _____	1 2 3 4 5	2.27	✓
Q ₂₉	You are planning a trip to city X. You've noticed a post "City X has seen a rise in crime-rates in recent years." Reading the post would more likely worry you if it were posted by your Facebook friend: _____ or _____	1 2 3 4 5	3.18	✓
Q ₃₀	You are planning a trip to city X. You've noticed a post "City X has seen a rise in crime-rates in recent years." Reading the post would more likely persuade you to re-plan the trip if it were posted by your Facebook friend: _____ or _____	1 2 3 4 5	3.09	✓

C. Pre-Questionnaire (Face Validity)

This pre-questionnaire was given to 22 *nonexperts* in form of 10 questions (items); each item is scored between [1.0, 5.0], with the top 5 best (plus one fixed) questions chosen for the main questionnaire. Next to each question Q_i is its score FV(Q_i).

D. Main Questionnaire (Algorithm Validity)

The main questionnaire was given to 16 *participants* with the goal of obtaining measurement scores for each algorithm, used in their validation and evaluation. The main questionnaire uses questions which "passed" both validities in pre-questionnaires; they are both content-valid and face-valid.

TABLE 6: Pre-questionnaire (face validity) given to nonexperts.

Q _i	Question text	Available answers	FV(Q _i)	Passed	Chosen
Q ₄	You are dissatisfied with your mobile operator. You've noticed a post "I've moved to my new telco X, I think they are better." You would more likely change your mobile operator if it were posted by your Facebook friend: _____ or _____	A or H B or G C or F D or E	3.32	✓	✓✓
Q ₁₀	You've noticed a post "World leaders at their last meeting decided to increase nuclear armament." You would search for more details if it were posted by your Facebook friend: _____ or _____	A or H B or G C or F D or E	3.73	✓	✓✓
Q ₁₄	You've noticed a post "I'm calling everyone to join a public protest against getting rid of future generation pensions." You would more likely join this protest if it were posted by your Facebook friend: _____ or _____	A or H B or G C or F D or E	3.45	✓	
Q ₁₅	You've noticed a post that explains the proven downside of your preferred political party. You would more likely change your vote if it were posted by your Facebook friend: _____ or _____	A or H B or G C or F D or E	3.86	✓	✓✓
Q ₁₈	You've noticed a post "Quickly pay your monthly bills, otherwise fines follow within 24 hours according to latest news." Reading the post would to a greater extent cause restlessness if it were posted by your Facebook friend: _____ or _____	A or H B or G C or F D or E	3.55	✓	
Q ₁₉	You've noticed a motivational post with thoughts about a brighter future, more jobs and possibilities, and greater salaries where you live. Reading the post would more likely calm you down if it were posted by your Facebook friend: _____ or _____	A or H B or G C or F D or E	3.59	✓	✓✓

TABLE 6: Continued.

Q ₂₀	You've noticed a post "Tensions between Balkan EU members might lead to war." Reading the post would more likely cause restlessness if it were posted by your Facebook friend: _____ or _____	<i>A or H B or G C or F D or E</i>	3.77	✓	✓✓
Q ₂₁	You've noticed a post "Immigrants in EU constantly on the rise." Reading the post would more likely spark an interest if it were posted by your Facebook friend: _____ or _____	<i>A or H B or G C or F D or E</i>	3.45	✓	
Q ₂₃	You've noticed a motivational post about exercise, more physical activity, and health benefits. Reading the post would more likely motivate you if it were posted by your Facebook friend: _____ or _____	<i>A or H B or G C or F D or E</i>	3.45	✓	
Q ₂₄	You are planning on seeing the movie X. You've noticed a post "I've seen X, it's horrible." Reading the post would more likely dissuade you from watching the film if it were posted by your Facebook friend: _____ or _____	<i>A or H B or G C or F D or E</i>	3.55	✓	✓✓
Q ₊	You have reached the final question. It is unique and very important. You will reply to it in the same manner as you replied to the questions earlier. Once more, you will choose a Facebook friend from the 4 pairs offered in answers—only this time, pay attention to the definitions of social influence below: - social influence is a measure of how people, directly or indirectly, affect the thoughts, feelings, and actions of others; - social influence is the ability to drive action; and - social influence occurs when one's emotions, opinions, and behaviours are affected by others. Read the definitions of social influence above. For each of the pairs, choose the Facebook friend whom you consider has the GREATER social influence on you on Facebook. While doing so, try to encompass all 3 definitions above as best as you can.	<i>A or H B or G C or F D or E</i>	—	—	—

TABLE 7: Main questionnaire given to participants.

The questionnaire contains 6 questions and takes 10 minutes to complete. Your answers will be anonymized and analysed collectively for all publishing or discussion purposes. By participating, you are supporting the final phases of Vanja Smailović's PhD research. Each of the 6 questions requires reading an imaginary Facebook post. Each question offers several *pairs* of your Facebook friends which are offered as answers. Your task, for each of the pairs, is to choose the Facebook friend which you consider to be the correct answer for a given question. If the question seems absurd or inapplicable, choose the Facebook friend whom you consider to be MORE correct. Read all the questions in advance—it is advisable to at least skim through them all before proceeding. *Important:* If the pairs repeat among the 6 questions—choose your answer always while paying attention to the *question*. On the other hand, watch out for pairs that repeat in a *single* question—those pairs require the same answer, because their goal is to check consistency. In other words, the same pairs between *different* questions are allowed to (and can) have a different answer—same pairs within a *single question cannot!* Do not communicate or consult with others while filling out the questionnaire and remain concentrated. You are allowed to go back in steps—the questionnaire is finalized, submitted, and locked only after you press Submit. In case of any questions or doubts, please call Vanja at [telephone number provided] to avoid making mistakes or errors.

Q _i	Question text	Available answers
Q ₄	You are dissatisfied with your mobile operator. You've noticed a post "I've moved to my new telco X, I think they are better." You would more likely change your mobile operator if it were posted by your Facebook friend: _____ or _____	12 Facebook-friend pairs
Q ₁₀	You've noticed a post "World leaders at their last meeting decided to increase nuclear armament." You would search for more details if it were posted by your Facebook friend: _____ or _____	12 Facebook-friend pairs
Q ₁₅	You've noticed a post that explains the proven downside of your preferred political party. You would more likely change your vote if it were posted by your Facebook friend: _____ or _____	12 Facebook-friend pairs
Q ₁₉	You've noticed a motivational post with thoughts about a brighter future, more jobs and possibilities, and greater salaries where you live. Reading the post would more likely cause peacefulness in you if it were posted by your Facebook friend: _____ or _____	12 Facebook-friend pairs
Q ₂₀	You've noticed a post "Tensions between Balkan EU members might lead to war." Reading the post would more likely cause restlessness if it were posted by your Facebook friend: _____ or _____	12 Facebook-friend pairs
Q ₂₄	You are planning on seeing the movie X. You've noticed a post "I've seen X, it's horrible." Reading the post would more likely dissuade you from watching the film if it were posted by your Facebook friend: _____ or _____	12 Facebook-friend pairs

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this article.

Acknowledgments

The authors acknowledge the support of research projects “Managing Trust and Coordinating Interactions in Smart Networks of People, Machines and Organizations,” funded by the Croatian Science Foundation under the Grant UIP-11-2013-8813; “Ericsson Context-Aware Social Networking for Mobile Media,” funded by the Unity through Knowledge Fund; and “A Platform for Context-aware Social Networking of Mobile Users,” funded by Ericsson Nikola Tesla. This research has also been partly supported by the European Regional Development Fund under the Grant KK.01.1.1.01.0009 (DATACROSS). Furthermore, the authors would like to thank all participants who provided their personal data by installing the SmartSocial Android application and participating in questionnaires by which they greatly contributed to this research.

Supplementary Materials

The paper is supplemented with the Excel file named “Smart-Social Influence evaluation dataset (anonymised),” which contains detailed evaluation results. Data in the file is anonymised to assure the privacy of individuals who participated in the evaluation. This dataset is also available at <http://socialab.science/datasets>. (*Supplementary Materials*)

References

- [1] Statista August 2017 <http://www.statista.com>.
- [2] M. Pticek, V. Podobnik, and G. Jezic, “Beyond the internet of things: the social networking of machines,” *International Journal of Distributed Sensor Networks*, vol. 12, no. 6, Article ID 8178417, 2016.
- [3] Y. Liu, D. Pi, and L. Cui, “Mining community-level influence in microblogging network: a case study on Sina Weibo,” *Complexity*, vol. 2017, Article ID 4783159, 16 pages, 2017.
- [4] V. Smailovic and V. Podobnik, “Mining social networks for calculation of SmartSocial Influence,” *Journal of Universal Computer Science*, vol. 22, no. 3, pp. 394–415, 2016.
- [5] Q. Wang, X. Yu, and X. Zhang, “A connectionist model-based approach to centrality discovery in social networks,” in *Behavior and Social Computing*, L. Cao, H. Motoda, J. Srivastava, E.-P. Lim, I. King, P. S. Yu, W. Nejdl, G. Xu, G. Li, and Y. Zhang, Eds., vol. 8178 of Lecture Notes in Computer Science, Springer, Cham, 2013.
- [6] C. Kiss and M. Bichler, “Identification of influencers - measuring influence in customer networks,” *Decision Support Systems*, vol. 46, no. 1, pp. 233–253, 2008.
- [7] J. Golbeck and J. Hendler, “Inferring binary trust relationships in web-based social networks,” *ACM Transactions on Internet Technology*, vol. 6, no. 4, pp. 497–529, 2006.
- [8] V. Podobnik, D. Striga, A. Jandras, and I. Lovrek, “How to calculate trust between social network users?,” in *SoftCOM 2012, 20th International Conference on Software, Telecommunications and Computer Networks*, Split, Croatia, September 2012.
- [9] M. Stupalo, J. Ilić, L. Humski, Z. Skočir, D. Pintar, and M. Vranić, “Applying the binary classification methods for discovering the best friends on an online social network,” in *2017 14th International Conference on Telecommunications (ConTEL)*, pp. 155–162, Zagreb, Croatia, June 2017.
- [10] J. Ilic, L. Humski, D. Pintar, M. Vranic, and Z. Kocir, “Proof of concept for comparison and classification of online social network friends based on tie strength calculation model,” in *Proceedings ICIST 2016. Belgrade, Serbia: Society for Information Systems and computer networks*, pp. 159–164, Belgrade, Serbia, 2016.
- [11] J. Ugander, B. Karrer, L. Backstrom, and C. Marlow, *The Anatomy of the Facebook Social Graph*, 2011, <https://arxiv.org/abs/1111.4503>.
- [12] M. Gabielkov, A. Rao, and A. Legout, “Studying social networks at scale: macroscopic anatomy of the twitter social graph,” in *The 2014 ACM International Conference on Measurement and Modelling of Computer Systems*, pp. 277–288, ACM, Austin, USA.
- [13] S. Bhagat, M. Burke, C. Diuk, I. O. Filiz, and S. Edunov, “Three and a half degrees of separation,” 2016, November 2017, <https://research.fb.com/three-and-a-half-degrees-of-separation>.
- [14] A. Landherr, B. Friedl, and J. Heidemann, “A critical review of centrality measures in social networks,” *Business & Information Systems Engineering*, vol. 2, no. 6, pp. 371–385, 2010.
- [15] J. C. Turner, *Social Influence*, Brooks/Cole, Pacific Grove, Calif, 1st ed edition, 1991.
- [16] Klout, Inc., “Klout|Be Known For What You Love,” Klout,” 2016, July 2017, <https://klout.com/home>.
- [17] T. L. D. Kred, “Kred|The Home of Influence,” 2017, July 2017, <http://home.kred>.
- [18] Runtime Collective Limited (Brandwatch), “Peer Index: Social Analytics & Influence Tools-Brandwatch,” 2017, July 2017, <https://www.brandwatch.com/peerindex-and-brandwatch..>
- [19] Tellagence, “Tellagence,” 2017, July 2017, <http://www.tellagence.com>.
- [20] Openinfluence.Com, “Creative + Data-Driven Influencer Marketing Services,” previously instabrand.com,” 2017, August 2017, <http://www.openinfluence.com>.
- [21] Klout, Inc., “Klout Score,” Klout,” 2017, August 2017, <https://klout.com/corp/score>.
- [22] M. Deutsch and H. B. Gerard, “A study of normative and informational social influences upon individual judgment,” *The Journal of Abnormal and Social Psychology*, vol. 51, no. 3, pp. 629–636, 1955.
- [23] S. E. Asch, “Opinions and social pressure,” in *Scientific American*, vol. 193, pp. 31–35, W. H. Freeman and Co., San Francisco, California, 1955.
- [24] S. Milgram, “Behavioral study of obedience,” *The Journal of Abnormal and Social Psychology*, vol. 67, no. 4, pp. 371–378, 1963.
- [25] C. Haney, C. Banks, and P. Zimbardo, “Interpersonal dynamics in a simulated prison,” *International Journal of Criminology and Penology*, vol. 1, pp. 69–97, 1973.
- [26] B. Latané, “The psychology of social impact,” *American Psychologist*, vol. 36, no. 4, pp. 343–356, 1981.
- [27] R. B. Cialdini, *Influence: Science and Practice*, Allyn & Bacon, Boston, MA, 4 edition edition, 2000.
- [28] M. J. Rosenfeld and R. J. Thomas, “Searching for a mate: the rise of the internet as a social intermediary,” *American Sociological Review*, vol. 77, no. 4, pp. 523–547, 2012.

- [29] M. Fink and J. Spoerhase, "Maximum betweenness centrality: approximability and tractable cases," August 2010, <https://arxiv.org/abs/1008.3503>.
- [30] K.-I. Goh, E. Oh, B. Kahng, and D. Kim, "Betweenness centrality correlation in social networks," *Physical Review E*, vol. 67, no. 1, article 017101, 2003.
- [31] S. P. Borgatti, "Centrality and network flow," *Social Networks*, vol. 27, no. 1, pp. 55–71, 2005.
- [32] M. Cha, H. Haddadi, F. Benevenuto, and K. P. Gummadi, "Measuring user influence in twitter: the million follower fallacy," in *Fourth International AAAI Conference on Weblogs and Social media*, Washington, DC, USA, 2010.
- [33] A. A. Rad and M. Benyoucef, "Towards detecting influential users in social networks," in *E-Technologies: Transformation in a Connected World*, G. Babin, K. Stanoevska-Slabeva, and P. Kropf, Eds., pp. 227–240, Springer Berlin Heidelberg, 2011.
- [34] V. Smailovic, D. Striga, D.-P. Mamic, and V. Podobnik, "Calculating user's social influence through the smart social platform," in *Proceedings of Software, Telecommunications and Computer Networks Conference (Soft COM)*, pp. 383–387, Split, Croatia, 2014.
- [35] V. Smailovic, D. Striga, and V. Podobnik, "Advanced user profiles for the smart social platform: reasoning upon multi-source user data," in *ICT Innovations 2014 Web Proceedings*, Ohrid, Macedonia, 2014.
- [36] American Educational Research Association (AERA), American Psychological Association (APA), and National Council on Measurement in Education (NCME), *The Standards for Educational and Psychological Testing*, American Educational Research Association, Washington, DC, USA, 2014.
- [37] C. H. Lawshe, "A quantitative approach to content validity," *Personnel Psychology*, vol. 28, no. 4, pp. 563–575, 1975.
- [38] F. J. Gravetter and L.-A. B. Forzano, *Research Methods for the Behavioral Sciences*, Wadsworth, Australia; Belmont, CA, 4th edition, 2012.
- [39] I. B. Weiner and W. E. Craighead, Eds., *The Corsini Encyclopedia of Psychology*, Wiley, Hoboken, NJ, 4th edition, 2010.
- [40] R. McCarney, J. Warner, S. Iliffe, R. van Haselen, M. Griffin, and P. Fisher, "The Hawthorne effect: a randomised, controlled trial," *BMC Medical Research Methodology*, vol. 7, no. 1, p. 30, 2007.
- [41] D. L. Sackett, "Bias in analytic research," *Journal of Chronic Diseases*, vol. 32, no. 1-2, pp. 51–63, 1979.
- [42] R. Rosenthal and R. L. Rosnow, *Artifacts in Behavioral Research Robert Rosenthal and Ralph L. Rosnow's Classic Books: A Re-Issue of Artifact in Behavioral Research, Experimenter Effects in Behavioral Research and the Volunteer Subject*, Oxford University Press, New York, 2009.
- [43] A. A. J. Marley and J. J. Louviere, "Some probabilistic models of best, worst, and best-worst choices," *Journal of Mathematical Psychology*, vol. 49, no. 6, pp. 464–480, 2005.
- [44] J. J. Louviere, T. N. Flynn, and A. A. J. Marley, *Best-Worst Scaling: Theory, Methods and Applications*. Cambridge, Cambridge University Press, New York, 2015.
- [45] A. Papoulis, "Bernoulli trials," in *Probability, Random Variables, and Stochastic Processes*, pp. 57–63, McGraw-Hill, New York, 2nd edition, 1984.
- [46] R. A. Maronna and R. H. Zamar, "Robust estimates of location and dispersion for high-dimensional datasets," *Technometrics*, vol. 44, no. 4, pp. 307–317, 2002.
- [47] D. G. Bonett, "Confidence interval for a coefficient of quartile variation," *Computational Statistics & Data Analysis*, vol. 50, no. 11, pp. 2953–2957, 2006.
- [48] P. J. Rousseeuw and C. Croux, "Alternatives to the median absolute deviation," *Journal of the American Statistical Association*, vol. 88, no. 424, pp. 1273–1283, 1993.
- [49] P. J. Huber and E. Ronchetti, *Robust Statistics*, Wiley, Hoboken, N.J, USA, 2nd edition, 2009.
- [50] C. Leys, C. Ley, O. Klein, P. Bernard, and L. Licata, "Detecting outliers: do not use standard deviation around the mean, use absolute deviation around the median," *Journal of Experimental Social Psychology*, vol. 49, no. 4, pp. 764–766, 2013.
- [51] M. Krzywinski and N. Altman, "Points of significance: visualizing samples with box plots," *Nature Methods*, vol. 11, no. 2, pp. 119–120, 2014.
- [52] M. Spitzer, J. Wildenhain, J. Rappsilber, and M. Tyers, "BoxPlotR: a web tool for generation of box plots," *Nature Methods*, vol. 11, no. 2, pp. 121–122, 2014.
- [53] J. Masters, "Ozone Hole FAQ|Weather Underground," *Weather Underground*, 2011, March 2016, <http://www.wunderground.com/climate/holefaq.asp>.
- [54] C. Gourieroux and A. Monfort, *Statistics and Econometric Models, Vol. 1. Cambridge [England]*, Cambridge University Press, New York, NY, USA, 1995.
- [55] X. He, D. G. Simpson, and S. L. Portnoy, "Breakdown robustness of tests," *Journal of the American Statistical Association*, vol. 85, no. 410, pp. 446–452, 1990.
- [56] W. C. Navidi, *Statistics for Engineers and Scientists*, McGraw-Hill, New York, 3rd edition, 2011.
- [57] P. Wessa, *Free Statistics Software (version 1.1.23-r7)*, Office for Research Development and education, 2016, March 2016, <http://www.wessa.net>.
- [58] A. Rao, N. Spasojevic, Z. Li, and T. D. Souza, "Klout score: measuring influence across multiple social networks," 2015, <https://arxiv.org/abs/1510.08487>.

Research Article

Self-Adaptive K -Means Based on a Covering Algorithm

Yiwen Zhang ¹, Yuanyuan Zhou ¹, Xing Guo ¹, Jintao Wu,¹ Qiang He,² Xiao Liu ³,
and Yun Yang²

¹School of Computer Science and Technology, Anhui University, Hefei 230601, China

²School of Software and Electrical Engineering, Swinburne University of Technology, Melbourne, VIC 3122, Australia

³School of Information Technology, Deakin University, Melbourne, VIC 3125, Australia

Correspondence should be addressed to Xing Guo; guoxingahu@qq.com

Received 29 December 2017; Accepted 26 March 2018; Published 1 August 2018

Academic Editor: Xiuzhen Zhang

Copyright © 2018 Yiwen Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The K -means algorithm is one of the ten classic algorithms in the area of data mining and has been studied by researchers in numerous fields for a long time. However, the value of the clustering number k in the K -means algorithm is not always easy to be determined, and the selection of the initial centers is vulnerable to outliers. This paper proposes an improved K -means clustering algorithm called the covering K -means algorithm (C- K -means). The C- K -means algorithm can not only acquire efficient and accurate clustering results but also self-adaptively provide a reasonable numbers of clusters based on the data features. It includes two phases: the initialization of the covering algorithm (CA) and the Lloyd iteration of the K -means. The first phase executes the CA. CA self-organizes and recognizes the number of clusters k based on the similarities in the data, and it requires neither the number of clusters to be prespecified nor the initial centers to be manually selected. Therefore, it has a “blind” feature, that is, k is not preselected. The second phase performs the Lloyd iteration based on the results of the first phase. The C- K -means algorithm combines the advantages of CA and K -means. Experiments are carried out on the Spark platform, and the results verify the good scalability of the C- K -means algorithm. This algorithm can effectively solve the problem of large-scale data clustering. Extensive experiments on real data sets show that the accuracy and efficiency of the C- K -means algorithm outperforms the existing algorithms under both sequential and parallel conditions.

1. Introduction

The development of big data technologies, cloud computing, and the proliferation of data sources (social networks, Internet of Things, e-commerce, mobile apps, biological sequence databases, etc.) enables machines to handle more input data than human being could. Due to this dramatic increase in data, business organizations and researchers have become aware of the tremendous value the data contain. Researchers in the field of information technology have also recognized the enormous challenges these data bring. New technologies to handle these data, called big data, are required. Therefore, it is vital for researchers to choose suitable approaches to deal with big data and obtain valuable information from them. Recognizing valuable information in data requires the use of ideas from machine learning algorithms. Thus, big data analysis must combine

the techniques of data mining with those of machine learning. Clustering is one such method that is used in both fields. Clustering is a classic data mining method, and its goal is to divide datasets into multiple classes to maximize the similarities of the data points in each class and minimize the similarities between the classes. The cluster analysis method has been widely used in many fields of science and technology, such as modern statistics, bioinformatics, and social media analytics [1–5]. For example, clustering algorithms can be applied to social events to analyze big data to determine peoples’ opinions, such as predicting the winner of an election.

Based on the characteristics of different fields, researchers have proposed a variety of clustering types, which can be divided into several general categories, including hierarchy clustering, density-based clustering, graph theory-based clustering, grid-based clustering, model-based clustering, and

partitional clustering [1]. Each clustering type has its own style and optimization approaches. We focus on partitional clustering algorithms. The most popular algorithm is K -means [2, 3, 6, 7], which is one of the top ten clustering algorithms in data mining. The advantages of the K -means algorithm are its easy implementation and understanding, whereas its disadvantages are that the number of clusters k cannot be easily determined and the selection of the initial centers is easily disturbed by outliers, which has a significant impact on the final results [6]. Due to the simple iteration of the K -means algorithm, it has good scalability when dealing with big data and is easy to implement in parallel execution [8–10]. Researchers have proposed improved K -means algorithms to address the drawbacks of the K -means algorithm, and most of the improvements were made by optimizing the selection of the initial K -means centers [11–13]. Good initial centers can significantly affect the performance of the Lloyd iterations in terms of quality and convergence and eventually help the K -means algorithm to obtain the nearly optimal clustering results.

However, K -means and its improved algorithms still need to ascertain the number of clusters in advance and then determine the best data partitioning based on this parameter. However, the obtained results do not always represent the best data partitioning. To address these problems, this paper proposes a K -means clustering algorithm that is combined with an improved covering algorithm, which is called the C - K -means algorithm. Our improved covering-initialized algorithm has “blind” features. Without determining the number of clusters in advance, the algorithm can automatically identify the number of clusters based on the characteristics of the data and is independent of the initial centers. The C - K -means algorithm combines the advantages of the CA and K -means algorithms; it has both the “blind” characteristics of the CA and the advantages of fast, efficient, and accurate clustering of high dimensional data of the K -means algorithm. Moreover, CA is easy to implement in parallel and has good scalability. We implemented the parallel C - K -means clustering algorithm and baseline algorithms in the Spark environment. The experimental results showed that the proposed algorithm is suitable for solving the problems of large-scale and high-dimensional data clustering.

In particular, the major contributions of this paper are as follows:

- (1) We propose a covering-based initialization algorithm based on the quotient space theory with “blind” features. The initialization algorithm requires neither the number of clusters to be prespecified nor the initial centers to be manually selected. CA determines the appropriate number of clusters k and the k -specific initial centers quickly and adaptively.
- (2) The convergence algebra of the Lloyd iterations of the C - K -means clustering algorithm is much simpler than that of baseline algorithms.
- (3) The parallel implementation of C - K -means is much faster than parallel baseline algorithms.
- (4) Extensive experiments on real datasets show that the proposed C - K -means algorithm outperforms existing algorithms in both accuracy and efficiency under sequential and parallel conditions.

The remainder of this paper is organized as follows. Section 2 provides an overview of the related work. Section 3 gives an introduction to baseline algorithms and details of the C - K -means algorithm under both sequential and parallel conditions. Section 4 presents the experimental results and analysis, and Section 5 concludes the paper with future work identified.

2. Related Work

As a classic clustering algorithm, the K -means algorithm is widely used in the fields of database and data anomaly detection. Ordonez [14] implemented efficient K -means clustering algorithms at the top of a relational database management system (DBMS) for efficient SQL. They also implemented an efficient disk-based K -means application that takes into account the needs of the relational DBMS [15]. Efficient parallel clustering algorithms and implementation techniques are key to meet the scalability and performance requirements for scientific data analysis. Therefore, other researchers have proposed parallel implementation and applications of the K -means algorithm. Dhillon and Modha [16] proposed a parallel K -means clustering algorithm based on a message passing model, which utilized the inheritance of the K -means algorithm. Due to data parallelism, as the amount of data increases, the speedup and extendibility of the algorithm improve. Zhao et al. [8] implemented a K -means clustering algorithm based on MapReduce, which significantly improved the efficiency of the K -means algorithm. Jiang et al. [17] proposed a two-stage clustering algorithm to detect outliers. In the first stage, the algorithm used improves K -means to cluster the data. In the second stage, while searching for outliers in the clustering results of the first stage, it identifies the final outlier. Malkomes et al. [18] used the k -center clustering variant to handle noisy data, and the algorithms used are highly parallel. However, the selection of the initial center point of the K -means algorithm is easily disturbed by abnormal points, which has a significant impact on the final results. However, efficient methods to solve the issue in which the K -means algorithm is influenced by the initial centers have not been proposed.

Recently, scholars have focused on research into the issue that the selection of the initial centers of the K -means algorithm is easily disturbed by outlier points and have proposed several improved algorithms to help the K -means algorithm select the initial centers. The most classic improved algorithms are the K -means++ algorithm and the K -means|| algorithm. The K -means++ algorithm, which was proposed by Arthur and Vassilvitskii [12], helps the K -means algorithm to obtain the initial centers prior to the Lloyd iteration. It randomly selects a data point as the first cluster center, which is followed by selection based on the probability of the number of data points constituting the center point of

the initial set of k . The probability of selecting each successive center point is dependent on the previously selected cluster centers. However, due to the inherent sequential execution characteristics of K -means++, the k clustering centers must traverse the datasets k times and the current clustering center calculation depends on all of the previously obtained clustering centers, which makes the K -means++ initialization algorithm difficult to implement in parallel. Inspired by the K -means++ algorithm, Bahmani et al. [13] proposed the K -means|| algorithm to improve the performance of the parallelization and initialization phases. The K -means|| initialization algorithm introduces oversampling factors, obtains initial centers that are much larger than the value of k after a constant number of iterations, and assigns the weights to the center points. It then reclusters these weighted center points using the known clustering algorithm to obtain the final initial centers containing k points. K -means|| initialization has the advantages of the K -means++ algorithm and also addresses the drawback of K -means++ being difficult to extend. In follow-up research, researchers have proposed more improved algorithms of K -means and most are compared to these two classic improved algorithms. Cui et al. [10] proposed a new method of optimizing K -means based on MapReduce to process large-scale data, which eliminated the iterative dependence and reduced the computational complexity. Wei [19] improved the K -means++ algorithm by selecting the cluster centers using the sampling method in the K -means++ algorithm and then producing k centers with the expectation of having an approximately constant factor for the best clustering result. Newling and Fleuret [20] used the CLARANS to help K -means solve the problem of selecting k initial centers.

However, the number of clusters k in the K -means algorithm and its variations must be known in advance, and the best data division based on this parameter is then defined. The data division defined in this way is actually based on an imaginary model; it is not necessarily suitable for the best data division. In addition, the final clustering result is based on clustering under a hypothetical parameter without considering the actual structural relationship of the data.

In response to the problems described above, this paper presents a novel clustering algorithm called C- K -means that has both the “blind” feature of the CA and the fast, efficient clustering advantage of the K -means algorithm. It can be applied to high-dimensional data clustering with strong scalability. We implement the parallelized C- K -means algorithm on the Spark cloud platform. Extensive experimental results show that the C- K -means clustering algorithm is more accurate and efficient than the baseline algorithms.

3. The Algorithms

In this section, we first introduce the K -means clustering, K -means++ clustering, and K -means|| clustering algorithms. The motivation for using the CA as the initialization algorithm of the C- K -means clustering algorithm is then introduced, and the reason that the CA initialization can obtain clustering results that are approximately optimal is explained. Finally, we implement the parallel C- K -means

TABLE 1: Mathematical notations.

Symbol	Explanation
$X = \{x_1, \dots, x_n\}$	Denotes a set of points in the d -dimensional Euclidean space
$m = X $	Denotes that there are m data points in dataset X
k	Denotes a positive integer specifying the number of clusters
$C = \{c_1, \dots, c_k\}$	Denotes the set of cluster centers
$\ x_i - x_j\ $	Denotes the Euclidean distance between x_i and x_j
$Y \subseteq X$	Denotes that Y is a subset of X
$d(x, Y)$	Denotes the minimum Euclidean distance between x and set Y
centroid(Y)	Denotes the centroid of set Y
$\min_{y \in Y} \ x - y\ $	Denotes the minimum Euclidean distance between x and y in set Y
$\phi_Y(C)$	Denotes the cost of Y with respect to C
$\sum_{y \in Y} d^2(y, C)$	Denotes the sum of the squares of the minimum Euclidean distance between y in set Y and set C
$\sum_{y \in Y} \min_{i=1, \dots, k} \ y - c_i\ ^2$	Denotes the sum of the minimum Euclidean distances between y in set Y and set $C(c_i \in C)$
ϕ^*	Denotes the cost of optimal clustering algorithms
$\sigma_i = (\sigma_{1i}, \sigma_{2i}, \dots, \sigma_{ni})^T$	Denotes the standard deviation vector of a cluster

algorithm. Before explaining these questions, we summarize the notions used throughout this paper in Table 1.

3.1. State-of-the-Art Algorithms

3.1.1. K -Means. The K -means algorithm is one of the most classic clustering algorithms, because of its simple and fast performance, leading it to be widely-used. The description of the K -means algorithm is shown in Algorithm 1. First, we randomly select k data points from the original dataset X as the initial k cluster centers denoted by C , and we then calculate the distance between each data point x_i in X and each center in the initial centers C . Each data point can independently determine which center is closest to it, given an assignment of data points to clusters, the closest center is denoted by c_j . Then, the center of each cluster is updated, and each data point is repeatedly assigned to the cluster of the nearest center until the new set of cluster centers is equal to or less than the set of former cluster centers. This local search is called Lloyd iteration. The simple iteration of the K -means algorithm gives it good flexibility and can work effectively even with today’s big data. Algorithm 1 presents the pseudocode for the K -means algorithm [6, 12, 13].

3.1.2. K -Means++. Because the selection of the initial centers has a significant influence on the K -means clustering results,

<p>Input: X, θ Output: A set of clusters C_1, C_2, \dots Begin 1: $C \leftarrow$ sample k points uniformly at random from dataset X 2: $C_{new} \leftarrow C, C_{old} \leftarrow \phi$ 3: while $C_{new} - C_{old} \leq \theta$ do: 4: $C_{old} \leftarrow C_{new}$ 5: calculate all of the distances between x_i and C_{old_j}; $\text{get_distance}(x_i, C_{old_j}), x_i \in X, C_{old_j} \in C_{old}$ 6: assign x_i to the nearest C_{old_j} 7: calculate new centroid C_{new}: $C_{new_i} = \left(\frac{1}{ C_{old_i} } \sum_{i=1}^{C_{old_i}} x_i \right)$ 8: end while End</p>
--

ALGORITHM 1: (K -means algorithm).

<p>Input: X Output: Initial center set C Begin 1: $C \leftarrow$ sample a point uniformly at random from dataset X 2: while $C < k$ do: 3: sample $x \in X$ with probability $\frac{d^2(x, C)}{\phi_X(C)}$ 4: $C \leftarrow C \cup \{x\}$ 5: end while End</p>
--

ALGORITHM 2: (K -means++ initialization).

the K -means algorithm can only find a local optimal solution. To obtain the global optimal solution, it may be necessary to select the initial centers several times and then acquire the final values by constantly choosing these initial centers.

To overcome the disadvantages of K -means, researchers have proposed improved methods to help K -means find suitable initialization centers. K -means++, which was proposed by Arthur and Vassilvskii [12], is a typical representative algorithm (shown in Algorithm 2). The main idea of this algorithm is to select the initial centers one by one in a controlled way, and the calculation of the current cluster centers depends on all of the previously obtained cluster centers. Intuitively, the initialization algorithm selects relatively decentralized initial center points for K -means clustering, and the K -means++ initialization algorithm prioritizes the data points away from the previously selected centers when selecting a new clustering center. However, from the scalability point of view, the main disadvantage of K -means++ initialization is its inherent sequential execution properties. The acquisition of k centers must traverse the entire dataset k times, and the calculation of the current cluster center relies on all of the previously obtained clustering centers, which makes the algorithm not scalable in parallel and therefore greatly limits the applications of the algorithm to large-scale datasets. Algorithm 2 presents the pseudocode for the K -means++ algorithm [12].

3.1.3. K -Means||. Based on the advantages and disadvantages of the two initialization algorithms described above, researchers have proposed a new initialization algorithm called K -means|| [13] (see Algorithm 3 for details). The main idea of this algorithm is to change the sampling strategy during each traverse and propose an oversampling factor $l = \Omega(k)$. Each time the sample points are traversed in a nonuniform way and the sampling process is repeated for approximately $O(\log \psi)$ iterations, $O(\log \psi)$ is the clustering cost of the selected centers. We can then obtain the centers of $lO(\log \psi)$ sample points with repeated sampling. The number of intermediate centers is larger than k and much smaller than the original data size. Line 7 of Algorithm 3 shows that the center points in the set of center points C are assigned weights, and the center points of these weights are then reclustered in line 8, that is, the clustered k centers obtain the final k centers. Finally, these k points are fed into the Lloyd iteration as the initial centers. Algorithm 3 presents the pseudocode for the K -means|| algorithm [13].

3.2. *Intuition behind the Proposed Algorithm.* The traditional K -means random initialization method requires only one iteration and selects k centers uniformly and randomly. The K -means++ initialization method improves the method by randomly selecting the center point by selecting the initial center in a nonuniform way, but it requires k iterations. Only one data point is selected for each iteration to join the set of center points. Moreover, the selection of the current center point depends on the previously selected center. K -means++, which is a constantly updated nonuniform selection operation, increases the accuracy of K -means++ over random initialization, but it makes the K -means++ algorithm difficult to expand on a big dataset. Therefore, researchers proposed the K -means|| algorithm to improve the shortcomings of random initialization and K -means++ initialization and to choose k initial centers in a nonuniform manner with fewer iterations. However, both the K -means algorithm and its variant algorithms require the input of the clustering parameter k in advance and must define the best data partitioning for this parameter. However, the defined division of data is


```

Input:  $X$ 
Output: Initial center set  $C$ 
Begin
1:  $C \leftarrow$  sample a point uniformly at random from dataset  $X$ 
2:  $\psi \leftarrow \phi_X(C)$ 
3: for  $O(\log \psi)$  times do:
4:    $C' \leftarrow$  sample each point  $x \in X$  independently with probability  $p_x = \frac{l \cdot d^2(x, C)}{\phi_X(C)}$ 
5:    $C \leftarrow C \cup \{x\}$ 
6: end for
7: For  $x \in C$ , set  $w_x$  as the number of points in  $X$  that are closer to  $x$  than any other point in  $C$ 
8: Recluster the weighted points in  $C$  into  $k$  clusters
End

```

ALGORITHM 3: (K -means|| initialization).

actually based on a hypothetical value of k and may not be suitable for the best division of data, so the actual accuracy of the clustering results cannot be guaranteed.

Based on the geometric meaning of neural networks and the M-P neuron model, the covering algorithm was proposed by Zhang and Zhang [21]. It obtains a rule based on field covering and does not require the numbers of clusters and initial centroids to be prespecified. However, the traditional covering algorithm may face a problem in which some data points of the existing clusters are too large in the clustering process, which results in unreasonable clustering results. Therefore, based on the quotient space theory, we propose a covering algorithm called CA. The concept of granularity was first proposed by Zadeh in the 1970s [22], and Zhang and Zhang proposed the theory of quotient space [23]. This theory provided a reasonable formal model for mankind's ability to analyze and synthesize problems on a macroscopic and granular scale. Different granularities describe information at different levels. When the granularity is too small, all of the data points are self-formed and the inner knowledge cannot be mined. When the granularity is too coarse, all of the data are aggregated into a cluster, so some properties of the problems are obscured. Granularity is introduced to scientifically accomplish the task of covering clustering and obtain the optimal clustering results.

The CA requires neither the number of clusters to be prespecified nor the initial centers to be manually selected, and it automatically finds a set of fields that can separate samples with low similarity and merge samples with high similarity. The center of the set constitutes the initial clustering centers. Therefore, the CA has the beneficial feature of being "blind". Without knowing the number of clusters a priori, based on the relationships of the data, the CA can automatically identify the number of clusters and has no dependence on the initial clustering centers as well as fast computational speed. The CA also has good scalability. It is easy to implement in parallel, which is suitable for data processing in a big data environment. Therefore, this paper uses the improved CA as a K -means initialization algorithm to obtain the set of initial center points.

3.3. Overview of the C-K-Means Algorithm. In this section, we introduce the realization of the C- K -means clustering algorithm in detail. Figure 1 depicts the entire process of the C- K -means algorithm. The C- K -means algorithm is divided into two main phases: phase 1 and phase 2. Phase 1 performs the CA initialization, and phase 2 performs the Lloyd iterations. Next, we describe both phases in detail.

3.3.1. Phase 1: Overall Procedure of the CA. Algorithm 4 presents the pseudocode for the CA initialization. Below, we introduce the implementation process of CA in detail.

- (1). Find the center of gravity of all of the sample sets X that have not been clustered (covered) and then take the point denoted by center that is closest to the gravity as the initial center of the first cluster; this process is `get_center` (C_u) in Algorithm 4.
- (2). Find the distance r_x between each data point $x \in X$ and center that has not been clustered separately and obtain the sum of all of the distances denoted by $r_{X \rightarrow \text{center}}$. Next, $w_x = (r_x / r_{X \rightarrow \text{center}}) / (\sum_{x \in X} (r_x / r_{X \rightarrow \text{center}}))$ we set the weight $w_x = (r_x / r_{X \rightarrow \text{center}}) / (\sum_{x \in X} (r_x / r_{X \rightarrow \text{center}}))$ on all data points. Finally, we use r_x and w_x to calculate the covering radius, $\text{radius} = \sum_{x \in X} r_x w_x$, which is introduced in `get_weight_radius`(c, C_u) in Algorithm 4.
- (3). Find the centroids of the current spheres continually according to the obtained center and radius and obtain new clusters until the number of clusters in the data points does not increase. We can then determine the spheres (covering or clustering), which is introduced in `get_covering` (c, r, C_u) and lines 10 to 15 in Algorithm 4.
- (4). Repeat steps (1), (2), and (3) until all of the data points have been completely covered. This is introduced in lines 3 to 16 in Algorithm 4.

During the data clustering process, we can also automatically adjust the inner class and interclass relationships based

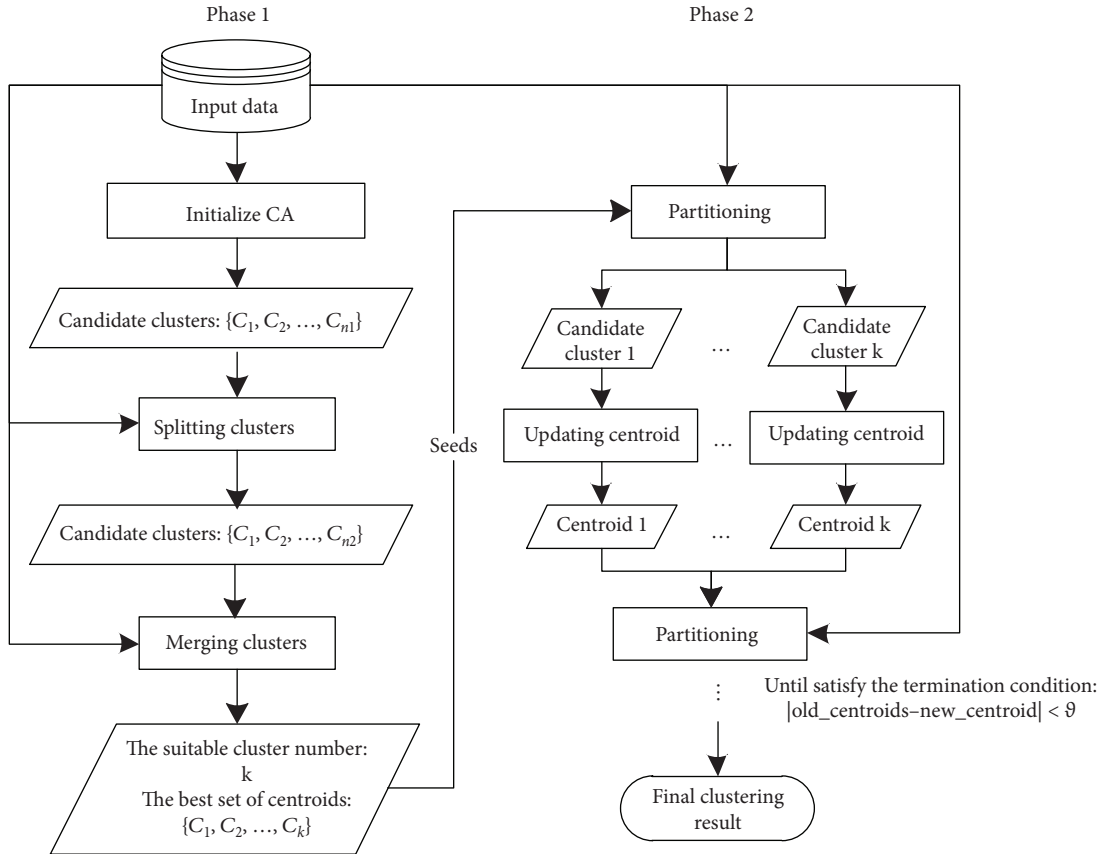


FIGURE 1: Overall procedure of the C-K-means algorithm.

Input: X
Output: Results of parallel covering with granularity analysis—A set of clusters $C = \{C_1, C_2, \dots\}$

Begin

- 1: center $c = \text{null}$
- 2: **Set** $C_u = X$
- 3: **do**
- 4: center $c \leftarrow \text{get_center}(C_u)$
- 5: radius $r \leftarrow \text{get_weight_radius}(c, C_u)$
- 6: Covering $C_{form} = \text{get_covering}(c, r, C_u)$
- 7: $c \leftarrow \text{get_centroid}(C_{form})$
- 8: $r \leftarrow \text{get_weight_radius}(c, C_u)$
- 9: Covering $C_{last} = \text{get_covering}(c, r, C_u)$
- 10: **while** $C_{last} \cdot \text{subtractByKey}(C_{form}) > 0$
- 11: $C_{form} \leftarrow C_{last}$
- 12: $c \leftarrow \text{get_centroid}(C_{form})$
- 13: $r \leftarrow \text{get_radius_centroid}(c, C_u)$
- 14: $C_{last} = \text{get_covering}(c, r, C_u)$
- 15: **end while**
- 16: **while** $(C_u \neq \emptyset)$
- 17: **Do** Split Operation
- 18: **Do** Merge Operation
- 19: **return** $C = \{C_1, C_2, \dots\}$

End

ALGORITHM 4: (CA initialization).

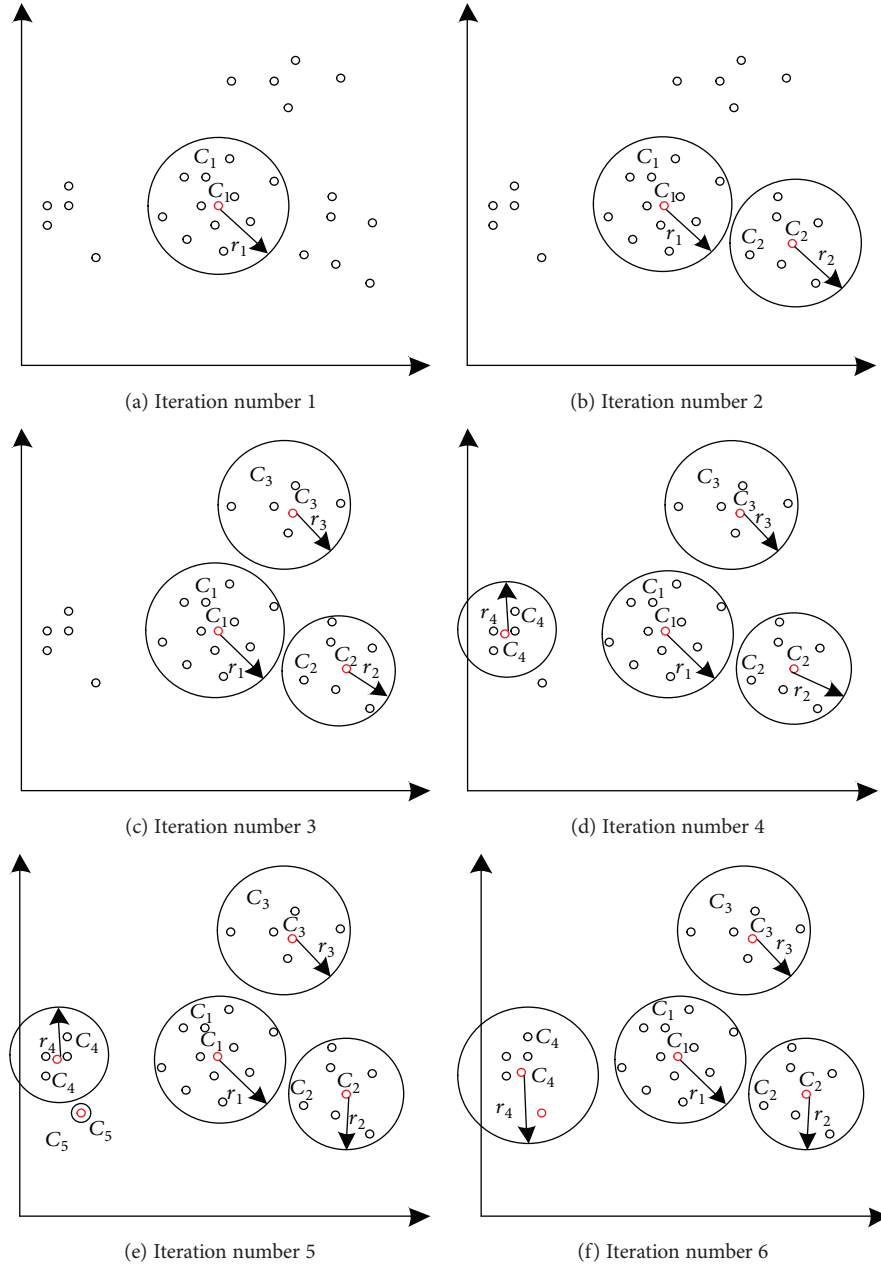


FIGURE 2: An example of clustering.

on the actual demand or the relationship between the data in the dataset. For a covering with fewer sample points, the single linkage method (using the Euclidean distance) in the hierarchical clustering algorithm [24, 25] is adopted to merge them to form an ellipsoidal domain, which means combing the most similar pair of clusters into a new cluster. Then, the similarities between the new cluster and the other clusters are updated, and the two most similar clusters are again merged. Based on the relationship between the data in the dataset or the actual demand, we can decide whether to continue merging the clusters with fewer data points or to split the spheres with more data points. Finally, we can obtain reasonable covering divisions with all of the similar data points that are distributed in one area (spherical or

ellipsoidal), which is introduced in lines 17 and 18 in Algorithm 4.

Figure 2 presents an illustrative example to intuitively demonstrate the clustering process of Algorithm 4. To cluster the data points, Algorithm 4 goes through five iterations to identify five clusters (covering or fields), C_1, \dots, C_5 . We then compute the relationship between the inner class and the interclasses and find that clusters C_4 and C_5 are very similar. Therefore, the sixth iteration merges them into one cluster and then updates the similarities between each cluster, where c_1, \dots, c_5 are the centers and r_1, \dots, r_5 are the radii, respectively.

When we study a dataset, we can divide it in different ways. Each division is a quotient space of different

granularities. We observe and analyze this dataset from different granularities. Based on the different granularities of the observation and analysis datasets, we can solve the problem in different granular worlds and can jump quickly from one granular world to another. This ability to handle different worlds of granularity is a powerful manifestation of the solution of human problems [26]. When we study the problem of reasonably clustered datasets, we can put the problem in the quotient space with different granularities for analysis. We can then obtain the solution to the clustering problem synthetically. In a different granularity quotient space, we can observe the different nature of the dataset and then find the properties of interest to the user, which can be maintained in different granular worlds or preserved to a certain extent. However, not every arbitrary division can achieve this goal. Therefore, the dataset division and its choice of granularity must be studied, that is, we need to select the appropriate dataset division. Based on the above, we propose the split-operation and merge-operation mechanisms in the C-K-means algorithm to help the datasets determine the appropriate partitioning and granularity. The C-K-means algorithm automatically adjusts the number of clusters during the iteration by merging similar clusters and splitting clusters with larger standard deviations. Finally, after a small number of constant iterations, C-K-means helps the dataset find the appropriate number of clusters k and k initial centers, and it then feeds the clustering centers into the Lloyd iteration to complete the final clustering process and determine the reasonable quotient space for the original dataset.

Adjustment Mechanism 1: Split Operation. First, we calculate the vector of the standard deviations for all of the samples in the cluster to the center of the cluster in all of the clusters: $\sigma_i = (\sigma_{1i}, \sigma_{2i}, \dots, \sigma_{ni})^T$, $i = 1, 2, \dots, N_c$, where N_c is the number of existing classes and n is the dimension of the samples. We then calculate the maximum component on $\sigma_{i \max}$ of the standard deviation vector σ_i of each class and determine the threshold value σ_s . For cluster C_u , we consider the following conditions: (1) the maximum component-wise standard deviation in the cluster, that is, $\max_{j=1, \dots, n} \sigma_{uj} > \sigma_s$; (2) the average distance between the samples in the cluster is greater than the overall average distance, that is, $\bar{d}_i > \bar{d}$, where \bar{d}_i and \bar{d} represent the average inner class distance of the i cluster (i.e., the average distance from the sample to the centroid in the calculation cluster) and the overall average distance (i.e., the overall average distance of each sample to its inner class centers), respectively; (3) the number of samples in the cluster is greater than θ_N , that is, $|C_u| > \theta_N$, where θ_N is the threshold cluster number, θ_N is the minimum number of samples allowed in each cluster (if less than this number, it cannot form a cluster), and $|C_u|$ denotes the number of samples in the i th cluster; and (4) the number of clusters is greater than $k/2$. If all of these conditions are satisfied, then split cluster C_u into two clusters with two cluster centers C_{u+} and C_{u-} and delete the original class C_i . The current number of clusters will increase by 1. The values of C_{u+} and C_{u-} are the components corresponding to $\sigma_{i \max}$ in the original C_u that to $\sigma_{i \max}$ are added to and subtracted from, respectively, while their components remain unchanged.

Adjustment Mechanism 2: Merge Operation. To sort the numbers of points contained in all clusters that have been formed, for clusters with fewer points, we calculate the similarity values between all other clusters and them: $S_{ij} = 1/1 + d_{ij}$, $i = 1, 2, \dots, N_{c-1}$ and $j = 1, 2, \dots, N_c$. To sort all of the obtained S_{ij} values according to the value of the final number of clusters k ; we merge the two clusters with the largest S_{ij} values and update the merged cluster centers. The current number of clusters will decrease by 1.

3.3.2. Phase 2: Overall Procedure of Lloyd's Iterations. Phase 1 determines the suitable value of k and k specific initial centers by performing the CA initialization. In phase 2, we assign the data points in the dataset to the cluster whose center is closest to the data point according to the cluster centers obtained in phase 1. We then update the class centers until the convergence condition is satisfied. All of the data are distributed to the cluster when the data point is closest to the cluster center, that is, the Lloyd iteration of the K-means clustering algorithm is completed, and the clustering results near the optimal clustering solution are obtained to complete the proposed C-K-means algorithm. Our CA initialization and final C-K-means algorithm can be easily parallelized, and we can rapidly complete the clustering operations.

3.4. Computational Complexity Analysis. This section discusses the computational complexity of the C-K-means algorithm with two phases. First, we analyze the computational complexity of the forming phase of C-K-means (i.e., CA initialization). In Algorithm 4, the computational complexity of line 5 is $O(m)$ because dataset X contains a maximum of m points. Similarly, the computational complexities of lines 5 and 6 are also $O(m)$, and those of lines 7, 8, and 9 are also $O(m)$ because the number of clusters is smaller than m . Lines 10–15 will be repeated until the data points in the cluster do not change. Lines 3–15 must also be repeated until all of the data points in X are covered, and the number of repetitions num_C is much smaller than m . In line 3, the radius of a cluster is the average distance between the center of the cluster and all of the data points that are not covered by any clusters. On average, each newly created cluster covers half of the uncovered data points, and the computational complexity is $O(\log m)$. In line 17, the computational complexity is $O(p)$ because there is a maximum of p clusters after the initial covering process. Similarly, the computational complexity of line 18 is $O(p)$ because there is a maximum of p clusters. The number of clusters is much smaller than m . Thus, the computational complexity of Algorithm 4 is $O(m) \times O(\log m) + O(p) = O(m \log m)$. We then introduce the second phase's computational complexity, which is the Lloyd iterations. The second phase performs num_iter iterations until the cluster centers do not change, so its computational complexity is $O(k * m * n * num_iter)$. The numbers of clusters k and iterations num_iter are much smaller than m . Therefore, the computational complexity of the C-K-means algorithm is $O(m) \times O(\log m) + O(p) + O(k * m * n * num_iter) = O(m \log m)$.

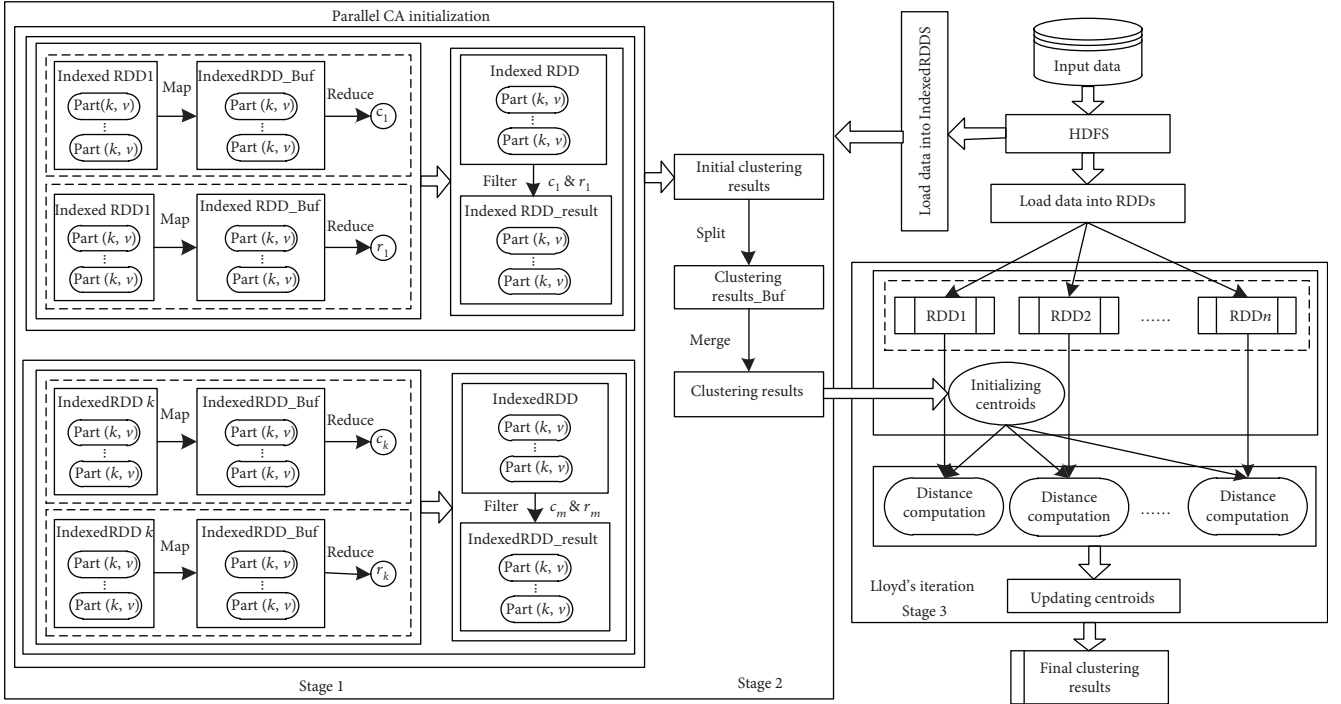


FIGURE 3: Overall procedure of the parallel C-K-means algorithm.

3.5. A Parallel Implementation. In this section, we discuss the proposed CA initialization and the parallel implementation of the C-K-means algorithm on Spark.

Spark is the de facto distributed computing platform for large data processing and is particularly suitable for iterative calculations. A main component of Spark is the *resilient distributed dataset* (RDD), which represents a read-only collection of objects partitioned across multiple machines that can be rebuilt if a partition is lost. Users can explicitly cache an RDD in memory across multiple machines and reuse it in multiple parallel operations. The RDD is the main reason that Spark is able to process big data efficiently. Due to the performance of memory computing, data locality, and transport optimization of Spark, it is particularly suitable for performing recursive operations on big data [27]. However, not all large-scale data can be efficiently processed via parallel implementation. Partitioning clustering algorithms require an exponential number of iterations [28]. Simultaneously, exponential job creation time and time of large-scale data shuffling are difficult to accept, especially for large amounts of data, so mere parallelism is not sufficient. High performance can be reached only by eliminating the partitioning clustering algorithm's dependence on the iteration.

The parallel implementation principle of the C-K-means clustering algorithm in Spark is illustrated in Figure 3. As demonstrated, C-K-means consists of three main stages. Stage 1 performs the parallel CA on Spark, and stage 2 analyzes the results of the initial covering clustering obtained from Stage 1 and splits or merges the clustering results through self-organization to determine the number of clusters k and the specific initial center set. Together, stages 1 and 2 constitute the parallel CA initialization process.

Stage 3 is the Lloyd iteration phase, in which Lloyd iteration is conducted on k initial centers to obtain the optimal clustering results.

The covering algorithm implemented on Spark is illustrated in stages 1 and 2 in Figure 3. The distributed files are read from Hadoop Distributed File System (HDFS) and transformed into IndexedRDD [29]. The parallel covering process in stage 1 consists of many covering processes. Each covering process comprises three processes, which obtain the cluster and its center, radius, and the cluster, respectively. Stage 1 describes the process for obtaining all of the clusters. We obtain the first cluster center c_1 through the *reduce* operation on Spark. This operation obtains the data point that is nearest to the centroid of all of the data in parallel. Next, we obtain the radius r_1 of the cluster through the *map* and *reduce* operations on Spark. Specifically, an intermediate variable IndexedRDD_Buf is obtained through the *map* operation on Spark. The *map* operation calculates the distance between the cluster center c_1 and each uncovered data point and forms IndexedRDD_Buf. Then, the radius r_1 is obtained through the *reduce* operation. This operation produces the radius r_1 by calculating IndexedRDD_Buf in parallel. Finally, we obtain *cluster_1* through the *filter* operation on Spark. Simultaneously, the *filter* operation filters the data points, where the distances between center c_1 and each uncovered data point are less than the radius r_1 . The radius and center are acquired using the process introduced above. The remaining clusters are obtained in a manner similar to the first cluster. These processes are repeated until no more data points can be identified, which indicates that all of the data points have been included in these clusters. This is the end of the covering process, which also indicates that stage 1 is complete. After the covering process, C-K-means

TABLE 2: Description of seven datasets.

Dataset	Number of attributes	Number of instances	Number of clusters (k)
Iris	4	150	3
Wine	13	178	3
Abalone	8	4177	29
Gauss	3	10,000	?
SPAM	57	4601	?
Cloud	10	1024	?
Individual household	7	2,049,280	?

performs the split and merge operations in stage 2 to obtain the final initialization centers. Through the CA initialization process, the initialization centers are adaptively obtained and then fed into Lloyd’s iteration in stage 3. As described earlier, Lloyd’s iteration can also be easily parallelized on Spark. Therefore, it is imperative that we implement an efficient CA initialization and C- K -means algorithm on the Spark platform.

4. Experimental Results

This section presents a detailed analysis and comparison of the experimental results, including sequential and parallel versions of the algorithm to confirm the merits of our C- K -means algorithm, which include the following: (1) the C- K -means algorithm can adaptively determine the number of clusters k and obtain a set of k cluster center points according to the similarity between the data, which then allows the C- K -means algorithm to obtain high-precision clustering results, (2) the C- K -means algorithm can obtain a clustering result that is near the optimal value which outperforms K -means in terms of its cost and is very similar to k -means++ and k -means||, and (3) compared with k -means++ and k -means||, the number of Lloyd’s iterations in the C- K -means algorithm is relatively small which converges quickly when accuracy and cost are ensured, meaning that the proposed C- K -means algorithm is accurate and efficient under parallel conditions.

In this paper, the C- K -means clustering algorithm and its counterparts are implemented sequentially and in parallel. The sequential implementation is evaluated on a stand-alone computer with a 6-core 3.60 GHz processor and 20 GB of memory. All of the parallel algorithms are implemented on a cluster of Spark 1.6 with Hadoop 2.6. The cluster has 16 nodes, each of which is an 8-core 3.60 GHz processor with 20 GB memory.

4.1. Datasets. We used 7 datasets in our experiments to evaluate the performance of the C- K -means algorithm. The summary statistics and information about these 7 datasets are shown in Table 2.

The question marks in Table 2 indicate that the number of clusters in the dataset is unknown.

Some of the datasets, such as Gauss, are synthetic, and the others are from real-world settings and are publicly available from the University of California Irvine (UCI) machine

learning datasets [30]. The Iris dataset [31–33] is a well-known database in clustering algorithm comparisons. It consists of three types of *Iris* plants (*setosa*, *versicolor*, and *virginica*) with 50 instances, each of which was measured with four features. The Wine dataset [31–33] is the result of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. It contains 178 instances measured with 13 continuous features. The Abalone dataset [34, 35] contains physical measurements of abalone shellfish. It contains 4177 instances with 9 features each (1 cluster label and 8 numeric and we apply 8 primary features), which are divided into 29 clusters. The age of an abalone can be determined by cutting the shell through the cone, staining it, and counting the number of rings with a microscope. In practice, measurements are used to estimate the age. The SPAM dataset [13] consists of 4601 instances with 57 dimensions and represents features available to an e-mail spam detection system. The Cloud dataset [12] consists of 1024 instances in 10 dimensions and represents the 1st cloud cover database. The individual household electric power consumption dataset [10] contains 2,049,280 instances with 9 features, 7 of which are applied in this paper because the other 2 are related to time, which are not applicable.

To effectively evaluate the experimental performance of the algorithm, we normalized the datasets. All of the algorithms use datasets that are normalized to frequent cases. When the dimension of the data points in a dataset is too high, it reduces the discrimination of the other dimensions with lower values during the clustering process. We normalized the datasets in an operation by

$$x_i^j = \begin{cases} \frac{x_i^{\max} - x_i^j(ori)}{x_i^{\max} - x_i^{\min}}, & \text{if } x_i^{\max} \neq x_i^{\min}, \\ 1, & \text{otherwise,} \end{cases} \quad (1)$$

where $x_i^j(ori)$ and x_i^j represent the j th dimension values of the i th data point in the dataset before and after normalization, respectively, and x_i^{\max} and x_i^{\min} are the maximum and minimum values of the j th dimension of all data points in the dataset, respectively.

4.2. Baselines. In the remainder of this paper, we assume that both the k -means++ and k -means|| initialization algorithms implicitly follow the Lloyd iteration process. The proposed C- K -means clustering algorithm outperforms the baseline algorithms as described below:

- (i) Traditional K -means algorithm (or K -means algorithm): this algorithm is based on random initialization and is often applied to randomly select k sample points as the initial centers for Lloyd’s iteration and complete the final clustering process accordingly (see Algorithm 1) [6].
- (ii) K -means++ algorithm: this method selects k centers as the initial centers for Lloyd’s iteration through multiple iterative processes. Based on the probability of each sample point, each iteration selects 1

sample point from the dataset to join the center set and completes the final clustering process (see Algorithm 2) [12].

- (iii) *K*-means|| algorithm: this method selects k centers as the initial centers for Lloyd's iteration through a constant number of processes. Based on the probability of each sample point, each iteration selects l sample points from the dataset to join the center set. It then reclusters the initial center set to obtain the final center set and feeds the final initial center point into Lloyd's iteration. The final clustering process is then completed (see Algorithm 3) [13].

4.3. Evaluation Metrics. The effectiveness of clustering is evaluated by numerous factors that determine the optimal number of clusters and the granularity of checking the clustering results. The evaluation of clustering results is often referred to as cluster validation, and researchers have proposed many measures of cluster validity. In this paper, we choose six standard validity measures to examine the soundness of the clustering algorithms, including Davies-Bouldins index (DBI) [10, 35, 36], the Dunn validity index (DVI) [36, 37], normalized mutual information (NMI) [38–40], the clustering cost function (ϕ), the Silhouette index (SI) [41, 42], and the SD index (SDI) [42]. These measures are described as follows:

$$\begin{aligned} \text{DBI} &= \frac{1}{k} \sum_{i=1}^k \max_{j \neq i} \left(\frac{\bar{C}_i + \bar{C}_j}{\|w_i - w_j\|_2} \right), \\ \text{DVI} &= \frac{\min_{0 < m \neq n \leq k} \left\{ \min_{\forall x_i \in \Omega_m, x_j \in \Omega_n} \{ \|x_i - x_j\| \} \right\}}{\max_{0 < m \leq k} \left\{ \max_{\forall x_i, x_j \in \Omega_m} \{ \|x_i - x_j\| \} \right\}}, \\ \text{NMI} &= \frac{I(X, Y)}{\sqrt{H(X)H(Y)}}, \\ \phi_Y(C) &= \sum_{y \in Y} d^2(y, C) = \sum_{y \in Y} \min_{i=1, \dots, k} \|y - c_i\|, \\ \text{SI} &= \sum_{0 < i \leq k} \frac{b(i) - a(i)}{\max \{a(i), b(i)\}}, \\ \text{SDI}(k) &= a \cdot \text{Scatt}(k) + \text{Dis}(k), \end{aligned} \quad (2)$$

where

$$\begin{aligned} \text{Scatt}(k) &= \frac{1}{k} \sum_{0 < i \leq k} \frac{\|\sigma(v_i)\|}{\|\sigma(X)\|}, \\ \text{Dis}(k) &= \frac{D_{\max}}{D_{\min}} \sum_{0 < i \leq k} \left[\sum_{0 < j \leq k} \|v_i - v_j\| \right]^{-1}. \end{aligned} \quad (3)$$

In the DBI validation measure, k denotes the number of clusters, \bar{C}_i denotes the average distance within the i th cluster, and $\|w_i - w_j\|$ denotes the distance between the i th cluster and the j th cluster. In the DVI validation measure, k denotes the number of clusters and $\|x_i, x_j\|$ denotes the

distance between two data points. In the NMI validation measure, X and Y denote the obtained cluster and true classes, respectively, where $I(X, Y)$ is the mutual information between X and Y and $H(X)$ and $H(Y)$ are the Shannon entropies of X and Y , respectively. The variables in the cost function ϕ are described in Table 1. In the SI validation measure, k denotes the number of clusters, $a_{(i)}$ denotes the average distance from the i th object to all of the objects in the same cluster, and $b_{(i)}$ denotes the minimum average distance from the i th object to all of the objects in a different cluster. In the SDI validation measure, k denotes the number of clusters, $\text{Scatt}(k)$ denotes the average scattering of the clusters, where $\sigma(v_i)$ denotes the variance of cluster i , $\sigma(X)$ denotes the variance of data set X , and $\text{Dis}(k)$ denotes the total separation between the clusters, where $D_{\max} = \max(\|v_i - v_j\|)$ denotes the maximum distance between the cluster centers, $D_{\min} = \min(\|v_i - v_j\|)$ denotes the minimum distance between cluster centers, and a denotes the weighting factor that is equal to $\text{Dis}(c_{\max})$, where c_{\max} is the maximum number of input clusters. DBI is a function of the ratio of the sum of the inner cluster distribution to the intercluster separation. The lower the DBI value is, the better the clustering performance will be because the distance within the clusters is small, but the distance among the clusters is large. DVI is a function of the ratio of the intercluster distribution separation to the sum of the inner cluster distributions. The larger the DVI value is, the better the clustering performance will be because the distance among the clusters is large and the distance within the clusters is small. NMI indicates the difference between the actual data type of the original data and the data type calculated by the clustering algorithm. Therefore, the NMI validation measure requires that the actual data type and the calculated data have the same number of class elements. The NMI values are in the interval $[0, 1]$, and a larger value means that the two clusters are very similar and also indicates a better clustering result. The value of the cost function ϕ indicates the sum of the distances from each data point to the nearest cluster center. Therefore, the lower the cost function ϕ is, the better the clustering performance will be. The purpose of SI is to calculate the average dissimilarity between points in the same cluster and a different cluster to describe the structure of the data. The SI values are in the interval $[-1, 1]$, and a larger SI value indicates a more optimal number of clusters in the dataset. The SDI is based on the average scattering of the clustering and the total separation of clusters. The minimum SDI value indicates that k is the optimal cluster number.

4.4. Determination of an Optimal Value of k in C-K-Means. CA self-organizes and recognizes the number of clusters k based on the similarities in the data without prior knowledge. By executing the CA algorithm, we can initially obtain the approximate number of clusters k . Next, we will conduct the split-operation and merge-operation mechanisms (see Section 3.3) to help the datasets determine the appropriate partitioning and granularity. To evaluate the resultant clusters for finding the optimal number of clusters, properties

TABLE 3: The value of k is known (Iris and Wine; * denotes the optimal value).

C-K-means	Iris				Wine			
	DBI	DVI	SI	SDI	DBI	DVI	SI	SDI
3	0.8280*	0.4958*	0.5043*	6.0402*	1.3702*	0.3683*	0.3013*	3.0225*
4	0.9792	0.3490	0.4435	6.8549	1.8091	0.2139	0.2313	3.7947
5	1.0775	0.2503	0.4100	9.4341	2.0714	0.2601	0.2055	4.0299
6	1.0612	0.2365	0.4304	10.3641	2.0543	0.2611	0.1996	4.2521
7	1.1013	0.4223	0.3416	10.1253	2.1805	0.2219	0.1259	4.7275
8	1.0926	0.4286	0.3281	10.2710	2.0461	0.2402	0.1284	5.0283
9	1.0649	0.4286	0.3200	11.1102	1.8996	0.2402	0.1337	5.1846

TABLE 4: The value of k is unknown (Cloud and Gauss; * denotes the optimal value).

C-K-means	Cloud				Gauss				
	DBI	DVI	SI	SDI	DBI	DVI	SI	SDI	
5	1.0479	0.2893	0.3611*	4.5859	9	1.1869	0.2672	0.2070	5.5891
6	1.0746	0.3469	0.3580	4.1325	10	1.1484	0.2945	0.2159	5.4769
7	1.0967	0.2935	0.3295	5.1985	11	1.1492	0.3121	0.2177	5.4715
8	1.0364	0.2748	0.3184	4.6350	12	1.1539	0.3375	0.2181	5.4793
9	1.0099	0.3428	0.3182	4.4439	13	1.1058*	0.3396*	0.2233*	5.4130*
10	0.9868	0.3516*	0.3160	4.0679*	14	1.1704	0.2788	0.2139	6.0448
11	1.0542	0.2739	0.2921	4.6985	15	1.1595	0.2230	0.2102	6.5191
12	1.0285	0.2708	0.2982	4.6481	16	1.2001	0.2535	0.2092	6.5085
13	1.0745	0.2482	0.2866	4.7405	17	1.1883	0.2697	0.2073	6.3310
14	0.9614*	0.2905	0.3036	4.4374	18	1.1653	0.2823	0.2080	6.3249

such as the cluster density, size, shape, and separability are typically examined by such as the DBI, DVI, SI, and SDI cluster validation indices. The clustering validity approach uses internal criteria to evaluate the results with respect to the features and quantities inherited from the data to determine how close the objects within the clusters are and the distances among the clusters.

Performing the CA on datasets Iris and Wine, the numbers of clusters are known (see Table 2). We initially obtain the approximate number of clusters 6 for the Iris dataset and 7 for the Wine dataset. We then conduct the split-operation and merge-operation mechanisms to get several numbers of clusters that close to 6 for the Iris dataset and 7 for the Wine dataset, respectively. The numbers of split operations are between 1 and 5 for both the Iris and Wine datasets. The numbers of merge operations need not be pre-given because they are determined by the numbers of clusters and split operations. To further evaluate the results, we choose the Cloud and Gauss datasets to execute the CA, in which the numbers of clusters are unknown (see Table 2). We initially obtain the approximate number of clusters 7 for the Cloud dataset and 13 for the Gauss dataset, respectively. Similarly, we then conduct the split-operation and merge-operation mechanisms to get several numbers of clusters that close to 7 for the Cloud dataset and 13 for the Gauss dataset, respectively. The numbers of split operations are between 0 and 6 for both the Cloud and Gauss datasets.

Table 3 shows a comparative analysis of the Iris and Wine datasets, using four validity measures. Because the numbers of clusters in the datasets are known, we can intuitively determine that the finite number k is obtained by our CA when most of the clustering indexes obtain the optimal value. Table 3 shows that 3 clusters are optimal on both datasets, which exactly match the actual numbers of clusters in the datasets. We used the results of the clusters from CA to check the performance of C-K-means in the Cloud and Gauss datasets and compared them to four existing validation indices. As shown in Table 4, the optimal validation indicators for the Cloud dataset are obtained with 10 clusters, thus the optimal cluster value is 10. For the Gauss dataset, each index shows that the optimal value is 13. The CA combined with split-operation and merge-operation mechanisms self-organizes and recognizes the reasonable number of clusters k based on the similarities in the data for any dataset.

4.5. Clustering Validation. Clustering validation is generally concerned with determining the optimal number of clusters and checking the suitability of the clustering results [10]. The evaluation of the clustering results is commonly referred to as cluster validation [10, 35, 43]. The accuracies of the baseline approaches and the C-K-means algorithm are measured in terms of three standard validity measures, namely DBI, DVI, and NMI, on datasets of different sizes. Other than the individual household dataset, the other datasets are small

TABLE 5: Accuracy comparison (Iris, Wine, and Abalone).

Algorithms	Dataset			
	Iris	Wine	Abalone	
<i>K</i> -means	DBI	0.9503	1.3970	1.1342
	DVI	0.0381	0.1378	0.0094
	NMI	0.656	0.8088	0.1697
<i>K</i> -means++	DBI	0.9220	1.3909	1.1309
	DVI	0.0577	0.1407	0.0106
	NMI	0.6737	0.8230	0.1706
<i>K</i> -means	DBI	0.8571	1.3903	1.1278
	DVI	0.0481	0.1393	0.0118
	NMI	0.7208	0.8268	0.1692
Covering	DBI	0.9608	1.4864	1.6721
	DVI	0.0860	0.1336	0.0073
	NMI	0.8342	0.7237	0.1594
C- <i>K</i> -means	DBI	0.8280	1.3702	1.1170
	DVI	0.0693	0.1893	0.0105
	NMI	0.7419	0.8529	0.1739

enough to be evaluated on a single machine. We compare the accuracies of C-*K*-means and the baseline approaches on the Iris, Wine, and Abalone datasets because the numbers of clusters and the labels to which the data belong are known in those datasets. The value of k is kept constant to effectively compare the C-*K*-means algorithm and the baseline algorithms. Using the split- and merge-operation mechanisms, the number of clusters of C-*K*-means is adjusted to be consistent with the number of clusters in the baseline algorithms. Table 5 shows a comparative analysis of the different approaches on the three datasets and the three validity measures. For the Iris and Wine datasets, the numbers of split operations are both 1. And for the Abalone dataset, the number of split operations is 8. To better verify the performance of the algorithms, we also choose the Gauss, SPAM, and Cloud datasets, the class categories of which are unknown for the experiments. To examine the soundness of our clusters, we discuss the DBI and DVI values of these three unknown data label datasets to those of C-*K*-means for moderate values of $k \in \{10, 20, 50\}$. For the Gauss dataset with different values of k , the numbers of split operations are 32, 4, and 10, respectively. For the SPAM dataset, the numbers of split operations are 10, 30, and 50, respectively. And for the Cloud datasets, the numbers of split operations are 6, 8, and 8, respectively. We also use other values of k and obtain similar results. The clustering results for C-*K*-means and the baseline approaches are listed in Table 6 for the Gauss dataset, Table 7 for the SPAM dataset, and Table 8 for the Cloud dataset. Obviously, the three tables show that the accuracies of proposed C-*K*-means are better than baseline approaches.

4.6. *Cost*. To evaluate the clustering cost of C-*K*-means, we compare it to the baseline approaches. We compare the cost of the SPAM and Gauss datasets to that of C-*K*-means for moderate values of $k \in \{20, 40, 50\}$. For the Gauss dataset

TABLE 6: Accuracy comparison (Gauss).

Gauss	$k = 10$		$k = 20$		$k = 50$	
	DBI	DVI	DBI	DVI	DBI	DVI
<i>K</i> -means	1.1511	0.0037	1.1620	0.0045	1.1121	0.0056
<i>K</i> -means++	1.1507	0.0051	1.1593	0.0056	1.1079	0.0061
<i>K</i> -means	1.1439	0.0049	1.1593	0.0055	1.1093	0.0065
C- <i>K</i> -means	1.1350	0.0061	1.1412	0.0053	1.1070	0.0081

with different values of k , the numbers of split operations are 4, 5, and 10, respectively. For the SPAM dataset, the numbers of split operations are 5, 4, and 4, respectively. The results of the Gauss and SPAM datasets are presented in Tables 9 and 10, respectively. For each algorithm, we list the cost of the solution at the end of the initialization step before Lloyd’s iteration as well as the final cost. In Tables 9 and 10, “seed” represents the cost after the initialization step and “final” represents the cost after the final Lloyd iteration. The initialization cost of C-*K*-means is similar to that of *K*-means|| and lower than that of *K*-means++. These results suggest that the centers produced by C-*K*-means, like those produced by *K*-means||, are able to avoid outliers. In addition, C-*K*-means guarantees high precision with high efficiency because CA runs very fast.

4.7. *Computational Time*. The individual household dataset is sufficiently large for large values of $k \in \{100, 200, 500\}$. We now consider the parallel algorithms for the individual household dataset. For the household dataset with corresponding values of k , the numbers of split operations are 6, 9, and 7, respectively. C-*K*-means is faster than *K*-means, *K*-means++, and *K*-means|| when implemented in parallel. The running time of C-*K*-means consists of two components: the time required to generate the initial solution and the time required for Lloyd’s iteration to converge. The former is proportional to k . The latter is considered, and C-*K*-means is compared to the baseline approaches. Table 11 shows the total running time of the clustering algorithms. For some values of k , C-*K*-means runs much faster than *K*-means and *K*-means++. C-*K*-means runs much faster than *K*-means|| when $k \in \{100, 200\}$. However, when k is 500, the total running time of C-*K*-means is similar to that of *K*-means|| because C-*K*-means needs to split and merge many times to obtain the number of clusters, which means that the initialization occupied a large proportion of the total running time.

Next, an expected advantage of C-*K*-means is demonstrated; the initial solution discovered by C-*K*-means contributed to a faster convergence of Lloyd’s iteration. Table 12 shows the number of iterations required to reach convergence of Lloyd’s iteration for the Cloud dataset with different initializations. C-*K*-means typically requires fewer iterations than the baseline approaches to converge to a local optimal solution. The convergence times of the iteration for datasets of different dimensions are also evaluated, and the Gauss and SPAM datasets are selected to verify the performance of the proposed C-*K*-means algorithm. The graphical representations of the number of iterations required to reach

TABLE 7: Accuracy comparison (SPAM).

SPAM	$k = 10$		$k = 20$		$k = 50$	
	DBI	DVI	DBI	DVI	DBI	DVI
K -means	2.3282	0.0010	2.0349	0.0007	1.7914	1.3874e-5
K -means++	2.2716	0.0020	1.8876	0.0044	1.5760	0.0042
K -means	1.9924	0.0023	1.7733	0.0031	1.5152	6.7375 e-4
C - K -means	1.8906	0.0034	1.6713	0.0085	1.2744	0.0076

TABLE 8: Accuracy comparison (Cloud).

Cloud	$k = 10$		$k = 20$		$k = 50$	
	DBI	DVI	DBI	DVI	DBI	DVI
K -means	1.1736	0.0207	1.23	0.02	1.3303	0.0186
K -means++	1.1644	0.0258	1.1946	0.0288	1.1973	0.0325
K -means	1.1474	0.0233	1.1888	0.029	1.2163	0.0386
C - K -means	0.9863	0.0369	0.9592	0.0484	1.1637	0.0582

TABLE 9: Median cost (over 10 runs) on the Gauss dataset.

Gauss	$k = 20$		$k = 40$		$k = 50$	
	Seed	Final	Seed	Final	Seed	Final
K -means	—	0.0108	—	0.007	—	0.006
K -means++	0.0124	0.0107	0.0082	0.007	0.0071	0.006
K -means	0.0118	0.0107	0.0078	0.007	0.0067	0.006
C - K -means	0.0119	0.0108	0.0076	0.007	0.0067	0.006

TABLE 10: Median cost (over 10 runs) on the SPAM dataset.

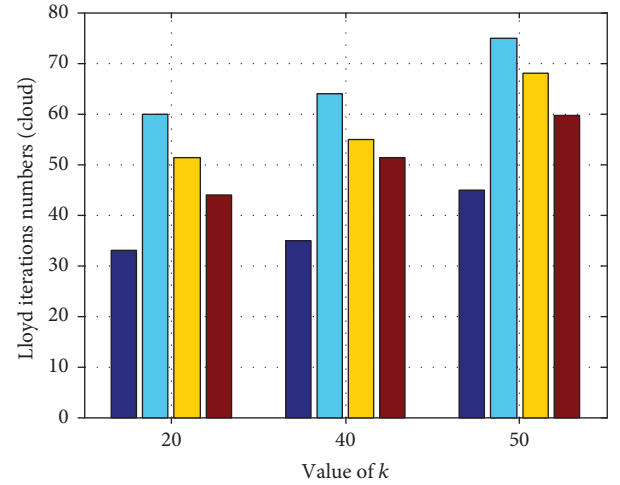
SPAM	$k = 20$		$k = 40$		$k = 50$	
	Seed	Final	Seed	Final	Seed	Final
K -means	—	0.1036	—	0.0771	—	0.071
K -means++	0.1136	0.0987	0.0886	0.076	0.08	0.0688
K -means	0.1098	0.0968	0.0846	0.0752	0.0765	0.0692
C - K -means	0.1022	0.0939	0.0861	0.0744	0.0788	0.0692

TABLE 11: Times (in minutes) for SPAM.

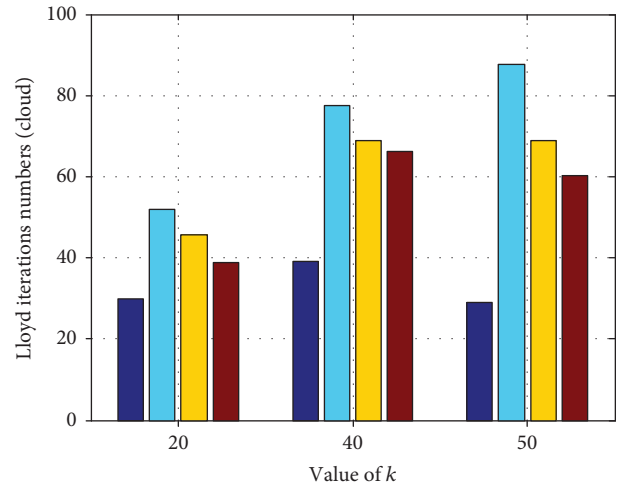
SPAM	$k = 100$	$k = 200$	$k = 500$
K -means	24.12	77.66	605.19
K -means++	31.66	63.64	153.00
K -means	28.11	41.96	94.98
C - K -means	16.38	24.78	99.42

TABLE 12: Numbers of Lloyd's iterations until convergence (averaged over 10 runs) for the Cloud dataset.

Cloud	$k = 10$	$k = 20$	$k = 50$
K -means	49	25.2	24.2
K -means++	30.4	23.2	19
K -means	28.2	21.8	18.6
C - K -means	13	16	12



(a) Gauss



(b) SPAM

FIGURE 4: Numbers of Lloyd's iterations until convergence (averaged over 10 runs).

convergence of Lloyd's iteration for datasets of several different dimensions with different initializations are shown in Figure 4(a) for the Gauss dataset (3 dimensions) and Figure 4(b) for the SPAM dataset (57 dimensions).

5. Conclusions and Future Work

This paper presents a covered K -means algorithm (C- K -means) that uses an improved covering algorithm (CA). First, based on the similarity between the data, the C- K -means algorithm uses the CA initialization to determine the number of clusters k and the specific cluster centers through self-organization. Because it is independent of the initial cluster centers, the CA is characterized as being "blind" without the need to have k prespecified. The K -means algorithm is then used to perform Lloyd's iteration on the k initial cluster centers determined by the CA until the cluster centers do not change, which means that the C- K -means clustering is complete, and the clustering results are close to optimal. In addition, a parallel implementation of C- K -means is performed on the Spark platform. Parallel computing is used to solve a large-scale data clustering problem and improve the efficiency of the C- K -means algorithm. A large number of experiments on real large-scale datasets demonstrated that the C- K -means algorithm significantly outperforms its counterparts under both sequential and parallel conditions. In future, we will optimize C- K -means and focus on the parameters that increase its speed and parallelism.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the National Key Technology R&D Program (no. 2015BAK24B01), the Natural Science Foundation of Anhui Province of China (no. 1808085MF197), and a Key Project of Nature Science Research for Universities of Anhui Province of China (no. KJ2016A038).

References

- [1] T. S. Madhulatha, "An overview on clustering methods," *IOSR Journal of Engineering*, vol. 2, no. 4, pp. 719–725, 2012.
- [2] M. Shindler, A. Wong, and A. W. Meyerson, "Fast and accurate k -means for large datasets," in *Proceedings of the 24th International Conference on Neural Information Processing Systems (NIPS'11)*, pp. 2375–2383, Granada, Spain, 2011.
- [3] M. Hajjar, G. Aldabbagh, N. Dimitriou, and M. Z. Win, "Hybrid clustering scheme for relaying in multi-cell LTE high user density networks," *IEEE Access*, vol. 5, pp. 4431–4438, 2017.
- [4] Q. Chen, X. Zhang, Y. Wan, J. Zobel, and K. Verspoor, "Sequence clustering methods and completeness of biological database search," in *Proceedings of the Workshop on Advances in Bioinformatics and Artificial Intelligence: Bridging the Gap*, pp. 8–14, Melbourne, VIC, Australia, 2017.
- [5] Q. Chen, Y. Wan, X. Zhang, Y. Lei, J. Zobel, and K. Verspoor, "Comparative analysis of sequence clustering methods for deduplication of biological databases," *Journal of Data and Information Quality*, vol. 9, no. 3, pp. 1–27, 2018.
- [6] L. Bottou and Y. Bengio, "Convergence properties of the k -means algorithms," in *Proceedings of the 7th International Conference on Neural Information Processing Systems (NIPS'94)*, pp. 585–592, Denver, CO, USA, 1995.
- [7] B. Castellani, R. Rajaram, J. Gunn, and F. Griffiths, "Cases, clusters, densities: modeling the nonlinear dynamics of complex health trajectories," *Complexity*, vol. 21, Supplement 1, p. 180, 2016.
- [8] W. Zhao, H. Ma, and Q. He, "Parallel K -means clustering based on MapReduce," in *Cloud Computing*, vol. 5931 of Lecture Notes in Computer Science, pp. 674–679, Springer, 2009.
- [9] Z. Tang, K. Liu, J. Xiao, L. Yang, and Z. Xiao, "A parallel K -means clustering algorithm based on redundancy elimination and extreme points optimization employing MapReduce," *Concurrency and Computation: Practice and Experience*, vol. 29, no. 20, 2017.
- [10] X. Cui, P. Zhu, X. Yang, K. Li, and C. Ji, "Optimized big data K -means clustering using MapReduce," *The Journal of Supercomputing*, vol. 70, no. 3, pp. 1249–1259, 2014.
- [11] R. Ostrovsky, Y. Rabani, L. J. Schulman, and C. Swamy, "The effectiveness of Lloyd-type methods for the k -means problem," *Journal of the ACM*, vol. 59, no. 6, pp. 1–22, 2012.
- [12] D. Arthur and S. Vassilvitskii, "k-means++: the advantages of careful seeding," in *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA '07)*, pp. 1027–1035, New Orleans, LA, USA, 2007.
- [13] B. Bahmani, B. Moseley, A. Vattani, R. Kumar, and S. Vassilvitskii, "Scalable k-means++," *Proceedings of the VLDB Endowment*, vol. 5, no. 7, pp. 622–633, 2012.
- [14] C. Ordonez, "Programming the K -means clustering algorithm in SQL," in *Proceedings of the 2004 ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '04*, pp. 823–828, Seattle, WA, USA, 2004.
- [15] C. Ordonez and E. Omiecinski, "Efficient disk-based k -means clustering for relational databases," *IEEE Transactions on Knowledge and Data Engineering*, vol. 16, no. 8, pp. 909–921, 2004.
- [16] I. S. Dhillon and D. S. Modha, "A data-clustering algorithm on distributed memory multiprocessors," in *Large-Scale Parallel Data Mining*, vol. 1759 of Lecture Notes in Computer Science, pp. 245–260, Springer, Berlin, Heidelberg, 2002.
- [17] M. F. Jiang, S. S. Tseng, and C. M. Su, "Two-phase clustering process for outliers detection," *Pattern Recognition Letters*, vol. 22, no. 6-7, pp. 691–700, 2001.
- [18] G. Malkomes, M. J. Kusner, W. Chen, K. Q. Weinberger, and B. Moseley, "Fast distributed k -center clustering with outliers on massive data," in *Proceedings of the 28th International Conference on Neural Information Processing Systems (NIPS'15)*, pp. 1063–1071, Montreal, QC, Canada, 2015.
- [19] D. Wei, "A constant-factor bi-criteria approximation guarantee for k -means++," in *Proceedings of the 30th International Conference on Neural Information Processing Systems (NIPS'16)*, pp. 604–612, Barcelona, Spain, 2016.
- [20] J. Newling and F. Fleuret, "K-medoids for k -means seeding," in *Proceedings of the International Conference on Neural Information Processing Systems (NIPS'17)*, pp. 5201–5209, Long Beach, CA, USA, 2017.

- [21] L. Zhang and B. Zhang, "A geometrical representation of McCulloch-Pitts neural model and its applications," *IEEE Transactions on Neural Networks*, vol. 10, no. 4, pp. 925–929, 1999.
- [22] L. A. Zadeh, "Some reflections on soft computing, granular computing and their roles in the conception, design and utilization of information/intelligent systems," *Soft Computing*, vol. 2, no. 1, pp. 23–25, 1998.
- [23] L. Zhang and B. Zhang, "The quotient space theory of problem solving," *Fundamenta Informaticae*, vol. 59, no. 2-3, pp. 287–298, 2004.
- [24] A. Roy and S. Pokutta, "Hierarchical clustering via spreading metrics," in *Proceedings of the 30th International Conference on Neural Information Processing Systems (NIPS'16)*, pp. 2316–2324, Barcelona, Spain, 2016.
- [25] R. Greenlaw and S. Kantabutra, "On the parallel complexity of hierarchical clustering and CC-complete problems," *Complexity*, vol. 14, no. 2, p. 28, 2008.
- [26] Y. Yao, "A triarchic theory of granular computing," *Granular Computing*, vol. 1, no. 2, pp. 145–157, 2016.
- [27] A. G. Shoro and T. R. Soomro, "Big data analysis: Apache Spark perspective," *Global Journal of Computer Science and Technology*, vol. 15, no. 1, pp. 9–14, 2015.
- [28] A. Vattani, "*k*-means requires exponentially many iterations even in the plane," *Discrete & Computational Geometry*, vol. 45, no. 4, pp. 596–616, 2011.
- [29] S. Peng, J. Sankaranarayanan, and H. Samet, "SPDO: high-throughput road distance computations on Spark using distance oracles," in *2016 IEEE 32nd International Conference on Data Engineering (ICDE)*, pp. 1239–1250, Helsinki, Finland, 2016.
- [30] A. Asuncion and D. Newman, "UCI machine learning repository," 2007, <http://www.ics.uci.edu/~mlern/MLRepository.html>.
- [31] P. Zhong and M. Fukushima, "Regularized nonsmooth Newton method for multi-class support vector machines," *Optimization Methods and Software*, vol. 22, no. 1, pp. 225–236, 2007.
- [32] B. Jia, B. Yu, Q. Wu et al., "Hybrid local diffusion maps and improved cuckoo search algorithm for multiclass dataset analysis," *Neurocomputing*, vol. 189, pp. 106–116, 2016.
- [33] Y. Wang, X. Duan, X. Liu, C. Wang, and Z. Li, "A spectral clustering method with semantic interpretation based on axiomatic fuzzy set theory," *Applied Soft Computing*, vol. 64, pp. 59–74, 2018.
- [34] N. Nouaouria and M. Boukadoum, "Improved global-best particle swarm optimization algorithm with mixed-attribute data classification capability," *Applied Soft Computing*, vol. 21, pp. 554–567, 2014.
- [35] N. Nouaouria, M. Boukadoum, and R. Proulx, "Particle swarm classification: a survey and positioning," *Pattern Recognition*, vol. 46, no. 7, pp. 2028–2044, 2013.
- [36] D. L. Davies and D. W. Bouldin, "A cluster separation measure," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-1, no. 2, pp. 224–227, 1979.
- [37] X. Ru, Z. Liu, Z. Huang, and W. Jiang, "Class discovery based on *k*-means clustering and perturbation analysis," in *2015 8th International Congress on Image and Signal Processing (CISP)*, pp. 1236–1240, Shenyang, China, 2015.
- [38] B. Wang, J. Yin, Q. Hua, Z. Wu, and J. Cao, "Parallelizing *k*-means-based clustering on Spark," in *2016 International Conference on Advanced Cloud and Big Data (CBD)*, pp. 31–36, Chengdu, China, 2016.
- [39] D. Lai and C. Nardini, "A corrected normalized mutual information for performance evaluation of community detection," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2016, no. 9, 2016.
- [40] A. Amelio and C. Pizzuti, "Correction for closeness: adjusting normalized mutual information measure for clustering comparison," *Computational Intelligence*, vol. 33, no. 3, pp. 579–601, 2017.
- [41] P. J. Rousseeuw, "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis," *Journal of Computational and Applied Mathematics*, vol. 20, pp. 53–65, 1987.
- [42] M. Hassani and T. Seidl, "Internal clustering evaluation of data streams," in *Trends and Applications in Knowledge Discovery and Data Mining*, vol. 9441 of Lecture Notes in Computer Science, pp. 198–209, Springer, 2015.
- [43] B. K. Mishra, A. Rath, N. R. Nayak, and S. Swain, "Far efficient *k*-means clustering algorithm," in *Proceedings of the International Conference on Advances in Computing, Communications and Informatics - ICACCI '12*, pp. 106–110, Chennai, India, 2012.

Research Article

Robust Semisupervised Nonnegative Local Coordinate Factorization for Data Representation

Wei Jiang ¹, Qian Lv,¹ Chenggang Yan,² Kewei Tang,¹ and Jie Zhang¹

¹School of Mathematics, Liaoning Normal University, Dalian 116029, China

²Institute of Information and Control, Hangzhou Dianzi University, Hangzhou 541004, China

Correspondence should be addressed to Wei Jiang; swxxjw@aliyun.com

Received 19 December 2017; Revised 20 March 2018; Accepted 24 April 2018; Published 1 August 2018

Academic Editor: Gao Cong

Copyright © 2018 Wei Jiang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Obtaining an optimum data representation is a challenging issue that arises in many intellectual data processing techniques such as data mining, pattern recognition, and gene clustering. Many existing methods formulate this problem as a nonnegative matrix factorization (NMF) approximation problem. The standard NMF uses the least square loss function, which is not robust to outlier points and noises and fails to utilize prior label information to enhance the discriminability of representations. In this study, we develop a novel matrix factorization method called robust semisupervised nonnegative local coordinate factorization by integrating robust NMF, a robust local coordinate constraint, and local spline regression into a unified framework. We use the $l_{2,1}$ norm for the loss function of the NMF and a local coordinate constraint term to make our method insensitive to outlier points and noises. In addition, we exploit the local and global consistencies of sample labels to guarantee that data representation is compact and discriminative. An efficient multiplicative updating algorithm is deduced to solve the novel loss function, followed by a strict proof of the convergence. Several experiments conducted in this study on face and gene datasets clearly indicate that the proposed method is more effective and robust compared to the state-of-the-art methods.

1. Introduction

Owing to the rapid development of data collection and storage techniques, there has been an increase in the demand for effective data representation approaches [1] to cope with image and gene information, particularly in the fields of pattern recognition, machine learning, and gene clustering. For large databases, an efficient representation of data [2–4] can improve the performance of numerous intelligent learning systems such as those used for classification and clustering analysis. In many application fields, the input samples are represented in high-dimensional form, which is infeasible for direct calculation. The efficiency and effectiveness of learning models exponentially decrease with each increase in the dimensionality of input samples, which is generally referred to as the “curse of dimensionality.” Accordingly, dimensionality reduction [5–7] is becoming increasingly important as it can overcome the curse of dimensionality, enhance the learning speed, and even offer critical insights

into the essence of the issue. In general, dimensionality reduction methods can be divided into two categories: feature extraction [5, 8, 9] and selection [10–14]. Feature selection involves selecting discriminative and highly related features from an input feature set, whereas feature extraction combines original features to form new features of data variables.

In recent years, there has been an increasing interest in feature extraction. Many feature extraction methods are designed to obtain a low-dimensional feature of high-dimensional data. These methods include singular value decomposition (SVD), principal component analysis (PCA) [5], nonnegative matrix factorization (NMF) [15, 16], and concept factorization (CF) [17]. Despite the different motivations of these models, they can all be interpreted as matrix decomposition, which often finds two or more low-dimensional matrices to approximate the original matrix. Factorization leads to a reduced representation of high-dimensional data and belongs to the category of methods employed for dimension reduction.

Unlike PCA [5] and SVD, NMF [15, 16] factorizes a sample matrix as a product of two matrices constrained by nonnegative elements. One matrix comprises new basis vectors that reveal the semantic structure, and the other matrix can be regarded as the set of coefficients composed of linear combinations of all sample points based on the new bases. Owing to their ability to extract the most discriminative features and their feasibility in computation, many extension versions [4, 18, 19] of NMF have been developed from various perspectives to enhance the original NMF. Sparseness-constrained NMF [20] has been introduced by adding l_1 norm minimization on the learned factor matrices to enhance sparsity for data representation. Fisher's criterion [21] has been incorporated into NMF formulation and is used to achieve discriminant representation. The semi- and convex-NMF formulations [22] relax the nonnegativity constraint of NMF by allowing the basis and coefficient matrices to have mixed signs, thereby extending the applicability of the method. Liu et al. [23] proposed a constrained NMF in which the label information is incorporated into the standard NMF for data representation. Cai et al. [24] extended NMF and proposed a graph-regularized NMF (GNMF) scheme, which imposes intrinsic geometry latent in a high-dimensional dataset onto the traditional NMF using an affinity graph. Chen et al. [9] presented a nonnegative local coordinate factorization (NLCF) method that imposes locality constraint onto the original NMF to explore faithful intrinsic geometry.

Traditional NMF and its variants usually adopt the square Euclidean distance to measure the approximation error. Although it has a solid theoretical foundation in mathematics and has shown encouraging performance in most cases, the square Euclidean distance is not always optimal for decomposition of a data matrix. The squared error has proved to be the best for both Gaussian and Poisson noise [25]. However, in real-world applications, data that violate the assumptions are usually involved. The squared loss is sensitive to outlier points and noises when the reconstruction error is measured. Even a single outlier point may sometimes easily dominate the objective function. In recent years, some variants have been presented to enhance the robustness of the classical NMF. A robust type of NMF that factorizes the sample matrix as the summation of two nonnegative matrices and one sparse error matrix was presented by Zhang et al. [26]. Zhang et al. [27] presented a robust NMF (RNMF) using the $l_{2,1}$ norm objective function, which can deal with outlier points and noises. Zhang et al. [28] presented a robust nonnegative graph-embedding framework (RNGE) that can simultaneously cope with noisy labels, noisy data, and uneven distribution.

Supervised learning algorithms [29–32] generally can achieve better performance than unsupervised learning techniques when label information is available in many applications. The motivation of semisupervised learning methods [33–38] is to employ numerous unlabeled samples as well as relatively few labeled samples to construct a better high-dimensional data analysis model. A surge of research interest in graph-based semisupervised learning techniques [37–39] [40] has recently occurred. Gaussian fields and harmonic

functions (GFHF) [33] is an efficient and effective semisupervised learning methods in which the predicted label matrix is reckoned on the graph with respect to manifold smoothness and label fitness. Xiang et al. [37] presented a method called local spline regression (LSR) in which an iterative algorithm is built on local neighborhoods through spline regression. Han et al. [38] presented a model of video semantic recognition using semisupervised feature selection via spline regression (S2FS2R). These methods not only consider label information but also employ the local and global structure consistency assumption.

Despite NMF's appealing advantages, it suffers from the following problems in real-world applications: (1) data may often be contaminated by noise and outliers due to illumination (e.g., specular reflections), image noises (e.g., scanned image data), occlusion (e.g., sunglasses and scarf in front of a face), among others. Although NMF can deal with noise in the test data to some extent, it will suffer from severe performance degradation when the training samples have noise. (2) In an NMF method, a data point may be represented by the base vectors, which are far from the data point, resulting in poor clustering performance. The standard NMF does not preserve the locality during its decomposition process, whereas local line coding can preserve such properties. (3) One of the challenges for classification tasks in the real world is the lack of labeled training data. Therefore, data labeled by an expert is often used as an alternative. Unfortunately, designating labels requires considerable human effort and is thus time-consuming and difficult to manage. In addition, an accurate label may require expert knowledge. However, unlabeled samples are relatively easy to obtain.

To address all the aforementioned issues, we present an efficient and effective matrix factorization framework called robust semisupervised nonnegative local coordinate factorization (RSNLCF) in which both data reconstruction functions and a local coordinate constraint regularization term are formulated in a $l_{2,1}$ norm manner to make our model robust to outlier points and noises. By integrating Green's functions and a set of primitive polynomials into the local spline, the local and global label consistency of data can be characterized based on their distribution. The main work of our study and its contributions are summarized as follows:

- (i) The proposed RSNLCF model is robust to outlier points and noises as a result of employing the $l_{2,1}$ norm formulations of NMF and a local coordinate constraint regularization term. In addition, to guarantee that the data representation is discriminative, local spline regression over labels is exploited.
- (ii) Unlike traditional dimension reduction approaches that treat feature extraction and selection separately, the proposed RSNLCF algorithm integrates the two aspects into a single optimization framework.
- (iii) We present an efficient algorithm to solve the presented RSNLCF model and provide the proof of rigorous convergence and correctness analysis of our model.

The remainder of this paper is organized as follows. Related studies are introduced in Section 2. We introduce our RSNLCF method and the optimization scheme in Section 3 and offer a convergence proof in Section 4. We describe and analyze the results of our experiments in Section 5. We conclude and discuss future work in Section 6.

2. Related Work

In this section, we summarize the notations and definitions of norm used in this study and briefly review NMF.

2.1. Notations and Definitions. Matrices and vectors are denoted by boldface capital and lowercase letters, respectively. $\|\mathbf{x}\|_p = (\sum_{i=1}^n |x_i|^p)^{1/p}$ denotes the l_p norm of the vector $\mathbf{x} \in \mathbb{R}^n$. \mathbf{x}^i and \mathbf{x}_j denote the i th row and the j th column of matrix $\mathbf{X} = (x_{ij})$, respectively. x_{ij} is the element in the i th row and j th column of \mathbf{X} , $\text{Tr}[\mathbf{X}]$ denotes the trace of \mathbf{X} if \mathbf{X} is a square matrix, and \mathbf{X}^T denotes the transposed matrix of \mathbf{X} . The Frobenius norm of the matrix $\mathbf{X} \in \mathbb{R}^{M \times N}$ is defined as

$$\|\mathbf{X}\|_F = \sqrt{\sum_{i=1}^M \sum_{j=1}^N x_{ij}^2}. \quad (1)$$

The $l_{2,1}$ norm of a matrix is defined as

$$\|\mathbf{X}\|_{2,1} = \sum_{i=1}^M \|\mathbf{x}^i\|_2 = \sum_{i=1}^M \sqrt{\sum_{j=1}^N x_{ij}^2} = \text{Tr}[\mathbf{X}^T \mathbf{D} \mathbf{X}], \quad (2)$$

where \mathbf{D} is a diagonal matrix with $D_{ii} = 1/2 \|\mathbf{x}^i\|_2$. However, $\|\mathbf{x}^i\|_2$ could approach zero. For this case, we define $D_{ii} = 1/2 \|\mathbf{x}^i\|_2 + \varepsilon$, where ε is a very small constant.

Assume that the matrix samples are represented as $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^L, \{\mathbf{x}_j\}_{j=L+1}^N$, where $\mathbf{x}_i\}_{i=1}^L, \{\mathbf{x}_j\}_{j=L+1}^N$ denotes labeled and unlabeled data, respectively. The labels of $\mathbf{x}_i\}_{i=1}^L$ are denoted as $l_i \in \{1, 2, \dots, L_c\}$ with L_c being the total number of categories. Let $\mathbf{F} \in \mathbb{R}^{L \times L_c}$ be a label indicator binary matrix with the j th entry $f_{ij} = 1$ if and only if \mathbf{x}_i is labeled with the j th class; $f_{ij} = 0$ otherwise. We also introduce a predicted label matrix $\mathbf{Y} \in \mathbb{R}^{N \times L_c}$, where each row is the predicted label vector of the data \mathbf{x}_i .

2.2. NMF. Given a nonnegative matrix $\mathbf{X} \in \mathbb{R}_+^{M \times N}$, each column of \mathbf{X} is a sample point. The main idea of NMF is to find two nonnegative matrices $\mathbf{U} = [u_{ik}] \in \mathbb{R}_+^{M \times K}$ and $\mathbf{V} = [v_{jk}] \in \mathbb{R}_+^{K \times N}$ that minimize the Euclidean distance between \mathbf{X} and \mathbf{UV} . The corresponding optimization problem is as follows:

$$\begin{aligned} \min_{\mathbf{U}, \mathbf{V}} \quad & \|\mathbf{X} - \mathbf{UV}\|_F^2 \\ \text{s.t.} \quad & \mathbf{U} \geq 0, \mathbf{V} \geq 0, \end{aligned} \quad (3)$$

where $\|\cdot\|_F$ is the Frobenius norm. To solve the objective function, Lee and Seung [15] proposed an iterative multiplicative updating algorithm as follows:

$$\begin{aligned} u_{jk}^{(t+1)} & \leftarrow u_{jk}^{(t)} \frac{(\mathbf{XV}^T)_{jk}}{(\mathbf{UVV}^T)_{jk}}, \\ v_{ki}^{(t+1)} & \leftarrow v_{ki}^{(t)} \frac{(\mathbf{U}^T \mathbf{X})_{ki}}{(\mathbf{U}^T \mathbf{UV})_{ki}}. \end{aligned} \quad (4)$$

By NMF, each column of \mathbf{U} and \mathbf{u}_i can be viewed as the basis, while the matrix \mathbf{V} can be treated as the set of the coefficients. Each sample point \mathbf{x}_i is approximated by a linear combination of the K bases, weighted by components of \mathbf{V} .

3. The Proposed RSNLCF Framework

In this section, we introduce our novel learning method for image clustering (RSNLCF), which is used to find an effective and robust representation of data.

3.1. Robust Sparse NMF. The square loss function based on the Frobenius norm is used to learn the data representations in NMF. However, it is very sensitive to outlier points and noises. Therefore, our robust representation model is represented as

$$\min_{\mathbf{U}, \mathbf{V}} \quad \|\mathbf{X} - \mathbf{UV}\|_{2,1} + \lambda \|\mathbf{V}\|_{2,1}, \quad (5)$$

where $\lambda > 0$ is the regularization parameter. Because the $l_{2,1}$ norm reduces the components occupied by the large magnitude of error in the loss function, the corrupted samples never dominate the objective function. In this sense, the loss function $\|\mathbf{X} - \mathbf{UV}\|_{2,1}$ is insensitive to outlier points and noises. Meanwhile, the regularization term $\|\mathbf{V}\|_{2,1}$ ensures that \mathbf{V} is sparse in rows. This means that some of \mathbf{V} 's rows approximate zero. Consequently, \mathbf{V} can be considered the combination coefficient for the most discriminative features. Feature selection is then achieved by \mathbf{V} , where only the features related to the nonzero rows in \mathbf{V} are chosen.

3.2. Robust Local Coordinate Constraint. Motivated by the concept of local coordinate coding [41], we present a robust local coordinate constraint as a regularization term for image clustering. First, we define coordinate coding.

Definition 1. Coordinate coding [41] can be written as concept pair (γ, C) , where C is defined as a set of anchor points with d dimensions and γ is a map of $\mathbf{x} \in \mathbb{R}^d$ to $[\gamma_v(\mathbf{x})]_{v \in C} \gamma_v(\mathbf{x})v$. It induces the following physical approximation of \mathbf{x} in \mathbb{R}^d : $\gamma(\mathbf{x}) = \sum_{v \in C} \gamma_v(\mathbf{x})v$.

For the local coordinate coding system, NMF can be considered as coordinate coding in which the columns of the matrix \mathbf{U} can be viewed as a set of anchor points, and each column of the coefficient matrix \mathbf{V} represents the corresponding coordinate coding for each data point. We might further hope that each sample point is represented as a linear combination of only a few proximate anchor points. A natural assumption here would be that if \mathbf{x}_i is far away from the anchor points \mathbf{u}_k , then its coordinate coding v_{ki} with respect to \mathbf{u}_k will tend to be zero and thus achieve sparsity and

locality simultaneously. The local coordinate constraint [41] can be defined as follows:

$$\min_{\mathbf{U}, \mathbf{V}} \sum_{i=1}^N \sum_{k=1}^K |v_{ki}| \|\mathbf{u}_k - \mathbf{x}_i\|_2^2 = \min_{\mathbf{U}, \mathbf{V}} \sum_{i=1}^N \left\| (\mathbf{x}_i \mathbf{1}^T - \mathbf{U}) \Lambda_i^{1/2} \right\|_F^2, \quad (6)$$

where \mathbf{x}_i denotes the i th column of \mathbf{X} , \mathbf{u}_k is the k th column of \mathbf{U} , v_{ki} is the coordinate of \mathbf{x}_i with respect to \mathbf{u}_k , and $\Lambda_i = \text{diag}(\mathbf{v}_i) \in \mathbb{R}^{K \times K}$, $\text{diag}(\mathbf{v}_i)$ indicates a conversion of the vector \mathbf{v}_i into a diagonal matrix in which the k th diagonal element is v_{ki} .

The local coordinate constraint employs a square loss. When the dataset is corrupted by outlier points and noises, the local coordinate constraint may fail to achieve sparsity and locality simultaneously. In order to alleviate the side effect of noisy data, our robust local coordinate constraint can be formulated as

$$\min_{\mathbf{U}, \mathbf{V}} \sum_{i=1}^N \left\| (\mathbf{x}_i \mathbf{1}^T - \mathbf{U}) \Lambda_i^{1/2} \right\|_{2,1}, \quad (7)$$

where the Frobenius norm-based square loss function has been substituted by the $l_{2,1}$ norm.

3.3. Local Spline Regression. In this subsection, we briefly introduce local spline regression [42].

Given N data points $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$ sampled from the underlying submanifold M , we use set $\mathcal{N}(\mathbf{x}_i) = \{x_{i_j}\}_{j=1}^k$ to denote \mathbf{x}_i and its $k-1$ nearest neighbor points, where $i_j \in \{1, 2, \dots, N\}$, and $\mathbf{Y}_i = [y_{i_1}, y_{i_2}, \dots, y_{i_k}]^T$ is the local predicted label matrix for the i th region. The task of local spline regression is to seek the predicting function $g_i : \mathbb{R}^M \rightarrow \mathbb{R}$ in order to map each data point $\mathbf{x}_{i_j} \in \mathbb{R}^M$ to the local predicted class label $y_{i_j} = g_i(x_{i_j})$. The model of local spline regression can be expressed as

$$\min_{g_i} \sum_{j=1}^k \left(y_{i_j} - g_i(x_{i_j}) \right)^2 + \gamma \mathcal{S}(g_i), \quad (8)$$

where $\mathcal{S}(g_i)$ is a regularization term and $\gamma > 0$ is a small positive regularization parameter to control the smoothness of the spline [42]. If $\mathcal{S}(g_i)$ is defined as a seminorm of a Sobolev space, g_i can be solved by the following objective function [43]:

$$g_i(\mathbf{x}) = \sum_{j=1}^d \beta_{i,j} p_j(\mathbf{x}) + \sum_{j=1}^k \alpha_{i,j} G_{i,j}(\mathbf{x}), \quad (9)$$

where $d = C_{M+s-1}^s$, in which s is the order of the partial derivatives [43]. $\{p_j(\mathbf{x})\}_{j=1}^d$ and $G_{i,j}$ are a set of primitive polynomials and a Green's function, respectively. The coefficients α_i and β_i can be achieved by solving the following problem:

$$\begin{pmatrix} \mathbf{K}_i & \mathbf{P}_i^T \\ \mathbf{P}_i & \mathbf{0} \end{pmatrix} \begin{pmatrix} \alpha_i \\ \beta_i \end{pmatrix} = \begin{pmatrix} \mathbf{Y}_i \\ \mathbf{0} \end{pmatrix}, \quad (10)$$

where \mathbf{K}_i is a symmetrical matrix with elements $K_{r,c} = G_{r,c}(x_{i_r} - x_{i_c})$, and \mathbf{P}_i is a matrix with its elements $P_{i,j} = p_i(x_{i_j})$. The local spline regression model can then be expressed as [42]

$$\min_{\mathbf{Y}_i} \mathbf{Y}_i^T \mathbf{M}_i \mathbf{Y}_i, \quad (11)$$

where \mathbf{M}_i is the upper left $k \times k$ submatrix of the inverse matrix of the coefficient matrix in (10). Because the local predicted label matrix \mathbf{Y}_i is a part of the global predicted label matrix \mathbf{Y} , we can construct a selection matrix $\mathbf{S}_i \in \mathbb{R}^{k \times N}$ for each \mathbf{Y}_i such that

$$\mathbf{Y}_i = \mathbf{S}_i \mathbf{Y}, \quad (12)$$

where the selection matrix \mathbf{S}_i is defined as follows:

$$S_i(r, c) = \begin{cases} 1, & \text{if } r = i_c, \\ 0, & \text{otherwise.} \end{cases} \quad (13)$$

After the local predicted label matrices are established, we combine them by minimizing the following loss function:

$$\min_{\mathbf{Y}_i} \sum_{i=1}^N \mathbf{Y}_i^T \mathbf{M}_i \mathbf{Y}_i = \sum_{i=1}^N \mathbf{Y}^T \mathbf{S}_i^T \mathbf{M}_i \mathbf{S}_i \mathbf{Y} = \mathbf{Y}^T \mathbf{M} \mathbf{Y}, \quad (14)$$

where

$$\mathbf{M} = \sum_{i=1}^N \mathbf{S}_i^T \mathbf{M}_i \mathbf{S}_i. \quad (15)$$

Based on the studies of [33, 34], the predicted label matrix \mathbf{Y} of the labeled data points should be consistent with the ground truth labels matrix \mathbf{F} . With the consistency constraints, the objective function (14) can be written as follows:

$$\min_{\mathbf{Y}} \text{Tr}(\mathbf{Y}^T \mathbf{M} \mathbf{Y}) + \eta \text{Tr}((\mathbf{Y} - \mathbf{F})^T \mathbf{E} (\mathbf{Y} - \mathbf{F})), \quad (16)$$

where \mathbf{E} is a diagonal matrix whose diagonal elements are 1 for labeled data and 0 for unlabeled data, and the elements of \mathbf{F} are defined as follows:

$$f_{ij} = \begin{cases} 1, & \text{if } x_i \text{ is labeled as class } j, \\ 0, & \text{otherwise.} \end{cases} \quad (17)$$

When η is sufficiently large, the optimal solution \mathbf{Y} to the problem (16) makes the second term approximately equal to zero. Thus, the objective function (16) guarantees local and global structural consistency over labels. All the elements of \mathbf{Y} are restricted to be nonnegative.

3.4. Objective Function of RSNLCF. By combining the RNMF (5), robust local coordinate constraint (7), and semisupervised local spline regression (16) into a unified

framework, we can formulate the objective function as follows:

$$\begin{aligned} \min_{\mathbf{U}, \mathbf{V}} \quad & \|\mathbf{X} - \mathbf{UV}\|_{2,1} + \mu \sum_{i=1}^N \|(\mathbf{x}_i \mathbf{1}^T - \mathbf{U}) \Lambda_i^{1/2}\|_{2,1} + \lambda \|\mathbf{V}\|_{2,1} \\ & + \tau \text{Tr}(\mathbf{Y}^T \mathbf{M} \mathbf{Y} + \eta(\mathbf{Y} - \mathbf{F})^T \mathbf{E}(\mathbf{Y} - \mathbf{F})), \end{aligned} \quad (18)$$

where τ and μ are two trade-off parameters. We call (18) our proposed RSNLCF.

4. Optimization

The objective function (18) involves the $l_{2,1}$ norm, which is nonsmooth and cannot have a closed form solution. Consequently, we propose to solve it as follows.

Denote $\mathbf{X} - \mathbf{UV} = [\mathbf{a}^1, \dots, \mathbf{a}^M]^T$, $(\mathbf{x}_i \mathbf{1}^T - \mathbf{U}) \Lambda_i^{1/2} = [\mathbf{b}^1, \dots, \mathbf{b}^K]^T$ and $\mathbf{V} = [\mathbf{v}^1, \dots, \mathbf{v}^M]^T$. When considering the nonnegative constraint on \mathbf{U} , \mathbf{V} , and \mathbf{Y} , the objective function (18) could be reformulated as

$$\begin{aligned} \mathcal{O} = & \text{Tr}((\mathbf{X} - \mathbf{UV})^T \mathbf{A}(\mathbf{X} - \mathbf{UV})) \\ & + \mu \text{Tr} \left(\sum_{i=1}^N ((\mathbf{x}_i \mathbf{1}^T - \mathbf{U}) \Lambda_i^{1/2}) \mathbf{B} ((\mathbf{x}_i \mathbf{1}^T - \mathbf{U}) \Lambda_i^{1/2}) \right)^T \\ & + \lambda \text{Tr}(\mathbf{V}^T \mathbf{C} \mathbf{V}) + \tau \text{Tr}(\mathbf{Y}^T \mathbf{M} \mathbf{Y} + \eta(\mathbf{Y} - \mathbf{F})^T \mathbf{E}(\mathbf{Y} - \mathbf{F})), \end{aligned}$$

s.t. $U \in \mathbb{R}^{M \times K} > 0, V \in \mathbb{R}^{K \times N} > 0, F \in \mathbb{R}^{N \times L_c} > 0,$ (19)

where \mathbf{A} , \mathbf{B} , and \mathbf{C} are three diagonal matrices with their diagonal elements given as $\mathbf{A}_{ii} = 1/2 \|\mathbf{a}^i\|_2$, $\mathbf{B}_{ii} = 1/2 \|\mathbf{b}^i\|_2$, and $\mathbf{C}_{ii} = 1/2 \|\mathbf{v}^i\|_2$, respectively.

4.1. Update Rules. The objective function \mathcal{O} of RSNLCF in (19) is not convex in \mathbf{U} , \mathbf{V} , and \mathbf{Y} together. Therefore, it is unrealistic to expect an algorithm to find the global minima. In this subsection, we describe our development of an iterative algorithm based on the Lagrangian multiplier method, which can achieve local minima. Following some algebraic steps, the objective function can be written as follows:

$$\begin{aligned} \mathcal{O} = & \text{Tr} \left(\mathbf{X} \mathbf{X}^T \mathbf{A} + \mathbf{UVV}^T \mathbf{U}^T \mathbf{A} - 2\mathbf{XV}^T \mathbf{U}^T \mathbf{A} \right. \\ & \left. + \mu \sum_{i=1}^N (\mathbf{x}_i \mathbf{1}^T \Lambda_i \mathbf{1} \mathbf{x}_i^T \mathbf{B} - 2\mathbf{x}_i \mathbf{1}^T \Lambda_i \mathbf{U}^T \mathbf{B} + \mathbf{U} \Lambda_i \mathbf{U}^T \mathbf{B}) \right) \\ & + \lambda \text{Tr}(\mathbf{V}^T \mathbf{C} \mathbf{V}) + \tau \text{Tr}(\mathbf{Y}^T \mathbf{M} \mathbf{Y} + \eta(\mathbf{Y} - \mathbf{F})^T \mathbf{E}(\mathbf{Y} - \mathbf{F})). \end{aligned} \quad (20)$$

To tackle the nonnegative constraint on \mathbf{U} , \mathbf{V} , and \mathbf{Y} , the objective (20) can be rewritten as the Lagrangian multiplier.

$$\begin{aligned} \mathcal{L} = & \text{Tr} \left(\mathbf{X} \mathbf{X}^T \mathbf{A} + \mathbf{UVV}^T \mathbf{U}^T \mathbf{A} - 2\mathbf{XV}^T \mathbf{U}^T \mathbf{A} \right. \\ & \left. + \mu \sum_{i=1}^N (\mathbf{x}_i \mathbf{1}^T \Lambda_i \mathbf{1} \mathbf{x}_i^T \mathbf{B} - 2\mathbf{x}_i \mathbf{1}^T \Lambda_i \mathbf{U}^T \mathbf{B} + \mathbf{U} \Lambda_i \mathbf{U}^T \mathbf{B}) \right) \\ & + \lambda \text{Tr}(\mathbf{V}^T \mathbf{C} \mathbf{V}) + \tau \text{Tr}(\mathbf{Y}^T \mathbf{M} \mathbf{Y} + \eta(\mathbf{Y} - \mathbf{F})^T \mathbf{E}(\mathbf{Y} - \mathbf{F})) \\ & - \text{Tr}(\mathbf{\Psi} \mathbf{U}^T) - \text{Tr}(\mathbf{\Phi} \mathbf{V}^T) - \text{Tr}(\mathbf{\Theta} \mathbf{Y}^T), \end{aligned} \quad (21)$$

where $\mathbf{\Psi} = [\psi_{jk}]$, $\mathbf{\Phi} = [\phi_{ki}]$, and $\mathbf{\Theta} = [\theta_{is}]$ are the Lagrangian multipliers. Let the partial derivatives of the objective function (21) with respect to \mathbf{U} , \mathbf{V} , and \mathbf{Y} be zero. Thus, we have

$$\begin{aligned} \mathbf{\Psi} &= 2\mathbf{AUVV}^T - 2\mathbf{AXV}^T - 2\mu\mathbf{BXV}^T + 2\mu\mathbf{BUH}, \\ \mathbf{\Phi} &= 2\mathbf{U}^T \mathbf{AUV} - 2\mathbf{U}^T \mathbf{AX} + \mu(\mathbf{G} - 2\mathbf{U}^T \mathbf{BX} + \mathbf{D}) + 2\lambda\mathbf{CV}, \\ \mathbf{\Theta} &= 2\tau\mathbf{MY} + 2\tau\eta\mathbf{E}(\mathbf{Y} - \mathbf{F}), \end{aligned} \quad (22)$$

where \mathbf{H} is a diagonal matrix whose entries are row sums of \mathbf{V} . $\mathbf{G} = (\mathbf{g}_1, \dots, \mathbf{g}_K)^T$ is a $K \times N$ matrix whose columns are $\mathbf{g} = \text{diag}(\mathbf{X}^T \mathbf{B} \mathbf{X}) \in \mathbb{R}^N$. $\mathbf{D} = (\mathbf{d}_1, \dots, \mathbf{d}_K)$ is a $K \times N$ matrix, and $\mathbf{d} = \text{diag}(\mathbf{U}^T \mathbf{B} \mathbf{U}) \in \mathbb{R}^K$.

Based on the Karush-Kuhn-Tucker conditions [44] $\psi_{jk} u_{jk} = 0$, $\phi_{ki} v_{ki} = 0$ and $\theta_{is} y_{is} = 0$, we obtain

$$\begin{aligned} & (\mathbf{AUVV}^T)_{jk} u_{jk} - (\mathbf{AXV}^T)_{jk} u_{jk} - \mu(\mathbf{BXV}^T)_{jk} u_{jk} \\ & + \mu(\mathbf{BUH})_{jk} u_{jk} = 0, \\ & 2(\mathbf{U}^T \mathbf{AUV})_{ki} v_{ki} - 2(\mathbf{U}^T \mathbf{AX})_{ki} v_{ki} + 2\lambda(\mathbf{CV})_{ki} v_{ki} \\ & + \mu(\mathbf{G} - 2\mathbf{U}^T \mathbf{BX} + \mathbf{D})_{ki} v_{ki} = 0, \\ & (\mathbf{MY})_{is} y_{is} + \eta(\mathbf{E}(\mathbf{Y} - \mathbf{F}))_{is} y_{is} = 0. \end{aligned} \quad (23)$$

The corresponding equivalent formulas are as follows:

$$\begin{aligned} & (\mathbf{AUVV}^T)_{jk} u_{jk}^2 - (\mathbf{AXV}^T)_{jk} u_{jk}^2 - \mu(\mathbf{BXV}^T)_{jk} u_{jk}^2 \\ & + \mu(\mathbf{BUH})_{jk} u_{jk}^2 = 0, \end{aligned} \quad (24)$$

$$\begin{aligned} & 2(\mathbf{U}^T \mathbf{AUV})_{ki} v_{ki}^2 - 2(\mathbf{U}^T \mathbf{AX})_{ki} v_{ki}^2 + 2\lambda(\mathbf{CV})_{ki} v_{ki}^2 \\ & + \mu(\mathbf{G} - 2\mathbf{U}^T \mathbf{BX} + \mathbf{D})_{ki} v_{ki}^2 = 0. \end{aligned} \quad (25)$$

$$(\mathbf{MY})_{is} y_{is}^2 + \eta(\mathbf{E}(\mathbf{Y} - \mathbf{F}))_{is} y_{is}^2 = 0. \quad (26)$$

Solving (24), (25), and (26), we obtain the following update rules, given by

$$u_{jk}^{(t+1)} \leftarrow u_{jk}^{(t)} \sqrt{\frac{(\mathbf{AXV}^T + \mu\mathbf{BXV}^T)_{jk}}{(\mathbf{AUVV}^T + \mu\mathbf{BUV})_{jk}}}, \quad (27)$$

$$v_{ki}^{(t+1)} \leftarrow v_{ki}^{(t)} \sqrt{\frac{2(\mathbf{U}^T \mathbf{AX} + \mu\mathbf{U}^T \mathbf{BX})_{ki}}{(2\mathbf{U}^T \mathbf{AUV} + \mu\mathbf{G} + \mu\mathbf{D} + 2\lambda\mathbf{CV})_{ki}}}, \quad (28)$$

$$y_{is}^{(t+1)} \leftarrow y_{is}^{(t)} \sqrt{\frac{\eta(\mathbf{EF})_{is}}{(\mathbf{MY} + \eta\mathbf{EY})_{is}}}. \quad (29)$$

In this manner, we obtain the solver for the objective function (19).

4.2. Convergence Analysis. In this subsection, we demonstrate that the objective function (20) converges to a local optimum by using the update rules (27), (28), and (29) after finite iterations. We adopt the auxiliary function approach [16] to prove the convergence. Here, we first introduce the definition of an auxiliary function.

Definition 1. $Z(q, q')$ is an auxiliary function for $F(q)$ if the following properties are satisfied:

$$Z(q, q') \geq F(q), Z(q, q) = F(q). \quad (30)$$

Lemma 1. If Z is an auxiliary function for F , then F is nonincreasing under the update:

$$h^{(t+1)} = \operatorname{argmin}_q Z(q, q^{(t)}). \quad (31)$$

Proof 1.

$$F(q^{(t+1)}) \leq Z(q^{(t+1)}, q^{(t)}) \leq Z(q^{(t)}, q^{(t)}) = F(q^{(t)}). \quad (32)$$

Lemma 2. For any nonnegative matrices $\mathbf{A} \in \mathbb{R}^{n \times n}$, $\mathbf{B} \in \mathbb{R}^{k \times k}$, $\mathbf{S} \in \mathbb{R}^{n \times k}$, $\mathbf{S}' \in \mathbb{R}^{n \times k}$, and \mathbf{A}, \mathbf{B} are symmetric, and then the following inequality holds

$$\sum_{i=1}^n \sum_{p=1}^k \frac{(\mathbf{A}\mathbf{S}'\mathbf{B})_{ip} \mathbf{S}_{ip}^2}{\mathbf{S}'_{ip}} \geq \operatorname{Tr}(\mathbf{S}'\mathbf{A}\mathbf{S}\mathbf{B}). \quad (33)$$

The convergence of the algorithms is demonstrated in the following:

For given \mathbf{X} , the optimizing objective function (20) w.r.t. \mathbf{V} is equivalent to minimizing

$$\begin{aligned} \mathcal{O}(\mathbf{V}) = & \operatorname{Tr}(\mathbf{V}^T \mathbf{U}^T \mathbf{A} \mathbf{U} \mathbf{V}) - 2 \operatorname{Tr}(\mathbf{U}^T \mathbf{A} \mathbf{X} \mathbf{V}^T) + \mu \operatorname{Tr}(\mathbf{V}^T \mathbf{G}) \\ & - 2\mu \operatorname{Tr}(\mathbf{U}^T \mathbf{B} \mathbf{X} \mathbf{V}^T) + \mu \operatorname{Tr}(\mathbf{V}^T \mathbf{D}) + \lambda \operatorname{Tr}(\mathbf{V}^T \mathbf{C} \mathbf{V}). \end{aligned} \quad (34)$$

Theorem 1. The following function

$$\begin{aligned} Z(\mathbf{V}, \mathbf{V}') = & \sum_{ki} \frac{(\mathbf{U}^T \mathbf{A} \mathbf{U} \mathbf{V}')_{ki} \mathbf{V}_{ki}^2}{\mathbf{V}'_{ki}} - 2 \sum_{ki} (\mathbf{U}^T \mathbf{A} \mathbf{X})_{ki} \mathbf{V}_{ki}' \\ & \cdot \left(1 + \log \frac{\mathbf{V}_{ki}}{\mathbf{V}'_{ki}} \right) + \mu \sum_{ki} \mathbf{G}_{ki} \frac{\mathbf{V}_{ki}^2 + (\mathbf{V}')_{ki}^2}{2\mathbf{V}'_{ki}} \\ & - 2\mu \sum_{ki} (\mathbf{U}^T \mathbf{B} \mathbf{X})_{ki} \mathbf{V}_{ki}' \left(1 + \log \frac{\mathbf{V}_{ki}}{\mathbf{V}'_{ki}} \right) \\ & + \mu \sum_{ki} \mathbf{D}_{ki} \frac{\mathbf{V}_{ki}^2 + (\mathbf{V}')_{ki}^2}{2\mathbf{V}'_{ki}} + \lambda \sum_{ki} \frac{(\mathbf{C} \mathbf{V}')_{ki} \mathbf{V}_{ki}^2}{\mathbf{V}'_{ki}} \end{aligned} \quad (35)$$

is an auxiliary function for $\mathcal{O}(\mathbf{V})$.

Proof 1. In one sense, $Z(\mathbf{V}, \mathbf{V}) = \mathcal{O}(\mathbf{V})$ is obvious. However, we need to prove that $Z(\mathbf{V}, \mathbf{V}') \geq \mathcal{O}(\mathbf{V})$. To accomplish this, we compare (34) and (35) to find out that $Z(\mathbf{V}, \mathbf{V}') \geq \mathcal{O}(\mathbf{V})$.

By applying Lemma 2, we obtain

$$\begin{aligned} \operatorname{Tr}(\mathbf{U}^T \mathbf{A} \mathbf{U} \mathbf{V} \mathbf{V}^T) & \leq \sum_{ki} \frac{(\mathbf{U}^T \mathbf{A} \mathbf{U} \mathbf{V}')_{ki} \mathbf{V}_{ki}^2}{\mathbf{V}'_{ki}}, \\ \lambda \operatorname{Tr}(\mathbf{V}^T \mathbf{C} \mathbf{V}) & \leq \lambda \sum_{ki} \frac{(\mathbf{C} \mathbf{V}')_{ki} \mathbf{V}_{ki}^2}{\mathbf{V}'_{ki}}. \end{aligned} \quad (36)$$

To obtain the upper bound for the third and fifth terms, we use the inequality $a^2 + b^2 \geq 2ab$, which holds for any $a, b \geq 0$, and these third and fifth terms in $\mathcal{O}(\mathbf{V})$ are bounded by

$$\begin{aligned} \mu \operatorname{Tr}(\mathbf{V}^T \mathbf{G}) & \leq \mu \sum_{ki} \mathbf{G}_{ki} \frac{\mathbf{V}_{ki}^2 + (\mathbf{V}')_{ki}^2}{2\mathbf{V}'_{ki}}, \\ \mu \operatorname{Tr}(\mathbf{V}^T \mathbf{D}) & \leq \mu \sum_{ki} \mathbf{D}_{ki} \frac{\mathbf{V}_{ki}^2 + (\mathbf{V}')_{ki}^2}{2\mathbf{V}'_{ki}}. \end{aligned} \quad (37)$$

To obtain lower bounds for the remaining terms, we adopt the inequality $z \geq 1 + \log z$, $\forall z$, and then

$$\begin{aligned} 2 \operatorname{Tr}(\mathbf{U}^T \mathbf{A} \mathbf{X} \mathbf{V}^T) & \geq 2 \sum_{ki} (\mathbf{U}^T \mathbf{A} \mathbf{X})_{ki} \mathbf{V}_{ki}' \left(1 + \log \frac{\mathbf{V}_{ki}}{\mathbf{V}'_{ki}} \right), \\ 2\mu \operatorname{Tr}(\mathbf{U}^T \mathbf{B} \mathbf{X} \mathbf{V}^T) & \geq 2\mu \sum_{ki} (\mathbf{U}^T \mathbf{B} \mathbf{X})_{ki} \mathbf{V}_{ki}' \left(1 + \log \frac{\mathbf{V}_{ki}}{\mathbf{V}'_{ki}} \right). \end{aligned} \quad (38)$$

Summing all inequalities, we can obtain $Z(\mathbf{V}, \mathbf{V}') \geq \mathcal{O}(\mathbf{V})$ which obviously satisfies $Z(\mathbf{V}, \mathbf{V}') \geq \mathcal{O}(\mathbf{V})$. Therefore, $Z(\mathbf{V}, \mathbf{V}')$ is an auxiliary function of $\mathcal{O}(\mathbf{V})$.

Theorem 2. The updating rule (28) can be obtained by minimizing the auxiliary function $Z(\mathbf{V}, \mathbf{V}')$.

Proof 1. To find the minimum of $Z(\mathbf{V}, \mathbf{V}')$, we set the derivative $\partial Z(\mathbf{V}, \mathbf{V}') / \partial \mathbf{V}_{ki} = 0$ and obtain

$$\begin{aligned} \frac{\partial Z(\mathbf{V}, \mathbf{V}')}{\partial \mathbf{V}_{ki}} = & \frac{2(\mathbf{U}^T \mathbf{A} \mathbf{U} \mathbf{V}')_{ki} \mathbf{V}_{ki}}{\mathbf{V}'_{ki}} - \frac{2(\mathbf{U}^T \mathbf{A} \mathbf{X})_{ki} \mathbf{V}_{ki}'}{\mathbf{V}_{ki}} \\ & + \mu \frac{\mathbf{G}_{ki} \mathbf{V}_{ki}}{\mathbf{V}'_{ki}} - \mu \frac{2(\mathbf{U}^T \mathbf{B} \mathbf{X})_{ki} \mathbf{V}_{ki}'}{\mathbf{V}_{ki}} \\ & + \mu \frac{\mathbf{D}_{ki} \mathbf{V}_{ki}}{\mathbf{V}'_{ki}} + \lambda \frac{2(\mathbf{C} \mathbf{V}')_{ki} \mathbf{V}_{ki}}{\mathbf{V}'_{ki}}. \end{aligned} \quad (39)$$

Thus, by simple algebraic formulation, we can obtain the iterative updating rule for \mathbf{V} as (28).

Based on the properties of the auxiliary, we prove that the objective function (20) monotonically decreases under the updating v_{ki} .

The converge proofs showing that updating u_{jk} and y_{is} can be accomplished using (27) and (29) are similar to the aforementioned.

5. Experiments and Discussion

We systematically evaluated the performance of our presented RSNLCF method and compared it to the popular clustering methods.

5.1. Datasets. Three standard face datasets and the gene dataset were selected to evaluate different methods. The four datasets are described as follows:

- (i) *Extended YaleB dataset:* the extended YaleB dataset contains 2414 frontal face images of 38 individuals. In this dataset, the size of each face image is 192×168 and each image was acquired from 64 illuminate conditions and nine individual poses. Each image was resized to 32×32 in our experiments.
- (ii) *ORL face dataset:* the OR dataset contains 400 images of 40 individuals. All images were captured at different times and with different variations including lighting, face expressions (open and closed eyes, smiling, and not smiling), and specific facial details (glasses and no glasses). The original images had a size of 92×112 . Each image was rescaled to 32×32 .
- (iii) *AR dataset:* the AR dataset contains over 4000 frontal face images of 126 individuals (70 men and 56 women) with different facial expressions, illumination conditions, and occlusions (sunglasses and scarf). All individuals participated in two photo sessions, and 26 images of each individual were captured. Each image was scaled to 32×32 .
- (iv) *Leukemia dataset:* the leukemia dataset contains data related to and samples of acute myelogenous leukemia (AML) and acute lymphoblastic leukemia (ALL). ALL can be further classified as T and B subtypes. This dataset consists of 5000 genes in 38 set of tumor data and contains 19 samples of B cell ALL B, eight samples of T cell ALL T, and 11 samples of AML.

5.2. Experimental Design. In this section, we describe our evaluation metrics, the compared methods, and our parameter selection.

5.2.1. Evaluation Metrics. In our experiments, two widely used metrics (i.e., accuracy (Acc) and normalized mutual information (NMI)) were adopted to evaluate the clustering results [45]. We evaluated the algorithms by comparing the cluster labels of each data point with its label provided by the dataset. The Acc metric is defined as follows:

$$\text{Acc} = \frac{\sum_{i=1}^n \delta(\text{map}(r_i), l_i)}{n}, \quad (40)$$

where n refers to the total number of samples, r_i denotes the cluster label of x_i , and l_i is the true class label. In addition, $\delta(x, y)$ is the delta function that is equal to 1 if $x = y$ and 0 otherwise, and $\text{map}(r_i)$ is the mapping function that maps the obtained label r_i to the equivalent label from the dataset. The best mapping function can be determined by using the Kuhn-Munkres algorithm [46]. The value of Acc is equal to 1 if and only if the clustering result and the true label are identical. The second measure is the NMI, which is adopted in order to evaluate the quality of clusters. Given a clustering result, the NMI is defined as follows:

$$\text{NMI} = \frac{\sum_{i=1}^c \sum_{j=1}^c n_{i,j} \log(n_{i,j}/n_i \hat{n}_j)}{\sqrt{(\sum_{i=1}^c n_i \log(n_i/n)) (\sum_{j=1}^c \hat{n}_j \log(\hat{n}_j/n))}}, \quad (41)$$

where n_i denotes the number of images contained in the i th cluster C_i based on clustering results, \hat{n}_j is the number of images belonging to the C'_j and $n_{i,j}$ is the number of images that are in the intersection of C_i and C'_j .

5.2.2. Compared Methods. To verify the clustering performance of our RSNLCF, several popular methods were compared using the same dataset. The methods are listed as follows:

- (i) RNMF using $l_{2,1}$ norm [27]
- (ii) Semisupervised graph-regularized NMF (semi-GNMF) [24]
- (iii) Constrained NMF (CNMF) [16]
- (iv) Local centroid-structured NMF (LCSNMF) [47]
- (v) Unsupervised robust seminonnegative graph embedding through the $l_{2,1}$ norm (URNAGE) [28]
- (vi) Nonnegative local coordinate factorization (NLCF) [9]
- (vii) Our proposed RSNLCF

Sample images are shown in Figure 1.

5.2.3. Parameter Selection. Some parameters had to be tuned in the evaluated algorithms. To compare different algorithms fairly, we ran them using different parameters and chose the best average performance obtained for comparison. We set the number of clusters to be the same as the true number of categories on three image datasets and the leukemia dataset. Note that there was no parameter selection for RNMF and CNMF when the number of clusters was given. The regularization parameters were searched over the grid $\{0.001, 0.01, 0.1, 1, 10, 100, 1000\}$ for semi-GNMF, URNAGE, NLCF, and RSNLCF. The neighborhood size k to build the graph was chosen from $\{1, 2, \dots, 10\}$, and the 0-1 weighting scheme was adopted for its simplicity in the graph-based methods of semi-GNMF and URNAGE. We applied the approach

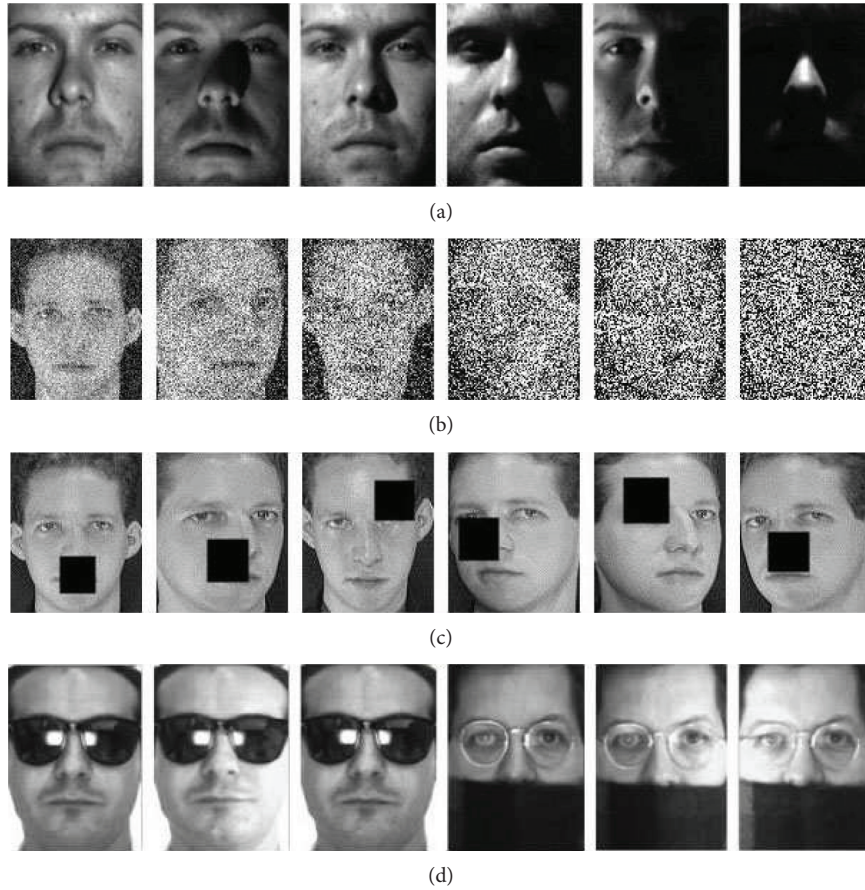


FIGURE 1: Sample images. (a) Extended YaleB dataset, (b) ORL dataset with random pixel corruption, (c) ORL dataset with random block occlusions, and (d) AR dataset with contiguous occlusions by sunglasses and scarves.

presented in literature [16] to adjust automatically the value of λ for LCSNMF.

5.3. Face Clustering under Illumination Variations. The robustness of the approaches to illumination changes was tested widely with the extended YaleB dataset. Figure 1(a) shows some samples from this dataset. We used only the frontal face images of the first 18 individuals. Our experiments were performed with various numbers of clusters. For the fixed cluster number k , the images of k categories from the extended YaleB dataset were randomly selected and mixed for evaluation. For semisupervised methods semi-GNMF, CNMF, and URNGE, eight face images per individual were randomly chosen as labeled samples; the rest of the dataset was used as unlabeled samples. On the clustering set, the compared methods were used to achieve new data representations. For a fair comparison, we used k -means to cluster samples based on the new data representations. The results of k -means are related to initialization. We repeated the experiments 20 times with different initialization parameters. The clustering results were measured by the commonly used evaluation metrics, Acc and NMI. Table 1 shows the detailed clustering results on different clustering numbers. The final row shows the average clustering accuracy (NMI) over k . Compared with the second best method, our method (RSNLCF) achieves an 11.41% improvement in clustering

accuracy. For mutual information, it achieved a 10.63% improvement over the second best algorithm.

5.4. Face Clustering under Pixel Corruptions. Two experiments were designed to test the robustness of RSNLCF against random pixel corruptions on the ORL face dataset. For the semisupervised algorithms of semi-GNMF, CNMF, URNGE, and RSNLCF, three images per individual were randomly chosen as labeled samples, and the remaining images were used as unlabeled samples. In the first experiment, each image was corrupted by replacing the pixel value with independent and identically distributed samples whose lower and upper bounds were the minimum and maximum pixel value of the image, respectively. The corrupted pixels of each image varied from 10 to 90% in increments of 10%. Figure 1(b) shows several examples. Because the corrupted pixels were randomly selected for each test sample, we repeated the experiments 20 times. Figure 2 displays the recognition accuracies over different levels of corruption. The recognition accuracies of the methods decreased rapidly as the level of corruption increased. From Figure 2, which depicts the recognition accuracies, we can observe that the proposed method consistently outperformed the others. When the samples had a high percentage of pixel corruption, the methods failed to obtain improved recognition performance because of inadequate discriminative information.

TABLE 1: Clustering performance on the extended YaleB dataset.

k	RNMF	Semi-GNMF	CNMF	LCSNMF	URNGE	NLCF	RSNLCF
Acc (%)							
2	78.43 ± 16.23	90.47 ± 15.25	76.61 ± 15.34	92.25 ± 17.34	91.02 ± 18.83	89.54 ± 13.26	96.25 ± 15.64
4	69.52 ± 15.01	84.85 ± 14.43	65.62 ± 12.36	88.75 ± 14.76	86.33 ± 16.76	83.03 ± 12.74	94.63 ± 14.53
6	53.45 ± 5.55	82.36 ± 9.63	63.74 ± 6.75	86.14 ± 4.94	84.23 ± 5.66	77.64 ± 13.24	92.37 ± 9.85
8	52.76 ± 3.46	83.71 ± 7.81	65.42 ± 6.41	85.79 ± 6.91	85.39 ± 5.78	75.83 ± 7.33	90.18 ± 6.63
10	53.24 ± 3.47	77.05 ± 4.17	63.68 ± 5.25	78.04 ± 5.79	74.58 ± 2.43	70.42 ± 4.87	88.31 ± 4.14
12	55.11 ± 4.53	72.84 ± 3.15	63.25 ± 4.34	75.65 ± 4.36	73.34 ± 2.76	71.72 ± 7.23	87.26 ± 3.56
14	52.05 ± 3.26	71.57 ± 2.24	61.58 ± 3.86	73.91 ± 3.43	72.16 ± 2.14	68.52 ± 2.13	87.11 ± 2.34
16	51.97 ± 3.17	67.85 ± 2.45	59.91 ± 3.42	69.35 ± 3.82	68.35 ± 1.82	65.46 ± 3.14	85.67 ± 3.16
18	51.53 ± 3.32	67.58 ± 3.01	57.33 ± 2.97	71.64 ± 3.58	69.09 ± 2.23	65.83 ± 3.71	85.32 ± 3.01
Avg.	57.56	77.59	64.13	80.17	78.28	74.22	89.69
NMI (%)							
2	82.91 ± 18.13	92.51 ± 18.52	78.81 ± 16.28	94.36 ± 19.84	93.35 ± 17.46	91.71 ± 17.15	98.46 ± 14.75
4	72.63 ± 17.92	87.24 ± 16.35	71.58 ± 13.41	92.71 ± 15.14	90.78 ± 15.46	85.53 ± 15.94	96.38 ± 13.52
6	63.42 ± 5.21	85.91 ± 6.47	68.37 ± 8.86	90.25 ± 9.48	86.13 ± 6.48	82.82 ± 12.03	94.35 ± 7.51
8	62.14 ± 2.96	86.86 ± 3.72	70.54 ± 7.97	89.37 ± 8.16	88.73 ± 6.52	80.34 ± 5.02	93.82 ± 5.73
10	64.06 ± 3.01	84.25 ± 3.26	69.72 ± 6.33	87.54 ± 5.42	86.82 ± 4.36	79.16 ± 2.92	92.73 ± 6.22
12	63.41 ± 3.05	80.82 ± 2.33	68.98 ± 4.64	85.28 ± 5.37	84.94 ± 4.53	80.47 ± 3.77	90.69 ± 4.91
14	58.99 ± 2.11	79.34 ± 3.44	67.67 ± 5.78	84.91 ± 4.98	83.12 ± 3.98	77.55 ± 1.76	90.43 ± 3.01
16	57.23 ± 2.17	77.85 ± 2.97	65.98 ± 4.58	82.11 ± 4.12	81.66 ± 3.54	76.46 ± 2.44	89.27 ± 4.62
18	55.59 ± 2.03	77.51 ± 2.25	64.67 ± 3.98	83.79 ± 4.43	82.36 ± 3.28	77.59 ± 1.99	89.21 ± 4.75
Avg.	64.49	86.02	69.59	87.81	86.43	81.29	97.06

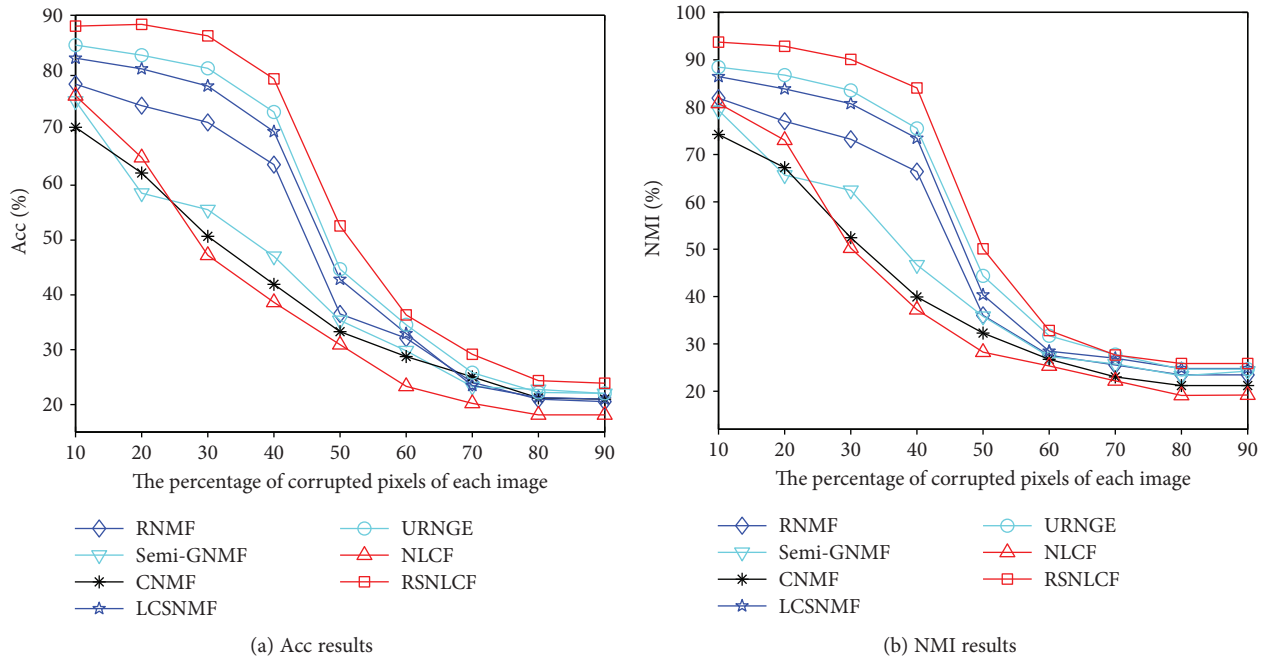


FIGURE 2: Clustering Acc and NMI curves across percentages of corrupted pixels of each image for the compared methods on the ORL dataset.

In the second experiment, 40% of the pixels randomly selected from each sample were replaced by setting the pixel value as 255. The number of corrupted samples of each individual is gradually increased from 10 to 90%. We conducted

the evaluations 20 times at different corruption percentages and computed the average recognition accuracies of Acc and NMI. Figure 3 illustrates clustering Acc and NMI curves of RSNLCF and the proposed method's six competitors

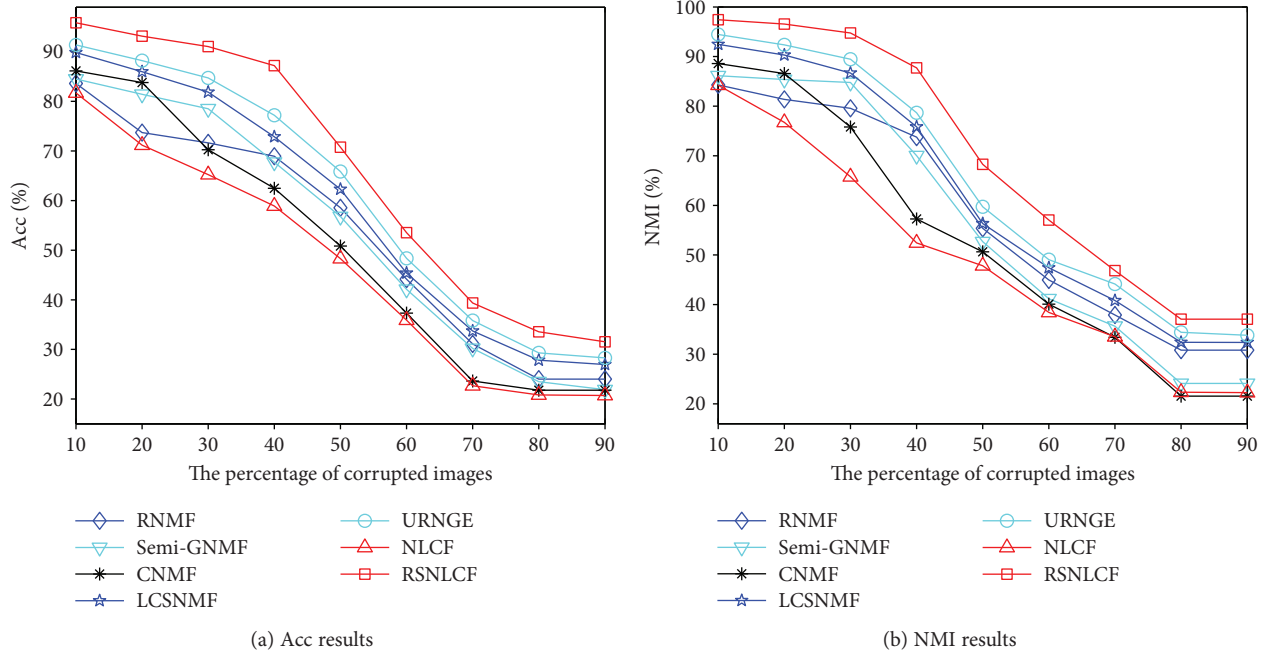


FIGURE 3: Clustering Acc and NMI curves across percentages of corrupted images for the compared methods on the ORL dataset.

versus the percentage of corrupted images. From Figure 3, which depicts the comparison results on the ORL dataset, we can clearly see that the RSNLCF obtained the best recognition accuracy in all situations.

5.5. Face Clustering under Contiguous Occlusions. We validated the robustness of RSNLCF against partial block occlusions (see Figure 1(c) for examples). Two experiments were conducted on the ORL face dataset. For the semisupervised algorithms of semi-GNMF, CNMF, URNGE, and RSNLCF, we randomly selected three samples from each category and used their category number as the label information. The first experiment was performed with a fixed contiguous block occlusion size of 40×40 pixels. We chose r of the face samples of each individual for occlusion, with r varying from 10 to 90%. The position of the block was randomly selected. The evaluations were performed 20 times for each r , and the means of Acc and NMI were recorded. Figure 4 shows the means of clustering Acc and NMI of the compared methods on different percentages of corrupted images. As shown in Figure 4, the performances of NMF, RNMF, semi-GNMF, CNMF, URNGE, and NLCF were lower than that of RSNLCF. With an increasing number of occluded samples, the clustering accuracy of RSNLCF dropped and thus matched expectations considerably.

In the second experiment, we simulated various levels of contiguous occlusions in each image by using an unrelated image of size $p \times p$ with $p \in \{5, 10, 20, \dots, 80\}$. The evaluations were conducted 20 times at each occlusion level, and the average Acc and NMI curves were recorded. Figure 5 plots clustering Acc and NMI results of the compared methods under different occlusion levels. Although the clustering accuracy of each method degraded with each increment

in occlusion level, RSNLCF consistently exceeded other methods. When the occlusion size increased to 50×50 , the occluding part dominated the image and caused the clustering performance to diminish rapidly.

5.6. Face Clustering under Real Occlusions. We evaluated the robustness of RSNLCF against real malicious occlusions. The AR dataset adopted in this experiment contains 2600 frontal face images from 100 individuals (50 males and 50 females from two photo sessions). Figure 1(d) shows some face samples with real occlusions by sunglasses and scarf. Note that because RNMF, LCSNMF, and NLCF are unsupervised algorithms, we did not compare them here. In this experiment, we randomly selected r face images per individual as labeled samples, in which r was varied from four to 18, respectively, in increments of two. The remaining images were unlabeled samples. For each configuration, we conducted 20 test runs with each method. The mean and the standard deviation of clustering accuracy were recorded. Table 2 tabulates the detailed clustering results by Acc and NMI on the AR dataset and shows our algorithm achieved 8.55, 12.82, and 14.53% Acc improvement over URNGE, CNMF, and semi-GNMF, respectively.

For NMI, the recognition rate of RSNLCF was 7.06, 9.66, and 10.87% higher than URNGE, CNMF, and semi-GNMF, respectively.

5.7. Gene Data Clustering on the Leukemia Dataset. Finally, we assessed clustering performance on the leukemia dataset. The gene expression dataset was rather challenging in terms of clustering issues, because it contains numerous features but only a few samples. We filtered out genes with $\max/\min < 15$ and $\max - \min < 500$, leaving a total of 1999 genes.

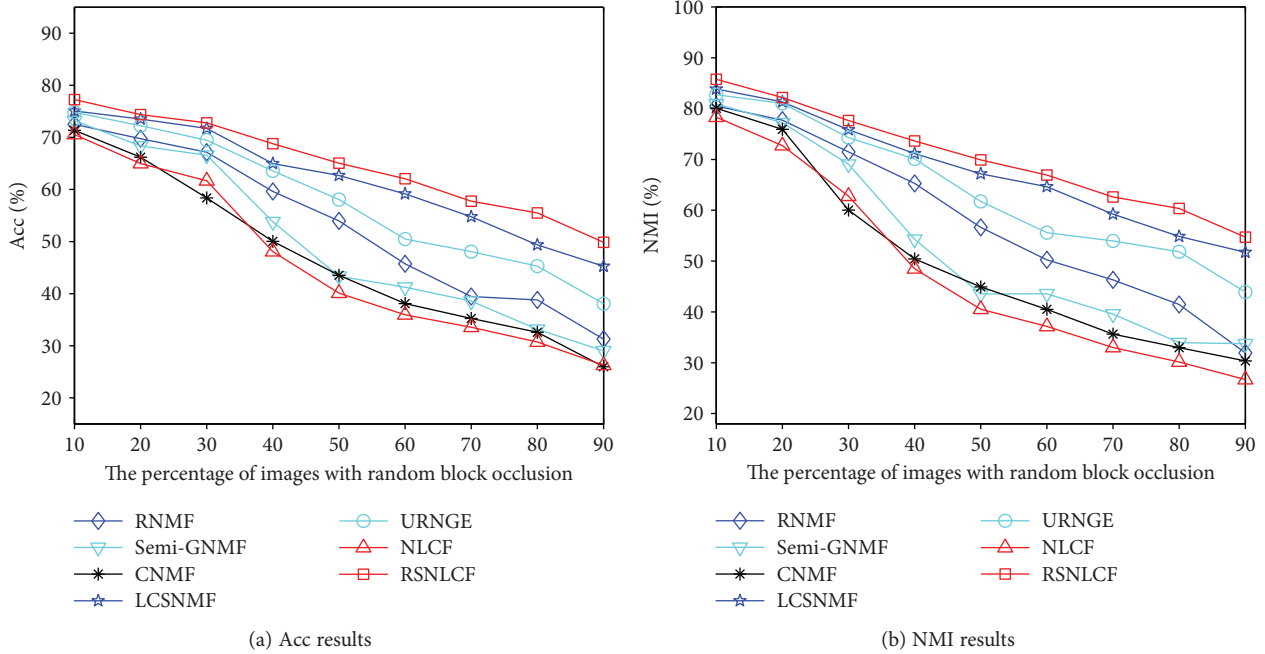


FIGURE 4: Clustering Acc and NMI curves of the compared methods on percentages of corrupted images with random block occlusions for the ORL dataset.

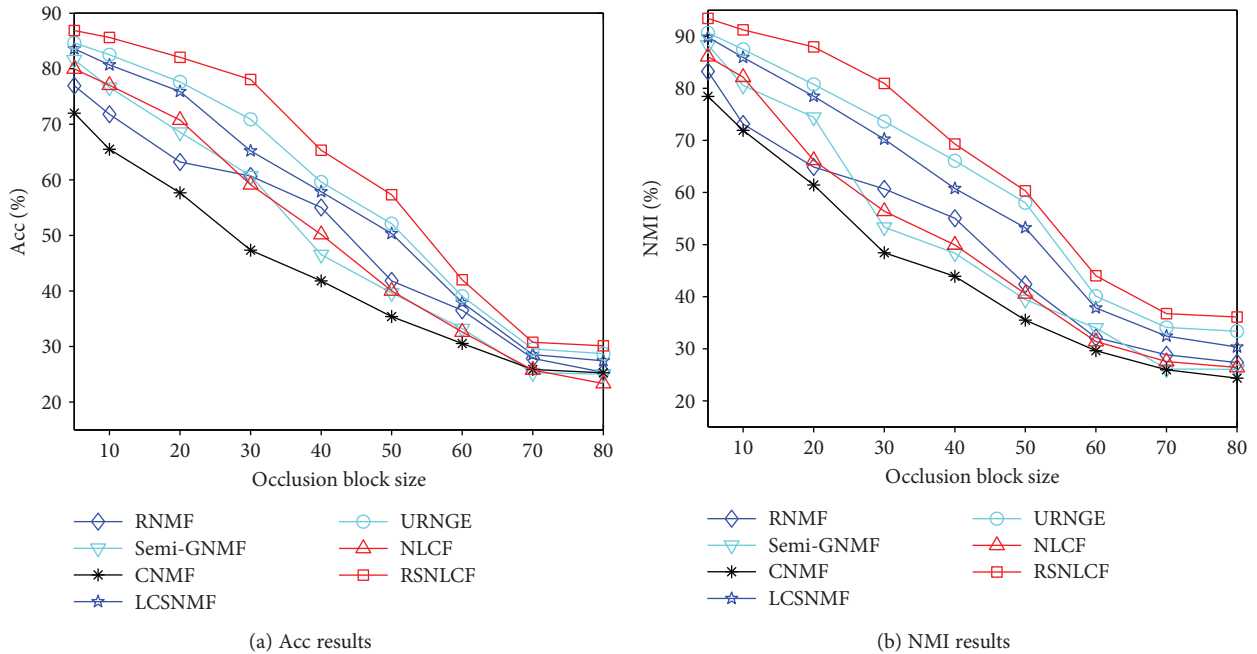


FIGURE 5: Clustering Acc and NMI curves of the compared methods under different occlusion levels with each image in the ORL dataset.

Note that because RNMF, LCSNMF, and NLCF are unsupervised algorithms, we did not compare them here. For each category of data, $c = 2, 3, 4, 5, 6, 7, 8$ samples were randomly chosen and labeled, with the remaining samples being unlabeled. As the samples were randomly selected, for each c , we repeated each experiment 20 times and calculated the average clustering accuracy. Figure 6 plots clustering Acc and NMI results of the compared methods under different

numbers of labeled samples. We can observe that our RSNLCF approach achieved the best clustering performance of all the compared approaches.

5.8. Parameter Sensitivity. In our proposed method, several parameters were tuned beforehand. We observed that RSNLCF is insensitive to τ in the range of $[10^{-3}, 10^3]$. Accordingly, we fixed η to be 10^6 and τ to be 10 for both the

TABLE 2: Clustering performances on the AR dataset.

r	Acc (%)				NMI (%)			
	Semi-GNMG	CNMF	URNGE	RSNLCF	Semi-GNMG	CNMF	URNGE	RSNLCF
4	58.29 ± 3.74	55.93 ± 3.72	63.28 ± 3.94	82.29 ± 2.54	68.65 ± 5.34	67.79 ± 3.51	72.37 ± 4.29	86.27 ± 2.28
6	64.53 ± 3.62	60.56 ± 4.54	69.06 ± 4.31	86.07 ± 4.19	72.49 ± 3.38	68.28 ± 4.43	76.85 ± 4.32	89.46 ± 4.93
8	67.37 ± 3.22	69.17 ± 3.93	74.19 ± 3.71	87.25 ± 2.68	78.13 ± 3.67	80.33 ± 3.25	81.24 ± 2.36	91.57 ± 2.28
10	74.87 ± 4.13	76.57 ± 2.76	79.73 ± 2.38	88.48 ± 3.85	83.48 ± 3.36	84.52 ± 2.37	87.63 ± 3.48	92.84 ± 3.58
12	79.38 ± 3.03	82.32 ± 2.83	88.01 ± 3.52	92.64 ± 2.23	85.21 ± 2.02	88.32 ± 3.76	90.27 ± 3.52	94.12 ± 2.93
14	85.95 ± 3.31	90.58 ± 2.53	91.49 ± 2.36	93.54 ± 2.17	88.27 ± 2.37	92.46 ± 2.43	92.34 ± 2.86	96.43 ± 3.54
16	86.39 ± 3.71	91.35 ± 4.54	92.92 ± 2.42	94.83 ± 2.79	89.38 ± 2.58	93.25 ± 3.38	93.94 ± 2.48	97.07 ± 2.72
18	88.83 ± 3.27	92.82 ± 4.55	94.73 ± 2.67	96.71 ± 2.35	91.03 ± 2.64	94.38 ± 3.46	95.52 ± 2.69	98.91 ± 2.76
Avg.	75.70	77.41	81.68	90.23	82.46	83.67	86.27	93.33

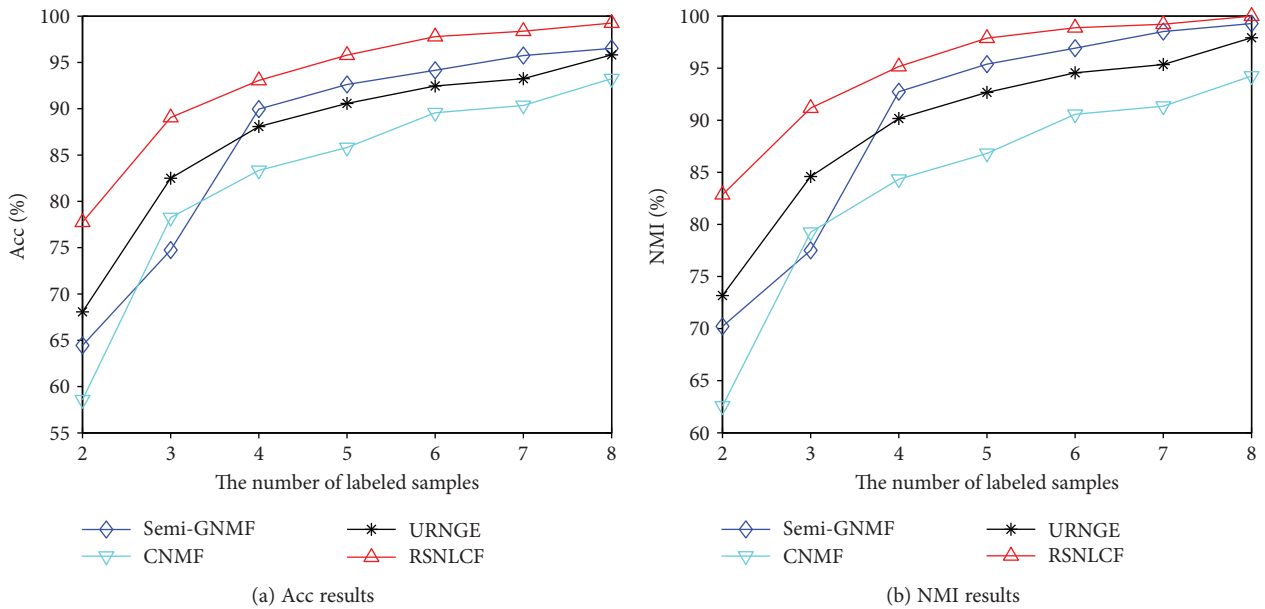


FIGURE 6: Clustering Acc and NMI curves of the compared methods under different numbers of labeled samples for the leukemia dataset.

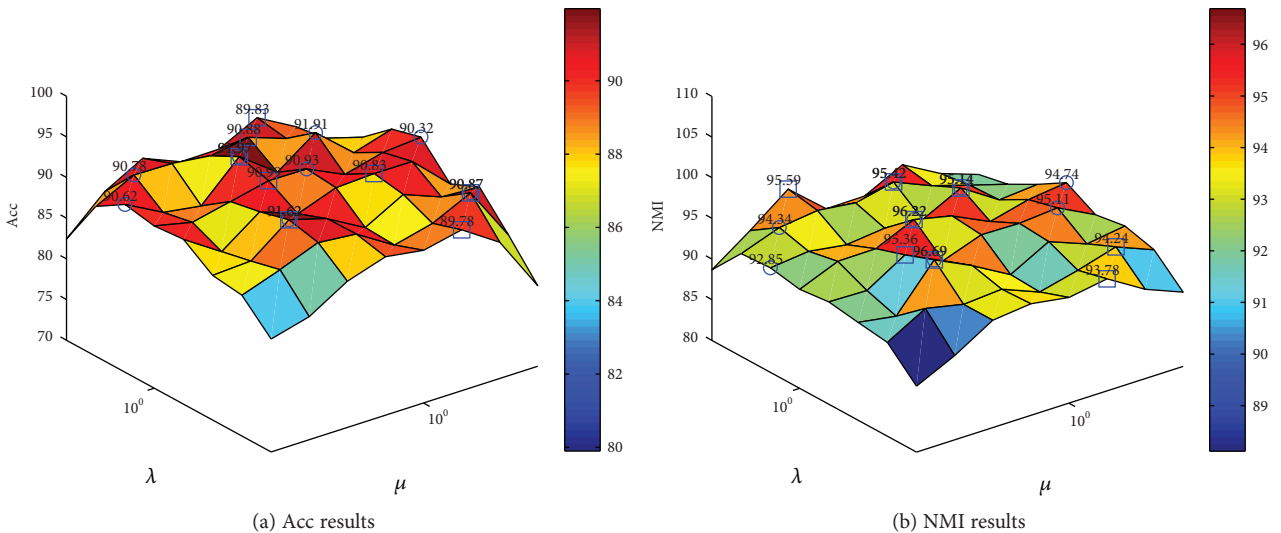


FIGURE 7: Clustering accuracy of the proposed method with respect to the parameters μ and λ on the extended YaleB dataset.

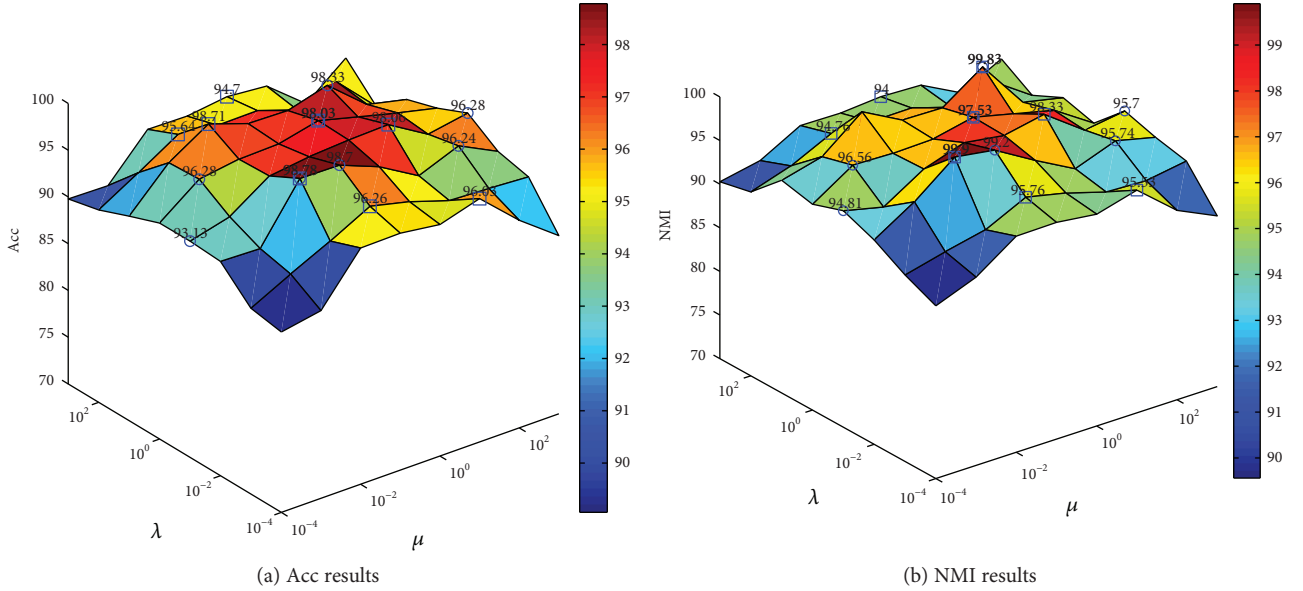


FIGURE 8: Clustering accuracy of the proposed method with respect to the parameters μ and λ on the leukemia dataset.

extended YaleB and leukemia datasets. To study the sensitivity of RSNLCF with respect to the remaining parameters (i.e., μ and λ), we varied these parameters. In the experiment, we plotted the Acc and NMI of RSNLCF with respect to μ and λ . Figures 7 and 8 show clearly the 3D results of RSNLCF. The horizontal axes are the parameters μ and λ , and the vertical axis represents the clustering accuracy of RSNLCF. In the 3D graphs, the square/circle marker indicates the best μ/λ for varying μ/λ . Next to each marker at the cross point is a digit number representing the value of Acc or NMI. We can notice from Figures 7 and 8 that the clustering performance varied with different combinations of μ and λ . However, it is unknown theoretically how to choose the best parameter. The regularization parameters should be associated with the characteristics of the dataset.

5.9. Convergence Analysis. In the previous section, we proved the convergence of our presented method. In our study, an experiment was performed to compare all algorithms' speed of convergence on the extended YaleB and leukemia datasets. The two parameters μ and λ were both fixed at 10. The time is measured using a computer with Intel Core™ I7 2600 and 16 GB memory. Figure 9 demonstrated the objective function value versus computational time for different algorithms. The horizontal and vertical axes here represent training times and the value of the objective function, respectively. We can observe from Figure 9 that the objective function value of all algorithms decreases steadily with the time increase, and RSNLCF requires less time than other graph-based methods, demonstrating that the proposed method was effective and efficient.

5.10. Overall Observations and Discussion. In our experiments, we considered several groups of experiments based on different databases, where the extended YaleB mainly involved illumination changes, the ORL database focused

on pixel corruptions and block occlusions, the AR database included face images with different facial variations, sunglasses, and scarf occlusions, and the leukemia dataset contained a large number of features but only a few samples. From the aforementioned experimental results, we gained the following attractive insights:

- (i) In most cases, the performance of CNMF was usually lower than that of the graph-based approach, which demonstrates the superiority of intrinsic geometrical structure representation in discovering potential discriminative information.
- (ii) Regardless of the datasets, our RSNLCF algorithm outperformed all six other methods. The reason lies in the fact that RSNLCF is designed for simultaneous application to local and global consistencies over labels simultaneously to uncover an underlying subspace structure. In addition, RSNLCF proved robust to outlier points and noises as a result of employing the $l_{2,1}$ norm formulations of NMF and the local coordinate constraint regularization term.
- (iii) Future research on this topic will include how to use multicore processors [48, 49] to accelerate our proposed method and how to extend the idea of semisupervised learning to the existing clustering algorithms.

6. Conclusion

In this study, we proposed a novel matrix decomposition method (RSNLCF) to learn an efficient representation for data in a semisupervised learning scenario. An efficient iterative algorithm for RSNLCF was also presented. The convergence of the presented method was theoretically proved. Extensive experiments over diverse datasets demonstrated

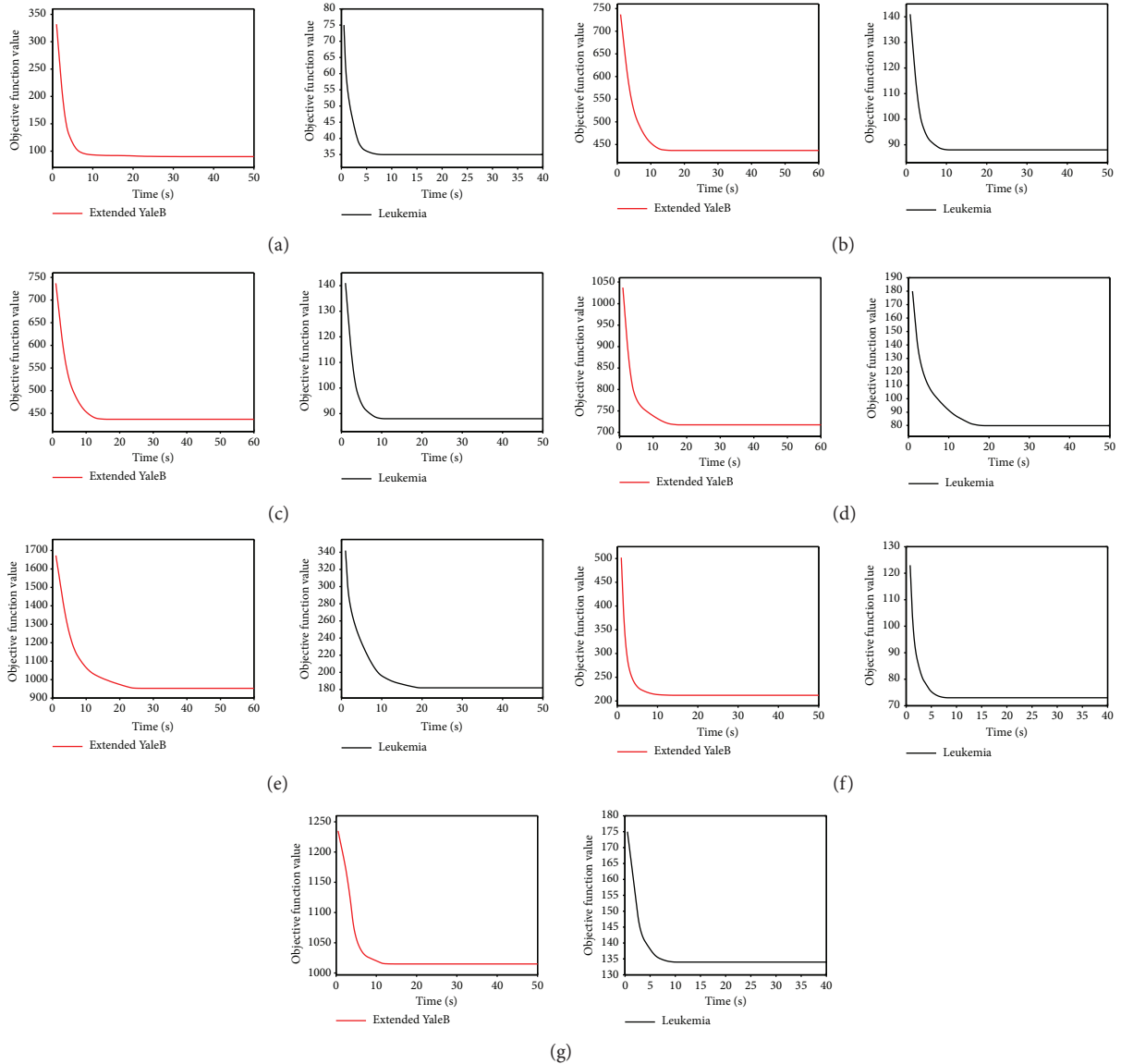


FIGURE 9: The curve of objective function value versus computational time on the extended YaleB and leukemia datasets. (a) RNMF, (b) semi-GNMF, (c) CNMF, (d) LCSNMF, (e) URNGE, (f) NLCF, and (g) RSNLCF.

that the presented method is quite effective and robust at learning an efficient data representation for clustering tasks. More importantly, experimental results revealed that our optimization algorithm quickly converges, indicating that our method can be utilized to solve practical problems.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work is supported by the Natural Science Foundation of Liaoning Province no. 2015020070 and the Natural Science Foundations of China no. 61771229, 61702243, and 61702245.

References

- [1] L. Qiao, S. Chen, and X. Tan, "Sparsity preserving projections with applications to face recognition," *Pattern Recognition*, vol. 43, no. 1, pp. 331–341, 2010.
- [2] H. Qi, K. Li, Y. Shen, and W. Qu, "Object-based image retrieval with kernel on adjacency matrix and local combined features," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 8, no. 4, pp. 1–18, 2012.
- [3] J. Wei, L. Min, and Z. Yongqing, "Neighborhood preserving convex nonnegative matrix factorization," *Mathematical Problems in Engineering*, vol. 2014, 8 pages, 2014.
- [4] P. Li, J. Bu, C. Chen, Z. He, and D. Cai, "Relational multimani-fold coclustering," *IEEE Transactions on Cybernetics*, vol. 43, no. 6, pp. 1871–1881, 2013.
- [5] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, Wiley, New York, NY, USA, 1973.

- [6] J. Zhao, L. Shi, and J. Zhu, "Two-stage regularized linear discriminant analysis for 2-D data," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 8, pp. 1669–1681, 2015.
- [7] Y. Gao, X. Wang, Y. Cheng, and Z. J. Wang, "Dimensionality reduction for hyperspectral data based on class-aware tensor neighborhood graph and patch alignment," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 8, pp. 1582–1593, 2015.
- [8] S. Yan, D. Xu, B. Zhang, H. J. Zhang, Q. Yang, and S. Lin, "Graph embedding and extensions: a general framework for dimensionality reduction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 1, pp. 40–51, 2007.
- [9] Y. Chen, J. Zhang, D. Cai, W. Liu, and X. He, "Nonnegative local coordinate factorization for image representation," *IEEE Transactions on Image Processing*, vol. 22, no. 3, pp. 969–979, 2013.
- [10] C. Hou, F. Nie, X. Li, D. Yi, and Y. Wu, "Joint embedding learning and sparse regression: a framework for unsupervised feature selection," *IEEE Transactions on Cybernetics*, vol. 44, no. 6, pp. 793–804, 2014.
- [11] Z. Ma, F. Nie, Y. Yang, J. R. R. Uijlings, and N. Sebe, "Web image annotation via subspace-sparsity collaborated feature selection," *IEEE Transactions on Multimedia*, vol. 14, no. 4, pp. 1021–1030, 2012.
- [12] F. Nie, H. Huang, X. Cai, and C. H. Ding, "Efficient and robust feature selection via joint $\ell_{1,2}$ -norms minimization," *Advances in Neural Information Processing Systems*, pp. 1813–1821, 2010.
- [13] Z. Li, J. Liu, Y. Yang, X. Zhou, and H. Lu, "Clustering-guided sparse structural learning for unsupervised feature selection," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 9, pp. 2138–2150, 2014.
- [14] L. Du and Y. D. Shen, "Unsupervised feature selection with adaptive structure learning," in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '15*, pp. 209–218, Sydney, NSW, Australia, 2015.
- [15] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, pp. 788–791, 1999.
- [16] H. Liu, Z. Wu, X. Li, D. Cai, and T. S. Huang, "Constrained nonnegative matrix factorization for image representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 7, pp. 1299–1311, 2012.
- [17] W. Xu and Y. Gong, "Document clustering by concept factorization," in *Proceedings of the 27th annual international conference on Research and development in information retrieval - SIGIR '04*, pp. 202–209, Sheffield, UK, 2004.
- [18] R. Zhi, M. Flierl, Q. Ruan, and W. B. Kleijn, "Graph-preserving sparse nonnegative matrix factorization with application to facial expression recognition," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 41, no. 1, pp. 38–52, 2011.
- [19] J. Ye and Z. Jin, "Dual-graph regularized concept factorization for clustering," *Neurocomputing*, vol. 138, pp. 120–130, 2014.
- [20] P. O. Hoyer, "Non-negative sparse coding. Neural Networks for Signal Processing, 2002," in *Proceedings of the 12th IEEE Workshop on Neural Networks for Signal Processing*, pp. 557–565, Martigny, Switzerland, 2002.
- [21] S. Zafeiriou, A. Tefas, I. Buciu, and I. Pitas, "Exploiting discriminant information in nonnegative matrix factorization with application to frontal face verification," *IEEE Transactions on Neural Networks*, vol. 17, no. 3, pp. 683–695, 2006.
- [22] C. H. Q. Ding, T. Li, and M. I. Jordan, "Convex and semi-nonnegative matrix factorizations," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 1, pp. 45–55, 2010.
- [23] H. Liu, Z. Yang, Z. Wu, and X. Li, "A-optimal non-negative projection for image representation," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1592–1599, Providence, RI, USA, 2012.
- [24] D. Cai, X. He, J. Han, and T. S. Huang, "Graph regularized nonnegative matrix factorization for data representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1548–1560, 2011.
- [25] A. Cichocki, R. Zdunek, and S. Amari, "Csiszár's divergences for non-negative matrix factorization: family of new algorithms," in *Independent Component Analysis and Blind Signal Separation. ICA 2006*, pp. 32–39, Springer, Berlin Heidelberg, 2006.
- [26] L. Zhang, Z. Chen, M. Zheng, and X. He, "Robust non-negative matrix factorization," *Frontiers of Electrical and Electronic Engineering in China*, vol. 6, no. 2, pp. 192–200, 2011.
- [27] H. Zhang, Z.-J. Zha, S. Yan, M. Wang, and T.-S. Chua, "Robust nonnegative matrix factorization using L21-norm," in *Proceedings of the 20th ACM international conference on Information and knowledge management*, pp. 673–682, Glasgow, Scotland, UK, 2011.
- [28] H. Zhang, Z.-J. Zha, S. Yan, M. Wang, and T.-S. Chua, "Robust non-negative graph embedding: towards noisy data, unreliable graphs, and noisy labels," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2464–2471, Providence, RI, USA, 2012.
- [29] C. Yan, H. Xie, D. Yang, J. Yin, Y. Zhang, and Q. Dai, "Supervised hash coding with deep neural network for environment perception of intelligent vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 1, pp. 284–295, 2018.
- [30] C. Yan, H. Xie, S. Liu, J. Yin, Y. Zhang, and Q. Dai, "Effective Uyghur language text detection in complex background images for traffic prompt identification," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 1, pp. 220–229, 2018.
- [31] W. Zhang, B. Ma, K. Liu, and R. Huang, "Video-based pedestrian re-identification by adaptive spatio-temporal appearance model," *IEEE Transactions on Image Processing*, vol. 26, no. 4, pp. 2042–2054, 2017.
- [32] W. Zhang, S. Hu, K. Liu, and J. Yao, "Motion-free exposure fusion based on inter-consistency and intra-consistency," *Information Sciences*, vol. 376, no. C, pp. 190–201, 2017.
- [33] X. Zhu, Z. Ghahramani, and J. D. Lafferty, "Semi-supervised learning using Gaussian fields and harmonic functions," in *Proceedings of the 20th International conference on Machine learning (ICML-03)*, pp. 912–919, Washington, DC, USA, 2003.
- [34] D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf, "Learning with local and global consistency," *Advance in Neural Information Processing Systems*, vol. 16, no. 16, pp. 321–328, 2003.

- [35] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000.
- [36] F. Wang and C. Zhang, "Label propagation through linear neighborhoods," *IEEE Transactions on Knowledge and Data Engineering*, vol. 20, no. 1, pp. 55–67, 2008.
- [37] S. Xiang, F. Nie, and C. Zhang, "Semi-supervised classification via local spline regression," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 11, pp. 2039–2053, 2010.
- [38] Y. Han, Y. Yang, Y. Yan, Z. Ma, N. Sebe, and X. Zhou, "Semisupervised feature selection via spline regression for video semantic recognition," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 2, pp. 252–264, 2015.
- [39] W. Zhang, C. Qu, L. Ma, J. Guan, and R. Huang, "Learning structure of stereoscopic image for no-reference quality assessment with convolutional neural network," *Pattern Recognition*, vol. 59, pp. 176–187, 2016.
- [40] W. Zhang, K. Liu, W. Zhang, Y. Zhang, and J. Gu, "Deep neural networks for wireless localization in indoor and outdoor environments," *Neurocomputing*, vol. 194, pp. 279–287, 2016.
- [41] K. Yu, T. Zhang, and Y. Gong, "Nonlinear learning using local coordinate coding," in *Advances in Neural Information Processing Systems*, pp. 2223–2231, MIT Press, 2009.
- [42] S. Xiang, F. Nie, C. Zhang, and C. Zhang, "Nonlinear dimensionality reduction with local spline embedding," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, no. 9, pp. 1285–1298, 2009.
- [43] J. Duchon, "Splines minimizing rotation-invariant seminorms in Sobolev spaces," in *Constructive Theory of Functions of Several Variables*, pp. 85–100, Springer, Berlin, Heidelberg, 1977.
- [44] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge university press, 2004.
- [45] W. Xu, X. Liu, and Y. Gong, "Document clustering based on non-negative matrix factorization," in *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval - SIGIR '03*, pp. 267–273, Toronto, Canada, 2003.
- [46] L. Lovsz and M. D. Plummer, *Matching Theory*, American Mathematical Society, 2009.
- [47] H. Gao, F. Nie, and H. Huang, "Local centroids structured non-negative matrix factorization," in *Thirty-First AAAI Conference on Artificial Intelligence (AAAI 2017)*, pp. 1905–1911, San Francisco, CA, USA, 2017.
- [48] C. Yan, Y. Zhang, J. Xu et al., "Efficient parallel framework for HEVC motion estimation on many-core processors," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 12, pp. 2077–2089, 2014.
- [49] C. Yan, Y. Zhang, J. Xu et al., "A highly parallel framework for HEVC coding unit partitioning tree decision on many-core processors," *IEEE Signal Processing Letters*, vol. 21, no. 5, pp. 573–576, 2014.

Research Article

AIRank: Author Impact Ranking through Positions in Collaboration Networks

Jun Zhang,¹ Yan Hu ,¹ Zhaolong Ning,¹ Amr Tolba ,^{2,3}
Elsayed Elashkar,^{4,5} and Feng Xia ¹

¹Key Laboratory for Ubiquitous Network and Service Software of Liaoning Province, School of Software, Dalian University of Technology, China

²Computer Science Department, Community College, King Saud University, Riyadh 11437, Saudi Arabia

³Mathematics and Computer Science Department, Faculty of Science, Menoufia University, Shebin-El-Kom 32511, Egypt

⁴Administrative Sciences Department, Community College, King Saud University, Riyadh 11437, Saudi Arabia

⁵Applied Statistics Department, Faculty of Commerce, Mansoura University, Mansoura 35516, Egypt

Correspondence should be addressed to Yan Hu; wohuyan@gmail.com

Received 30 November 2017; Revised 16 March 2018; Accepted 12 April 2018; Published 11 June 2018

Academic Editor: Xiuzhen Zhang

Copyright © 2018 Jun Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Citation is a universally acknowledged way for scientific impact evaluation. However, due to its easy manipulability, simply relying on citation cannot objectively reflect the actual impact of scholars. Instead of citation, we utilize the academic networks, in virtue of their available and abundant academic information, to evaluate the scientific impact of scholars in this paper. Through the collaboration among scholars in academic networks, we notice an interesting phenomenon that scholars in some special positions can access more kinds of information and connect researchers from different groups to promote the scientific collaborations. However, this important fact is generally ignored by the existing approaches. Motivated by the observations above, we propose the novel method AIRank to evaluate the scientific impact of scholars. Our method not only considers the impact of scholars through the mutual reinforcement process in heterogeneous academic networks, but also integrates the structural holes theory and information entropy theory to depict the benefit that scholars obtain via their positions in the network. The experimental results demonstrate the effectiveness of AIRank in evaluating the impact of scholars more comprehensively and finding more top ranking scholars with interdisciplinary nature.

1. Introduction

The development of modern research technologies allows researchers to get access to the plentiful scholarly data timely and facilitates the academic cooperation among scholars with diverse backgrounds. The easy access to the various scholarly data and the diverse data analysis technologies make researchers conduct their work more efficiently [1, 2]. However, due to the large volume of scholarly data, it is time-consuming to filter the influential and related scholars or references from the massive data. The evaluation of scientific impact not only sheds light on the above problem, but also provides basis for academic awards applications, faculty employments, fund decisions, etc. [3]. Therefore, evaluating

the scientific impact is of great significance, and our primary concern is on measuring the impact of scholars in this paper.

The existing evaluation methods generally prefer using the qualities and quantities of scholars' papers to measure the scientific impact. For a long time, citation has been widely used to gauge the influence of scholars and articles, such as h -index [4], g -index [5], and the journal impact factor [6]. However, some crucial shortcomings exist with such approaches that heavily rely on citation counts. The first problem is that the accumulation process of citation counts is involved with time. Therefore, previously published papers obviously have the advantage of having longer time cited by other literature than newly published papers. Another existing problem is that the citation counts can be easily

manipulated through self-citations or citations via acquaintanceships. As a consequence, citation counts cannot accurately reflect the qualities of scholarly articles to some extent.

Apart from the citation-based methods, researchers also utilize the academic networks to measure the scientific impact. Typical academic networks include various kinds of entities and relationships, such as papers, authors, venues, citation relationship, and coauthorship. Therefore, by considering the above-mentioned attributes of heterogeneous networks, it is obvious that using heterogeneous network topology [7] to depict the academic networks is more suitable than applying homogeneous network topology. The PageRank [8] and HITS algorithms [9] are the most commonly used ones to rank the importance of scholarly entities in academic networks. Considering the distinct importance of different entities and relationships in academic networks, researchers have proposed a number of weighting schemes, together with the variants of PageRank or HITS algorithm, to evaluate the scientific impact in academic networks [10].

Academic networks have been widely employed for scientific impact evaluation in the above-mentioned network-based methods. It not only provides plentiful information about scholarly entities, but also explicitly indicates relationships among them [11]. Under the coauthor network structure, we find that scholars that possess some special positions can access diverse information from various kinds of scholars and act as bridges that connect different groups of scholars. These scholars can benefit from the various information, and consequently their research capacities can be improved. In addition to the gains that these positions bring to the scholars themselves, they also accelerate the dissemination of knowledge among scholars in different fields. Simultaneously, the communications between scholars also promote the interdisciplinary collaborations and, furthermore, propel the development of science. Therefore, the effect of scholars' positions is of great significance for the evaluation of scholars' impact.

Although current works have proposed many solutions on evaluating the scientific impact, they mainly ignore the vital effect of scholars' positions on their impact. In this paper, we propose the AIRank to evaluate scholars' impact. In order to measure the overall scientific impact of scholars, our method considers the scholar's impact in heterogeneous academic networks through the mutual influence mechanism among academic entities and combines this with the effects of scholars' positions in the network.

To investigate the effects of scholars' positions in the network on their impact, we look into this question from the angle of sociology. In sociology, the structural holes theory [12] indicates that the positions of individuals in the networks are closely related to their benefits. The structural holes theory suggests that individuals can access richer information and let the disconnected people know each other through them if they are in the positions that act as bridges between different groups of individuals. Figure 1 shows an illustration of the structural holes theory; the nodes represent scholars from different domains in computer science area. It is obvious that the red node in the center can connect and cooperate with scholars from different domains. Therefore, when facing problems, researchers can apply ideas and techniques

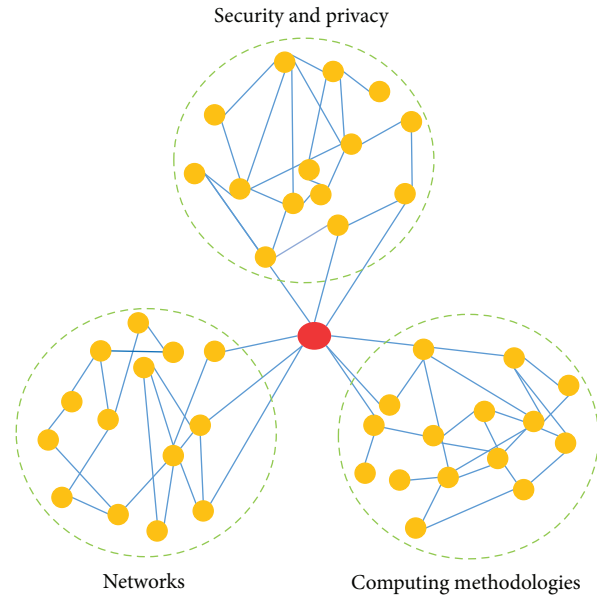


FIGURE 1: Illustration of structural holes.

obtained from other groups to solve them if they span structural holes. Several studies have indicated that the social success is positively correlated with the structural holes [13]. Thus we apply the structural holes theory to depict the importance of scholars' positions and their abilities on both accessing rich information and connecting different researchers.

To explore the diversity of information that researchers obtained, we solve this issue by considering the diverse backgrounds of coauthors. Researchers can directly acquire information or ideas from their coauthors due to the close cooperation in publishing scholarly articles. Hence the varieties of coauthors' backgrounds can indicate researchers' abilities to acquire diverse information. Besides acquiring information through the direct connections with coauthors, another way is the attendance of academic activities. Researchers can encounter other scholars and may further establish cooperation relationship through attending academic activities [14]. Scholars publishing articles in conferences have the opportunities to make acquaintance with other people through the attendance. Therefore, the quantities and qualities of articles published in conferences can represent the diverse information researchers acquire to some extent, and we utilize them to represent the diversity of information that researchers can acquire.

Generally speaking, we make the following contributions in this paper.

- (i) *New Insight into Scientific Impact Evaluation.* We creatively provide a new solution to solve the impact evaluation issues from the angle of scholars' network positions for the first time, to the best of our knowledge.
- (ii) *Novel Features for Evaluating Scholars.* We present three new indicators through utilizing the structural holes theory and information entropy theory to depict the effects of scholars' positions in collaboration

networks and furthermore integrate the interplay among diverse scholarly entities in heterogeneous academic networks together to quantify scholars' scientific impact.

- (iii) *Effectiveness in Identifying Outstanding Interdisciplinary Scholars.* The experiments on real datasets verify the significant role of scholars' positions in their impact, and our method outperforms the state-of-the-art methods in evaluating scholars' impact more comprehensively and identifying more outstanding interdisciplinary scholars.

The rest of the paper is organized as follows. Related work is discussed in the next section. Section 3 formulates the studied problem of scholar's scientific impact evaluation. Section 4 introduces our proposed method. Section 5 presents the experimental results of our method, followed by a section dedicated to the conclusion.

2. Related Work

The problem of scientific impact evaluation has been studied for a long time and became a popular and significant research direction [15–17]. The evaluation of scientific impact can assist scholars in diffusing their work and maximizing the academic influence [18, 19]. Generally, there are two major kinds of methods for measuring scholars' scientific impact, i.e., citation-based methods and network-based methods. In this section, we survey the existing literature in the above areas, respectively.

2.1. Citation-Based Methods. The achievements of scholars are often represented by their articles; therefore, the qualities of articles are usually used to measure the scientific impact of scholars. To measure the qualities of articles, the citation counts are one of the most widely used indicators. A series of metrics has been put forward to measure the scientific impact according to citations. Initially, the journal impact factor is proposed for evaluating the quality of journals [6]. Continually, the h -index [4] is proposed to measure scholar's impact by considering the productivity and the quality of their research work. Moreover, Pan and Fortunato [20] proposed the AIF to depict the dynamics of scholars' impact by considering the ever-increasing characteristic of h -index. These works all successfully depict the scientific impact and are commonly used due to the uncomplicated calculation process.

However, there exist critical shortcomings of using citation counts to evaluate the impact of scholars. The first problem is citation counts aggregate with time. Therefore, it is obvious that articles published for a long period have the advantage of occupying more time for citations than newly published articles. Similarly, using the same time interval to evaluate the scientific impact is unfair for young researchers comparing to senior researchers. Considering the above facts, researchers have proposed several methods to alleviate the effects of publishing time [21]. In addition, citations take time to happen; therefore, it cannot reflect the current impact of scholars timely.

Another problem existing in citation counts metrics is that citation counts can be distorted by self-citations or citations from colleagues, etc. Therefore, some researchers argue that the diverse citations should be considered disparately instead of regarding them equally [22]. Motivated by this observation, scholars have proposed diverse methods to differentiate the importance of citations. Valenzuela et al. [23] determined the significance of citations based on their appearing sections. Bai et al. [24] proposed a COIRank method to distinguish the conflict of interest citation relationship when measuring the impacts of articles. Other researches considered different aspects, such as citation distribution and coercive induced self-citation, to assess the qualities of citations.

2.2. Network-Based Methods. Considering the drawbacks of citation-based metrics, another way of measuring the impact of scholars is the network-based methods. Typically, the academic networks contain several main entities and relationships, e.g., articles, authors, venues, citing relationship, and coauthorship. Researchers have proposed a variety of ranking algorithms to gauge scholars' impact based on academic networks [25, 26].

A series of network-based methods has been proposed through calculating the degrees of scholars in academic networks by different methods to measure the impact of scholars. For instance, degree centrality, closeness centrality, Katz-Bonacich centrality, and eigenvector centrality are the commonly used measures to calculate the degrees of scholars based on different network structures [27, 28]. In addition, due to the merits of different measurements, researchers also integrate them together to quantify the scientific impact [29, 30].

Except for the above-mentioned centrality measurements, researchers also apply the commonly known ranking algorithms, i.e., the PageRank algorithm and HITS algorithm, to evaluate the scientific impact of scholars [31]. Previous researches utilize the PageRank and HITS algorithms to quantify the impact of scholars in homogeneous network. While the real academic networks contain various kinds of entities and links, diverse evaluation metrics have been proposed using different heterogeneous academic networks because of their topological merits. Figure 2 shows an illustration of a heterogeneous academic network; articles can be linked through citation relationship; authors can be linked to the articles they write; articles can be linked to the venues they published on; and authors can be related through the coauthorship.

Based on the above-mentioned heterogeneous academic network structures, researchers have proposed a series of the PageRank and HITS algorithms based methods to evaluate the impact of scholars. Considering the various kinds of relations that might exist among different entities, researchers have constructed distinct academic networks that contain novel relationships to measure the impact of scholars. A major kind of network-based methods is extending the original PageRank and HITS algorithms, which primarily focus on exploring new weights of the entities and links in the networks by considering the diverse importance of

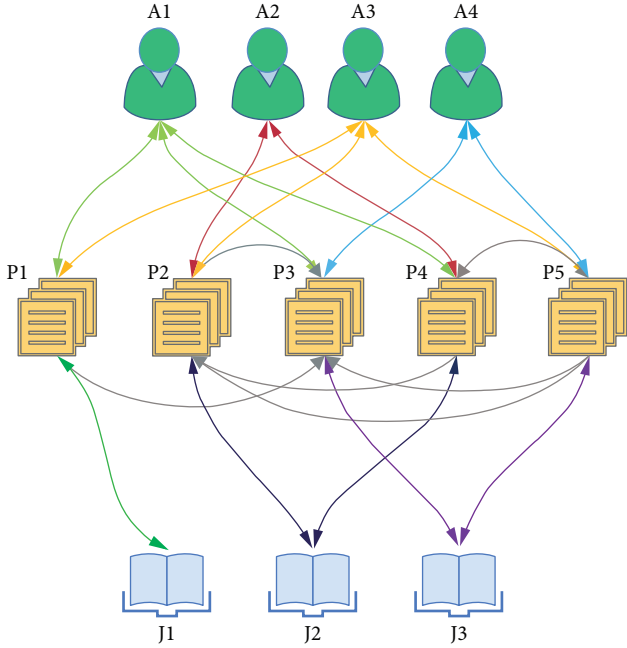


FIGURE 2: Illustration of a heterogeneous academic network, where P1 to P5 represent the papers, A1 to A4 indicate their corresponding authors, and J1, J2, and J3 represent the venues.

them. There also exist some works that utilize the PageRank algorithm and the HITS algorithm in the meantime to find the more appropriate one [32]. Instead of applying single kind of algorithm, researchers also combine the PageRank and HITS algorithm together to measure the scientific impact in order to utilize the advantages of both algorithms.

The primary mechanism of PageRank and HITS algorithm is that nodes would have higher influence value if the nodes that point to them are influential through the iterative process. Therefore, there exist mutual effects among the entities in the networks through the links. For instance, papers would become influential if they are cited by other articles with high qualities in the citation network, while the corresponding authors would be ranked high in the paper-author network, respectively. Several studies have been carried out on jointly evaluating the impact of scholars, articles, and venues according to specific academic networks. Based on the journals' impact, Nykl et al. [10] proposed an author ranking system through utilizing the PageRank algorithm. Considering the diverse research topics, Amjad et al. [33] measured the impact of scholarly entities by the topic-based heterogeneous rank in academic networks.

In addition, researchers also combine the citation and network-based evaluation metrics together to measure the impact of scholars because using single type of indicators is unable to capture the impact of scholars comprehensively. Wang et al. [34] explored the effect of citations, time information, and the combination of PageRank and HITS algorithm to quantify the scientific impact of scholars. Furthermore, Wang et al. [26] proposed the MRCoRank, which integrates the text features and HITS algorithm to determine the impact of scholars.

However, one important fact has been ignored by the existing approaches, that is, the effects of scholars' positions in the network and their abilities to acquire multiplicities of information via the existing relationships on their own impact. It is universally acknowledged that scholarly articles commonly represent the cooperation achievements of several coauthors; therefore, scholars can be influenced through the coauthorship. Although some researchers have investigated that scholars' impact can be affected by their coauthors' abilities [35, 36], no prior work exists to explore the influence of researchers' positions in the network and their capacities of obtaining diverse information on evaluating the scientific impact.

3. Problem Formulation

Generally, the task of scientific impact evaluation is formulated as statistical analysis problems or importance ranking algorithms. However, such existing approaches tend to evaluate scholars within the same disciplines and may be incapable of capturing the increasing interdisciplinary collaborations among researchers. Meanwhile, some scholars have noticed that the interactions among researchers can promote the quality and quantity of scientific achievements. Inspired by this interesting phenomenon, we propose a novel method which can identify influential scholars with interdisciplinary nature, thus formulating the following task: given the detailed information of scholars' publications, we evaluate the scientific impact of scholars with our proposed indicators implying their interdisciplinary collaborations in heterogeneous academic networks.

To solve our task, we decompose it into three subtasks. We first extract the coauthor network according to the information of scholars' publications. Let $G_c(V_{a_i}, E_{a_{ij}})$ denote the coauthor network, where V_{a_i} represents the node, and $E_{a_{ij}}$ exists if a_i has cooperated with a_j . Under the coauthor network, we then define and calculate several indicators $\{x_1, x_2, x_3, \dots, x_n\}$ of scholars. Based on the above analysis, the first subtask can be formalized as follows: given that an undirected graph $G_c(V_{a_i}, E_{a_{ij}})$ represents the cooperation relationships among researchers and given a set of factors $\{x_1, x_2, x_3, \dots, x_n\}$ of scholars, a function $f(a_i)$ that calculates the benefits of scholars through their positions in the network can be obtained.

Considering the overall task is to quantify scholars' scientific impact, we then compute the importance of scholars in heterogeneous academic networks. In order to fulfil this subtask, three academic networks need to be built. Let $G_{cit}(V_{p_i}, E_{p_{ij}})$ indicate the citation network, where V_{p_i} represents the node and $E_{p_{ij}}$ exists if p_i has cited p_j . From the citation network, the importance of scholars' corresponding papers can be obtained. Based on the values of papers, the importance degrees of corresponding venues and scholars can be calculated in paper-venue network ($G_{pv}(V_{p_i} \cup U_{v_j}, E_{p_i v_j})$) and paper-author network ($G_{pa}(V_{p_i} \cup U_{a_j}, E_{p_i a_j})$), respectively. V_{p_i} represents the paper, U_{v_j} is the publishing venue of papers, and $E_{p_i v_j}$ exists if p_i has been published on v_j . Similarly, U_{a_j} is the author of papers, and $E_{p_i a_j}$ exists if

p_i was written by a_j . In this part, we study scholars' importance in heterogeneous academic networks, formally defined as follows: given directed graphs $G_{cit}(V_{p_i}, E_{p_{ij}})$, $(G_{pv}(V_{p_i} \cup U_{v_j}, E_{p_i, v_j}))$, and $(G_{pa}(V_{p_i} \cup U_{a_j}, E_{p_i, a_j}))$ and a set of intermediate results $\{r_1, r_2, r_3, \dots, r_n\}$ obtained from the above-mentioned networks, a function $g(a_i)$ that calculates the importance of scholarly entities in heterogeneous academic networks can be obtained.

Our main purpose is to gauge the scientific impact of scholars. According to the above-mentioned subtasks, the final scientific impact can be obtained and formalized as follows.

Input. This includes the results obtained from functions $f(a_i)$ and $g(a_i)$.

Output. This includes the overall scientific impact of scholars.

The scientific impact evaluation problem we solve in this paper is formulated to be distinct from the traditional problem of simply relying on citation counts or network-based evaluation metrics. We explore the effect of scholars' network positions on the scientific impact. The primary advantage of our formulation is transforming the complex problem into three subtasks with low computational complexity, so that the efficiency of our method can be improved.

4. Design of AIRank

In most previous works, scholars are evaluated in the same time interval and their academic ages are commonly ignored. However, it is unfair for young researchers to be evaluated in the same time period compared to senior researchers. As a consequence, we choose scholars with the same academic age for evaluation to alleviate the effects of different research lengths. The real academic networks include various kinds of entities and relationships; therefore, we employ the heterogeneous network topology to represent academic network in order to depict it more appropriately.

The structural holes theory can indicate scholars' abilities to connect different people; therefore we utilize it in our method to depict scholars' positions in the network. To capture the multiplicities of information that researchers acquire through their relationships with other people, we measure these multiplicities from two aspects, which are the diversity of their coauthors and the quantity and quality of academic conferences they attend. In addition, we also consider the mutual effects among different academic entities in the networks together to quantify scholars' scientific impact.

Our proposed method consists of three main steps, the architecture of which is shown in Figure 3. The first part is calculating scholar's structural index (SI) value which captures the effect of scholars' positions in the networks. Three factors are proposed and the structural holes theory is employed in SI. In addition, we also consider the impact of scholars in academic networks through our proposed network index (NI). We apply the PageRank and HITS algorithms to measure scholars' impact in the three constructed academic networks. Finally, considering the above two parts, the overall impact of scholars is calculated according to the

final formula. The calculation procedure of our proposed AIRank is shown as follows.

Step 1. Calculate the value of SI, which consists of the three proposed indicators and will be introduced in detail in the following.

Step 2. Calculate the value of NI, which utilizes the PageRank and HITS algorithms together to measure the impact of scholars in the networks.

Step 3. Calculate scholar's final score according to the above two steps.

4.1. Calculation Procedure of SI. With the development of research techniques, researchers nowadays can easily trace the studies of scholars from related areas and keep up with the research trends. Due to the convenience of the Internet, scholars can establish cooperation relationships even though they may never meet before in reality. Consequently, interdisciplinary cooperation happens more frequently than in the past, and the positions of scholars in the network play an important role in promoting the collaborations. The academic collaborations among diverse domains accelerate the advancements of science; meanwhile, researchers can also obtain information or techniques through the collaborations with diverse researchers.

4.1.1. Scholars' Structural Holes Measurements. To depict scholars' positions in the network, we first apply the structural holes theory. The main principle of structural holes is that people would benefit more if they are in the positions that can link people from different groups. Typically, there are several ways of measuring the structural holes; we apply the most commonly used measurements which are the bridge counts and the betweenness centrality. To find the appropriate measures of structural holes for our algorithm, we apply the above methods, respectively, in the calculation of SI to evaluate their performances. The specific calculation processes are illustrated as follows.

Bridge Counts. It is an intuitively appealing measure. The link between two people is a bridge if there are no indirect connections between the two people. Equation (1) indicates the calculation formula:

$$\text{BrC}(a_i) = \sum_{a_j=1}^{n-1} b_{ij} \quad (1)$$

where $\text{BrC}(a_i)$ is the total number of bridges between authors a_i and a_j ; n is the number of authors in the network. If there exists a bridge between a_i and a_j , the value of b is 1; otherwise, the value of b_{ij} is 0.

Betweenness Centrality. The betweenness centrality is the count of the structural holes to which a person has monopoly access. Given that a network contains n nodes, the maximum possible value for node is degree which is $n - 1$, and the maximum possible value for its betweenness centrality equals

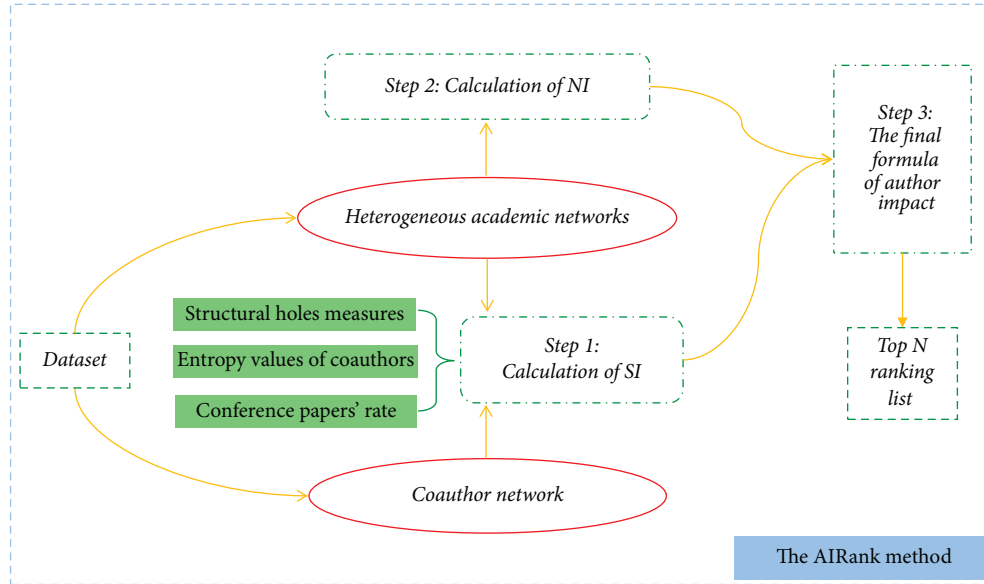


FIGURE 3: Architecture of AIRank.

the hub node which is betweenness centrality value in a star network. More specifically, the shortest path between all the other node pairs is unique and definitely via the hub node. Therefore, node is betweenness centrality value this is the sum of all the above-mentioned shortest paths which equals the following formula:

$$\frac{(n-1)(n-2)}{2} = \frac{n^2 - 3n + 2}{2} \quad (2)$$

Based on the above equation, the normalized betweenness centrality is defined as follows:

$$\text{BeC}(a_i) = \frac{2}{n^2 - 3n + 2} \sum_{a_i \neq a_s \neq a_t} \frac{N_{a_{st}}^{a_i}}{g_{a_{st}}} \quad (3)$$

where $\text{BeC}(a_i)$ is the betweenness centrality value of author a_i , n is the size of the network, $g_{a_{st}}$ is the number of the shortest paths from a_s to a_t , and $N_{a_{st}}^{a_i}$ is the number of shortest paths that go through author a_i .

4.1.2. Diversity of Cooperators. The scholarly articles usually are the collective efforts of several coauthors, and researchers can benefit a lot from their coauthors through the cooperation relationship. Research ideas or techniques can be exchanged among coauthors through the collaboration process; as a consequence, scholars' academic achievements can be affected by the information they acquired and the people they interact with. Previous studies have investigated that researchers' academic level can be influenced by their coauthors' impact; however, the effect of the diversity of information and scholars that researchers accessed still needs to be explored.

To capture the variety of information, we consider two apparent information sources that researchers directly contact with. The first one is acquiring information through

their cooperators. Ideas, problems, or techniques can be discussed and shared through the collaborative working towards publishing scholarly articles among coauthors. Therefore, the background of a scholar can represent the variety of information he or she commands. As in our previous work [37], the theory of entropy is utilized in measuring the diverse backgrounds of cooperators which only considers the differences between institutions, while in this work we not only think about the differences of institutions, but also take the distinctions of research interests into consideration. The calculation process is as follows:

$$\text{Div}(a_i)_{\text{inst}} = - \sum_{m=1}^r w_m \log_2(w_m) \quad (4)$$

$$\text{Div}(a_i)_{\text{key}} = - \sum_{\rho=1}^q k_\rho \log_2(k_\rho) \quad (5)$$

$$\text{Div}(a_i) = \text{Div}(a_i)_{\text{inst}} + \text{Div}(a_i)_{\text{key}}$$

where $\text{Div}(a_i)_{\text{inst}}$ and $\text{Div}(a_i)_{\text{key}}$ represent the diversities of cooperators' institutions and their papers' keywords of author a_i , and $\text{Div}(a_i)$ is the overall cooperators' diversities of author a_i . w_m is the frequency of occurrences of word m in the combination of words extracted from the institutions' information of a_i 's collaborators, and r is the total amount of word m in (4). k_ρ is the frequency of occurrences of word ρ in all the papers' keywords of a_i 's collaborators, and q is the sum of words ρ .

4.1.3. Benefit Obtained via Academic Conferences. Another universal way of getting information is through attending academic conferences. Researchers publishing articles in the same conference commonly have similar research interests, and they can share their ideas or exchange information

```

Step 1 SI ( $r, w_m, q, k_\rho, S(C), \text{Num}_{a_i}^{\text{conf}}, \text{Num}_{a_i}^p, b$ )
(01) for  $m \leftarrow 1$  to  $r$  do
(02)    $\text{Div}(a_i)_{\text{inst}} \leftarrow -(\text{Div}(a_i)_{\text{inst}} + w_m \log_2(w_m))$ 
(03) end for
(04) for  $\rho \leftarrow 1$  to  $q$  do
(05)    $\text{Div}(a_i)_{\text{key}} \leftarrow -(\text{Div}(a_i)_{\text{key}} + k_\rho \log_2(k_\rho))$ 
(06) end for
(07)  $\text{Div}(a_i) \leftarrow \text{Div}(a_i)_{\text{inst}} + \text{Div}(a_i)_{\text{key}}$ 
(08) for  $v \leftarrow 1$  to  $t$  do
(09)    $\text{temp} \leftarrow \text{temp} + S(C_v)$ 
(10) end for
(11)  $\text{Bene}(a_i) \leftarrow \frac{\text{Num}_{a_i}^{\text{conf}}}{\text{Num}_{a_i}^p} \text{temp}$ 
(12) for  $a_j \leftarrow 1$  to  $n - 1$  do
(13)    $\text{BrC}(a_i) \leftarrow \text{BrC}(a_i) + b$ 
(14) end for
(15)  $\text{SI}_{a_i}^{\text{BrC}} = \frac{1 - \chi - \psi - \varphi}{n} + \chi Z_{\text{Div}(a_i)} + \psi Z_{\text{Bene}(a_i)} + \varphi Z_{\text{BrC}(a_i)}$ 
(16)  $\text{SI}_{a_i}^{\text{BeC}} = \frac{1 - \tau - \lambda - \varepsilon}{n} + \tau Z_{\text{Div}(a_i)} + \lambda Z_{\text{Bene}(a_i)} + \varepsilon Z_{\text{BeC}(a_i)}$ 

```

ALGORITHM 1

through attending the conference unlike publishing journal articles. Therefore, researchers can benefit a lot through participating in academic conferences, and the benefit that researchers get is captured by the following equation:

$$\text{Bene}(a_i) = \frac{\text{Num}_{a_i}^{\text{conf}}}{\text{Num}_{a_i}^p} \times \sum_{v=1}^t S(C_v) \quad (6)$$

where $\text{Bene}(a_i)$ represents a_i 's benefit obtained through attending academic conferences, $\text{Num}_{a_i}^{\text{conf}}$ is the number of conference papers that author a_i published, and $\text{Num}_{a_i}^p$ is the total number of published papers of author a_i . $S(C_v)$ is the impact value of the conferences (C_v) that author a_i published papers in, and t is the total number of v . The value of $S(C_v)$ equals its PageRank value in the paper-venue network.

4.1.4. Final Formula of SI. In this paper, we propose three factors to measure the effect of scholars' positions in the networks, which are scholars' structural holes values, the diversity of coauthors, and the benefits obtained via academic conferences. The pseudocode of SI is shown in Algorithm 1, and its specific calculation procedure is illustrated as follows.

Step 1. Calculate scholars' structural holes values, which exist with two ways of calculation (bridge counts and betweenness centrality).

Step 2. Calculate the diversity of coauthors, which utilizes the concept of information entropy to measure the diversity of scholars' cooperators.

Step 3. Calculate the benefits researchers obtained through attending academic conferences.

Step 4. Calculate scholar's final SI values, which exist in two ways ($\text{SI}_{a_i}^{\text{BrC}}$ and $\text{SI}_{a_i}^{\text{BeC}}$), according to the normalized above-mentioned factors.

The calculation procedure of $\text{Div}(a_i)$, $\text{Bene}(a_i)$, $\text{BeC}(a_i)$, and $\text{BrC}(a_i)$ can be obtained based on the above equations. While these three indicators cannot be arithmetically operated directly due to their different scales, therefore, we need to normalize them before the calculation process. The normalization process is shown as follows:

$$Z_i = \frac{v_i - \min_A}{\max_A - \min_A} (\text{new}_{\max_A} - \text{new}_{\min_A}) + \text{new}_{\min_A} \quad (7)$$

where A is the set of scholars' attributes, which includes the $\text{Div}(a_i)$, $\text{Bene}(a_i)$, $\text{BeC}(a_i)$, and $\text{BrC}(a_i)$. \max_A is the maximum value and \min_A is attribute A 's minimum value. v_i is attribute A 's original value, and Z_i is the normalization value of v_i in the range of $[\text{new}_{\min_A}, \text{new}_{\max_A}]$, which equals $[0, 1]$.

To find the appropriate measures of structural holes for our algorithm, we apply $\text{BeC}(a_i)$ and $\text{BrC}(a_i)$, respectively, in the SI method to find the most efficient one. Therefore, the overall assessment of scholars' abilities to acquire diverse information and their positions in the networks can be interpreted in two ways ($\text{SI}_{a_i}^{\text{BrC}}$ and $\text{SI}_{a_i}^{\text{BeC}}$) through the following equations:

$$\begin{aligned} \text{SI}_{a_i}^{\text{BrC}} &= \frac{1 - \chi - \psi - \varphi}{n} + \chi Z_{\text{Div}(a_i)} + \psi Z_{\text{Bene}(a_i)} \\ &\quad + \varphi Z_{\text{BrC}(a_i)} \\ \text{SI}_{a_i}^{\text{BeC}} &= \frac{1 - \tau - \lambda - \varepsilon}{n} + \tau Z_{\text{Div}(a_i)} + \lambda Z_{\text{Bene}(a_i)} \\ &\quad + \varepsilon Z_{\text{BeC}(a_i)} \end{aligned} \quad (8)$$

where χ , ψ , ι , λ , ε , and φ are parameters; $SI_{a_i}^{\text{BrC}}$ and $SI_{a_i}^{\text{BeC}}$ represent the value of SI_{a_i} which utilize BrC and BeC, respectively, to measure the positions of scholars in the network. $Z_{\text{Div}(a_i)}$, $Z_{\text{Bene}(a_i)}$, $Z_{\text{BrC}(a_i)}$, and $Z_{\text{BeC}(a_i)}$ are the normalization value of $\text{Div}(a_i)$, $\text{Bene}(a_i)$, $\text{BrC}(a_i)$, and $\text{BeC}(a_i)$ according to (7).

4.2. Calculation Procedure of NI. The next procedure of our method is measuring the influence of scholars in heterogeneous academic networks through utilizing the PageRank and HITS algorithms. Considering the mutual influence among academic entities through different relationships in the networks, we construct three academic networks to evaluate the scientific impact, i.e., the citation network, the paper-venue network, and the paper-author network.

- (i) Citation network: it contains one type of entities and relationships, i.e., papers, and the citation relationship among them.
- (ii) Paper-venue network: it composes two kinds of nodes and one kind of relationships. The nodes in the network are the papers and venues, and the publication relationship links the papers and their corresponding venues.
- (iii) Paper-author network: it consists of two kinds of entities, which are papers and their corresponding authors. Only one type of relationships is included in this network which depicts the writing relationship between papers and their authors.

We first apply the original PageRank algorithm to evaluate the importance score of articles in the citation network. According to this, the initial importance of papers in the citation network can be obtained. Then we calculate the impact of venues and authors in the constructed paper-venue network and paper-author network, respectively, through using the HITS algorithm, and we set the initial value of the entities in the networks accordingly. The pseudocode of NI is shown in Algorithm 2, and its specific calculation procedure is conducted as follows:

- (1) The initial value of publications is set as $1/N$, where N is the total number of articles in the network.
- (2) Calculate the scores of papers through utilizing the PageRank algorithm in the citation network.
- (3) Calculate the scores of papers and the corresponding venues in the paper-venue network by HITS algorithm; the initial values of papers are set according to their PageRank scores obtained in the above step.
- (4) Calculate the scores of scholars in the paper-author network through the HITS algorithm; the initial values of papers are set according to their values obtained from Step (3).
- (5) Repeat Steps (2)–(4) until convergence is encountered.

4.2.1. Article's Score in Citation Network. Initially, the PageRank algorithm is proposed to evaluate and rank the importance of webpages since there may pop up many searching

```

Step 2 NI (S, U, Pr,  $\alpha$ , h)
(01)  $G \leftarrow \alpha S + \frac{1-\alpha}{n} U$ 
(02) for  $i \leftarrow 0$  to  $n$  do
(03)   pr_next  $\leftarrow$  GPr
(04)   Pr  $\leftarrow$  Pr_next
(05) end for
(06)  $a \leftarrow$  copy(Pr)
(07) for  $i \leftarrow 0$  to  $n$  do
(08)   for  $i \leftarrow 0$  to  $n$  do
(09)      $h_i \leftarrow h_i + a_i$ 
(10)      $h_i \leftarrow \frac{h_i}{\max(h_i)}$ 
(11)   end for
(12)   for  $i \leftarrow 0$  to  $n$  do
(13)      $a_i \leftarrow a_i + h_i$ 
(14)      $a_i \leftarrow \frac{a_i}{\max(a_i)}$ 
(15)   end for
(16) end for
(17) return  $a$ 

```

ALGORITHM 2

results and it is time-consuming for users to discover the useful one. The fundamental principle of the PageRank algorithm is that the webpages would be ranked high if it is pointed by high-rank webpages, and top ranking webpages are more likely to be pointed to than lower ranked webpages. Other than ranking the importance of websites, researchers nowadays also use it to measure the importance of diverse entities in a variety of networks, such as ranking the importance of scholars in academic networks. The PageRank values of articles can be obtained by the following formula:

$$\text{PR}(p_i) = \frac{1-d}{N} + d \sum_{j=1}^m \frac{\text{PR}(p_j)}{L(p_j)} \quad (9)$$

where p_i represents the paper, N is the total amount of the articles, p_j is the node that links to p_i , and $L(p_j)$ is p_j 's total outgoing links. $\text{PR}(p_i)$ and $\text{PR}(p_j)$ indicate the importance values of p_i and p_j correspondingly. d is the damping factor which controls the visiting probability of node p_i that can be visited by the link directed to it. A variety of researches have studied the influence of damping factor's different values, and they all believe that it is more suitable for the whole calculation process when set as 0.85. Therefore, in our paper, the values of damping factor are all set as 0.85 as mentioned above. Since the PageRank calculation procedure is iterated, we update each paper's value at every step of the computations based on (9). When the values of all the papers are converged to a steady state, the calculations are stopped, and finally the PageRank value of each article is obtained.

4.2.2. Updated Scores of Papers and Venues in the Paper-Venue Network. Next, the undirected paper-venue network is constructed to calculate the importance of papers and venues considering the mutual influence among them by using the

HITS algorithm. Because the qualities of papers are different originally, we take the PageRank scores of them that are obtained from the last step as their initial value in the step. The major function of HITS algorithm is similar to the PageRank algorithm, which also calculates the importance of entities in the networks. In HITS algorithm, each node possesses two values, which are the authority and hub values. The hub value indicates the value of node's links to other nodes, and the authority represents the quality of node itself. If a node is widely known as a hub, it can guide the users to the nodes with high authority values. On the contrary, if a node's authority value is high, it can be regarded as the node with important content. The authority and hub values of nodes can be calculated as follows:

$$\begin{aligned} \text{auth}(a_k) &= \sum_{i=1}^s \text{hub}(l_i) \\ \text{hub}(a_k) &= \sum_{i=1}^v \text{auth}(p_i) \end{aligned} \quad (10)$$

where a_k is the node, $\text{auth}(a_k)$ is the authority value of it, and we apply it to represent its impact in the network. l_i is the node links to a_k in the network, and s is the sum of l_i . p_i indicates the node that a_k points to, and v is the total number of p_i . At the beginning, if a_k is a venue, its initial authority and hub values are set as 1; otherwise, its initial authority and hub values are set equal to its PageRank score that is obtained from the last step.

4.2.3. Scores of Scholars in the Paper-Author Network. In this part, the paper-author network is established to evaluate scholars' impact. Other than the PageRank algorithm, we also utilize the HITS algorithm to measure the importance of scholars based on the paper-author network. To obtain scholars' authority values, the above-mentioned calculation equations are still applied; however, we set the initial values differently. If the node is a paper, we set its initial values equal to its value obtained from the last step; else the values of the node are set equal to 1. The overall measurement of scholars' impact in heterogeneous academic networks (NI) is calculated as follows:

$$\text{NI}(a_i) = \text{auth}(a_i) \left\{ \sum_{p=1}^n \text{PR}(p_i) \text{auth}(j_k) \right\} \quad (11)$$

where n is a_i 's total amount of scholarly articles, $\text{PR}(p_i)$ is the PageRank value of a_i 's paper in the citation network, $\text{auth}(a_i)$ is author a_i 's authority value in the paper-author network, and $\text{auth}(j_k)$ is the authority value of p_i 's corresponding venue j_k in the paper-venue network.

With the above analysis and the applications of three heterogeneous academic networks, the mutually reinforced procedure of scholarly entities can be explored. In addition, the hybrid of the PageRank and HITS algorithms also can highlight their different advantages in adapting different network topologies and improve the ranking results of scholarly entities in the networks.

4.3. Final Calculation of Scholars' Impact. After finishing the calculation of the above two parts, we then come up with the final formula for evaluating the impact of scholars. In our proposed AIRank method, it consists of two major parts, which are scholars' positions in the network and the hybrid importance values of scholarly entities in the above-mentioned three subnetworks. The theory of structural holes can indicate scholars' abilities to connect different people; therefore we utilize it in our method to depict scholars' positions in the network. To capture the multiplicities of information that researchers acquire through their relationships with other people, we measure these multiplicities from two aspects, which are the diversity of their coauthors and the quantity and quality of academic conferences they attend. In addition, we also consider the mutual effects among different academic entities in the networks together to gauge the scientific impact of scholars. As a consequence, we calculate scholar's final score according to the following formula:

$$F(a_i) = \frac{1 - \xi - \omega}{n} + \xi Z_{\text{SI}_{a_i}} + \omega Z_{\text{NI}_{a_i}} \quad (12)$$

where $F(a_i)$ represents the final impact score of author a_i , $Z_{\text{SI}_{a_i}}$ and $Z_{\text{NI}_{a_i}}$ are the normalization values of SI_{a_i} and NI_{a_i} according to (7), and ξ and ω are parameters.

With the above descriptions, we propose a scholars' impact evaluation method which measures the scientific impact from two aspects. Our method not only considers the impact of scholars in heterogeneous academic networks through the mutual influence mechanism among academic entities, but also integrates the positions of scholars in the networks and their abilities to access various kinds of information and researchers to measure their overall scientific impact.

5. Experimental Results

In this section, we explore the performance of AIRank in the real dataset. Since there is no ground truth for the evaluation of scholars' impact, the citation counts are applied as the ground truth to validate their performance. In academia, it is commonly acknowledged that if one scholar is outstanding, he or she has higher citation counts comparing to other researchers. To explore the effectiveness of the AIRank in selecting high-impact scholars with interdisciplinary nature, we first compare each method's top ranking scholars' average citation counts, common members with citation's ranking lists, and ranking positions of scholars. To specifically show the detailed information of the top researchers selected by each method, we then list the detailed citation counts and cross-domain citations of top 10 scholars in each method to prove the efficiency of our AIRank. In addition, the Pearson Correlation Coefficient between the citation counts and each ranking list is also calculated to show the correlations.

5.1. Dataset and Experimental Setup. The subdataset used for our experiments is acquired from the Microsoft Academic Graph (MAG) datasets. It provides the detailed information of each article. To improve the efficiency of our experiments, the dataset needs to be extracted. In order to alleviate



FIGURE 4: The ACM Computing Classification System.

the effect of different research areas and years of entering academia, we choose scholars that are from the same area and whose academic careers ages are the same for scientific impact evaluation. The academic age in our paper refers to the years between scholar publishing his or her first article and the last article in the database. The final dataset includes 79,321 scholars and 105,123 publications.

When calculating the values of NI, we apply both the PageRank and HITS algorithms to rank the importance of scholars in heterogeneous academic networks. The operation mechanism of these two algorithms is similar in which they both need a sufficient number of iterations to converge. In our case, we set the iteration numbers as 500 times, and the difference value of the sum of all the scholars' values obtained from two successive iterations is smaller than a threshold (set as 0.000001).

5.2. The CCS Classification. To measure the cross-domain citations of articles and their authors, we adopt the ACM Computing Classification System (CCS) from the website

<https://www.acm.org/>. It is a subject classification system for computing, to classify the articles into the related areas according to their keywords. The specific classification criteria are shown in Figure 4. As it shows, there are several major domains and a set of keywords is included in each main kind. According to the keywords listed above, articles can be sorted to the corresponding domains.

5.3. Baseline Methods. In order to investigate the effectiveness of our proposed AIRank method, we employ the different variants of our method, the PageRank algorithm, and h -index for comparison. The details of the above methods are as follows:

- (i) SI^{BrC} : it represents the value of SI which utilizes BrC (see (1)) to measure the positions of scholars in the network.
- (ii) SI^{BeC} : it represents the value of SI which utilizes BeC (see (3)) to measure the positions of scholars in the network.

- (iii) NI: it is part of our proposed AIRank, which only considers the combination results through applying PageRank and HITS algorithm under heterogeneous academic networks to evaluate the impact of scholars.
- (iv) AIRank^{BrC}: it is our proposed method, which utilizes BrC to measure the positions of scholars in the network.
- (v) AIRank^{BeC}: it is our proposed method, which utilizes BeC to measure the positions of scholars in the network.
- (vi) PageRank: it applies the PageRank algorithm to evaluate the impact of each scholar.
- (vii) h -index: it is the h -index value of each scholar.

To start the research work, scholars often need to review the existing literature from related areas. Therefore, it is commonly recognized that scholars may be inspired by articles in areas other than articles within the same area. As a consequence, the citations of articles may be not only from a single area, but also from other disciplines due to their impact on other areas. To understand the interdisciplinary nature of citations, we first investigate the citation distributions of different domains in MAG dataset.

As shown in Figure 5, it is a chord graph, which indicates the proportions of articles from each domain in the MAG dataset. Different domains are represented with different colors, and the citation distributions of articles in each domain can be easily observed. The diagram displays that the total numbers of papers in applied computing and computing methodologies areas are larger than the numbers of articles in other domains. Furthermore, papers in these two areas also shed light on the scientific inventions of other areas due to their citation distributions. Generally, it is obvious that almost every article cites papers from other areas. The areas in computer science correlate with each other closely and promote the development of computer science together.

With the above analysis, the tendency of citation distributions is apparently showing an increasing trend of interdisciplinary collaborations, i.e., the number of cross-domain citations. To further explore the effect of cross-domain citations on the scientific impact of scholars, we then list the cross-domain citations of top ranking scholars by citation counts and h -index. As shown in Figure 6(a), the top percentile ranking scholars with bigger citation counts also obtain higher cross-domain citations. The same phenomenon is also observed in Figure 6(b), where the higher the h -index values of scholars, the more the average cross-domain citation counts that they will get. The trends of these two figures are alike; however, their concrete average cross-domain citation counts of top ranking scholars appear to be different. There exists a great numerical difference of top 10% scholars' average cross-domain citation counts between Figures 6(a) and 6(b), and the numerical differences decrease with the increase of top percentile ranking scholars. The reason behind this phenomenon correlates closely with the principle of calculating scholar's h -index. Although there exist some numerical differences, the overall trend of these figures is

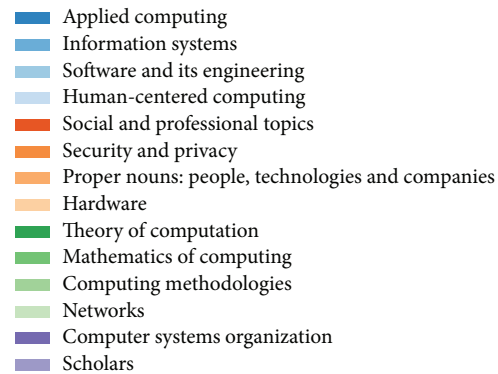
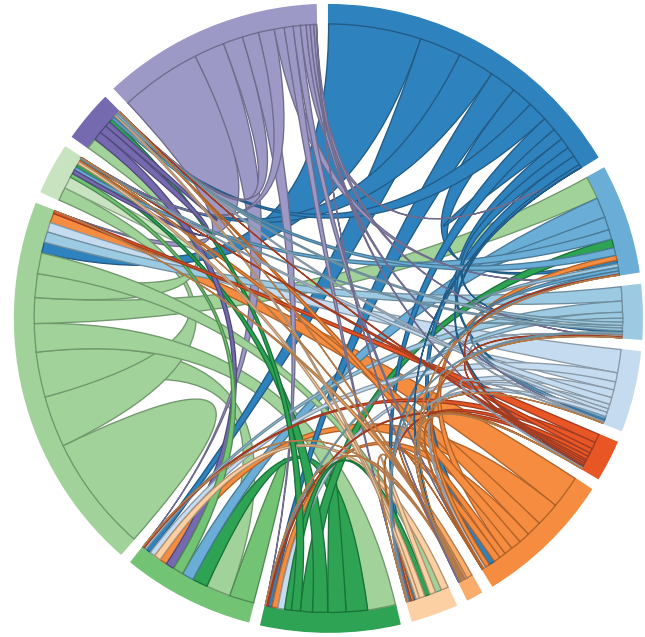


FIGURE 5: The interdisciplinary citations among articles in computer science area.

similar, which validates the fact that high-impact scholars also gain high reputations in other domains.

In order to investigate each method's ability to identify influential scholars more exquisitely and in convincible manner, we first compare the number of common members between each methods ranking list and citation rankings. A ranking list of scholars can be obtained through their final scores by each method. As shown in Figure 7(a), the SI shows a better result than the performance of NI. Meanwhile, the overall performances of AIRank variants are better than other methods. Our proposed AIRank^{BrC} method can get the most common members with the citation counts rankings when comparing the top 5%, top 10%, and top 20% ranking lists by each method. Furthermore, we then compare each method's average citation counts of top ranking scholars. As shown in Figure 7(b), the number of the average citation counts of top scholars according to our AIRank method is the highest among other methods, while the AIRank^{BrC} method still achieves the best performance comparing with other methods. Through Figures 6(a) and 6(b), we find that

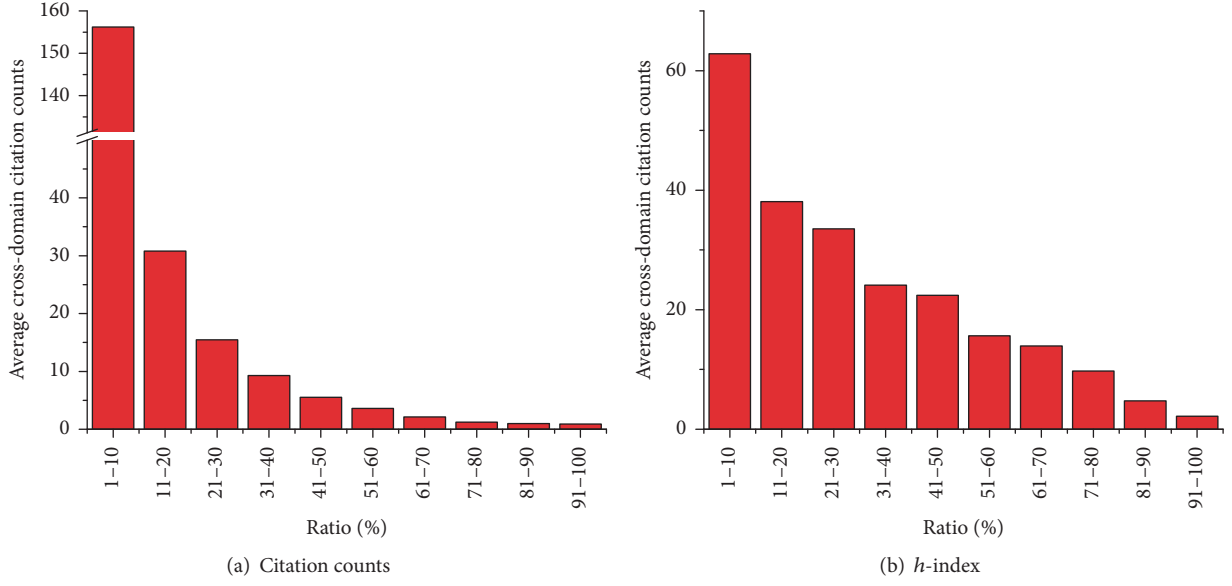
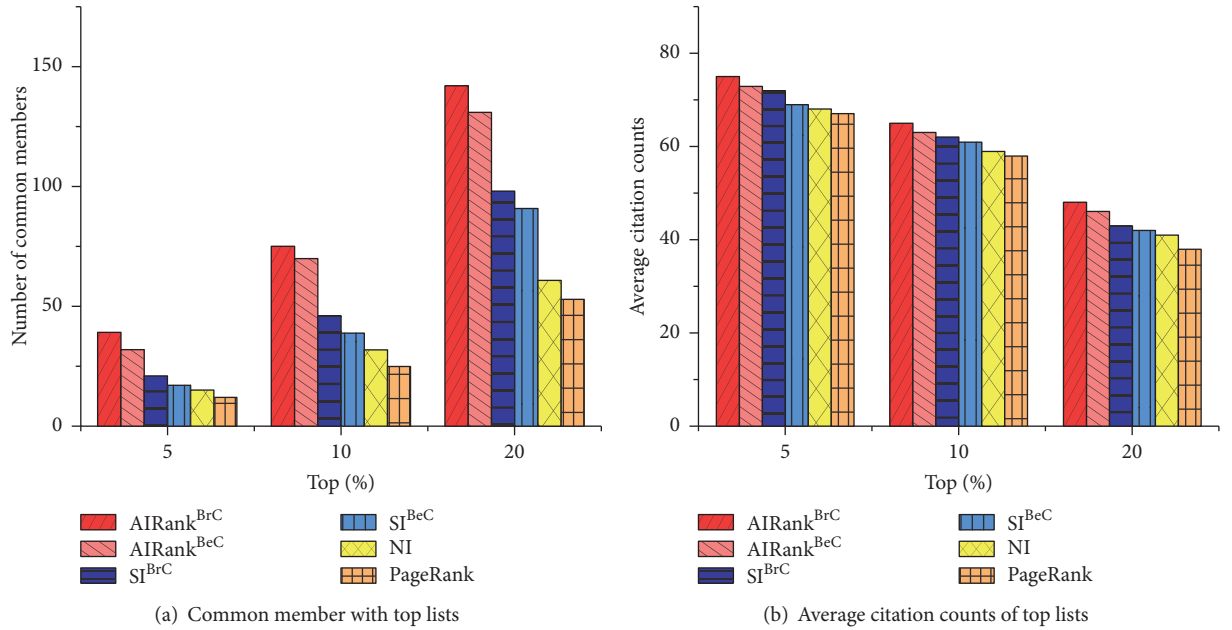


FIGURE 6: The average cross-domain citation counts of top ranking scholars.

FIGURE 7: The performance of top ranking scholars by SI^{BrC} , SI^{BeC} , NI , $AIRank^{BrC}$, and $AIRank^{BeC}$.

the more influential the scholars, the more the cross-domain citation counts that they will obtain. We then specifically show each method's top 10 researchers' citation counts and cross-domain citation counts. As shown in Table 1, it is clear that performance of our method is better than the PageRank method. Due to the mechanism of PageRank algorithm, the higher value of PageRank score indicates the more citations from influential scholars; therefore, the top 3 scholars' citation counts according to PageRank algorithm are high while the rest decrease distinctly. As shown in Tables 1 and 2 and Figures 7(a) and 7(b), the results demonstrate that the performance of our method is better than other

approaches when comparing top ranking scholars' overall average citation counts and cross-domain citations. These results also confirm the findings displayed in the above tables. Generally, the AIRank method has a better performance when applying the bridge counts to measure the positions of scholars in the network.

The ranking positions of the top 100 scholars according to the citation counts in our proposed methods are also investigated. Since the specific calculation process of each method is different, scholars' ranking positions by each method are distinct either. In this paper, the number of citation counts is chosen as the ground truth; hence we assume

TABLE 1: Top 10 scholars of each method.

Top 10	AIRank ^{BrC}		AIRank ^{BeC}			PageRank		
	Citations	Cross-domain citations	Top 10	Citations	Cross-domain citations	Top 10	Citations	Cross-domain citations
7F2CEC81	1127	1011	8023C793	846	818	80EB57FC	613	596
7FB76008	857	793	7E2B1F64	783	726	7F2CEC81	1127	1011
7E2B1F64	783	726	7FB76008	857	793	8023C793	846	818
8I73CEDE	68	34	80EB57FC	613	514	7D6A4BFF	187	179
7D6A4BFF	187	179	7F2CEC81	37	23	80AD9709	98	87
8023C793	846	818	7D6A4BFF	98	98	7F680B0B	98	98
80EB57FC	613	514	7DE7A740	485	409	756F9F32	23	14
7DE7A740	485	409	0838B97F	87	80	4899EC1B	79	53
80D1979B	134	126	78322C72	126	113	78322C72	97	80
7BB5A93A	137	122	7F78CE41	112	102	7FC94B6B	89	81

TABLE 2: Top 10 scholars of each method.

Top 10	SI ^{BrC}		SI ^{BeC}			NI		
	Citations	Cross-domain citations	Top 10	Citations	Cross-domain citations	Top 10	Citations	Cross-domain citations
7DE7A740	485	409	7DE7A740	485	409	7DE7A740	485	409
80EB57FC	613	514	80EB57FC	613	514	80EB57FC	613	514
802E02C5	168	161	802E02C5	69	61	802E02C5	69	61
0857BCE0	286	286	0857BCE0	286	286	0857BCE0	286	286
7ED3570E	228	213	7ED3570E	228	213	7ED3570E	228	213
7FF53EE6	89	79	7FF53EE6	89	79	7FF53EE6	89	79
80FE41D4	112	112	80FE41D4	112	112	80FE41D4	112	112
8I73CEDE	68	34	8043DB84	45	30	7EFAE119	24	18
7F2CEC81	37	23	7F2CEC81	34	27	7F2CEC81	34	27
113BBABC	63	45	75ADB28C	28	21	4769E8AE	16	11

that the more effective in identifying influential scholars of the above-mentioned method it is, the higher the ranking positions of the top 100 scholars by citation counts are. For instance, one scholar ranks the first by citations counts while in other methods he or she, respectively, ranks the 4th, 10th, and 3rd; then it is obvious that the method which ranks this scholar the 3rd achieves the best performance among others. The top 100 scholars' ranking positions by each method are shown in Figure 8, and the ranking differentials can be directly obtained. It is apparent that the AIRank method achieves the best performance, whose range of the ranking positions for top scholars is the smallest. Among these methods, it is obvious that the AIRank^{BrC} still performs the best in scholars' ranking positions.

Other than the efficiency in identifying high-impact scholars, the performance of evaluating the overall scientific impact of scholars still needs to be explored. We first examine the performance from the angle of distinguishing scholars with different scientific impact. According to scholars' citation counts, the higher ranked scholars are considered as positive entities, and authors that ranked

low are deemed as negative entities. The above-mentioned methods are used as classifiers to evaluate their ranking results. In general, the classification results can have four types: top ranking scholar is classified as higher ranked (true positive); the scholar is higher ranked but is considered as top ranking scholar (false positive); lower ranked scholar is classified as lower ranked (true negative); lower ranked scholar but classified as top ranking scholar (false negative). With these four kinds of classification results, the four rates can be calculated. The true positive rate (TPR) can be calculated as $\sum \text{truepositive} / \sum \text{conditionpositive}$, the false positive rate (FPR) can be calculated as $\sum \text{falsepositive} / \sum \text{conditionnegative}$, the true negative rate (TNR) can be calculated as $\sum \text{truenegative} / \sum \text{conditionnegative}$, and the false negative rate (FNR) equals $\sum \text{falsenegative} / \sum \text{conditionpositive}$.

The Receiver Operating Characteristic (ROC) curves of each method can be obtained through the above-mentioned rates. As shown in Figure 9, the ordinate is the Sensitivity = $\text{TPR} / (\text{TPR} + \text{FNR})$, and the abscissa is the $1 - \text{Specificity} = \text{TNR} / (\text{TNR} + \text{FPR})$. The ROC curves in Figure 9 indicate

TABLE 3: AUC of each method.

	SI^{BrC}	SI^{BeC}	NI	$AIRank^{BrC}$	$AIRank^{BeC}$	PageRank
AUC	0.64504	0.66962	0.61812	0.73476	0.80749	0.59133

TABLE 4: Comparison of Pearson Correlation Coefficient.

	SI^{BrC}	SI^{BeC}	NI	$AIRank^{BrC}$	$AIRank^{BeC}$
Citation counts	0.453	0.437	0.496	0.538	0.522
h -index	0.230	0.220	0.098	0.231	0.222
PageRank	0.738	0.782	0.305	0.785	0.742

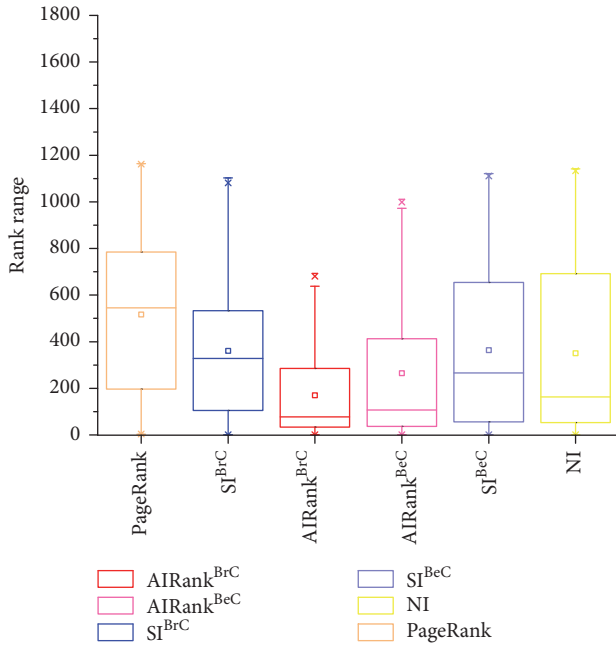
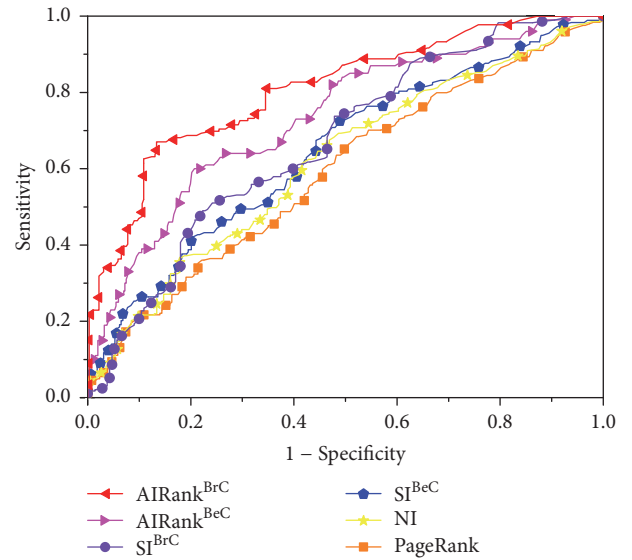


FIGURE 8: Boxplots of ranking positions for top scholars.

that our AIRank method can classify different scholars with the best performance. Moreover, we calculate the area that the ROC curves cover (AUC) which indicates the classifying accuracy rate. It is clear that our AIRank method has the highest accuracy rate according to Table 3. Through the above results, we can observe that the AIRank method performs better than other methods in classifying the scholars.

We adopt the universally acknowledged citation counts and h -index values to evaluate the performance of each method. The Pearson Correlation Coefficient is commonly used to measure the correlation between two sets of data. The value of it ranges from -1 to 1 , which represents the fact that the correlations of two sets of data are from the most negative to the most positive ones. We apply the Pearson Correlation Coefficient to calculate the correlation among all the baseline methods (SI^{BrC} , SI^{BeC} , NI, $AIRank^{BrC}$, and $AIRank^{BeC}$) with the citation counts, h -index, and the PageRank algorithm. As shown in Table 4, the results indicate that the AIRank method outperforms other methods with higher values, and it makes a great improvement comparing to applying the SI and NI

FIGURE 9: ROC curves of SI^{BrC} , SI^{BeC} , NI, $AIRank^{BrC}$, and $AIRank^{BeC}$.

measurements alone. Meanwhile, the $AIRank^{BrC}$ method still achieves the best performance compared to other methods.

Generally, we examine the performance of each method from two main aspects: the ability to identify influential scholars and the comprehensiveness of evaluating the overall impact of scholars. We compare the cross-domain citations, ranking positions, common members, and average citations of the top ranking scholars in each method to investigate the capacity of identifying influential scholars. The results indicate that our AIRank method, specifically the $AIRank^{BrC}$ method, shows the best performance among all the other methods in identifying influential scholars. In addition, the ROC curve, the value of AUC, and the Pearson Correlation Coefficient are utilized to measure each method's efficacy in evaluating the overall impact of scholars. Similarly, the $AIRank^{BrC}$ method still prevails over all the other methods.

6. Conclusion

In this paper, our primary concern is to quantify scholars' scientific impact by utilizing the heterogeneous academic network topology. The positions of scholars in the coauthor network are taken into consideration to measure the scientific

impact of scholars and their effects as well. We depict it from three aspects, which are the diversity of coauthors, the qualities of conference papers that scholars published, and their measurements of structural holes. Besides, we also integrate the interplay between different scholarly entities in heterogeneous academic networks through the random walk algorithms. Based on these indicators and scholars' impact in heterogeneous academic networks, we propose the AIRank method.

We construct the experiments on MAG dataset to prove the efficiency of AIRank and select the appropriate measurements on the positions of scholars in the network. Through the experiments on the real dataset, we find that influential scholars in some specific areas also obtain high reputation in other domains. The results also demonstrate that our algorithm performs better than other methods in selecting top ranking scholars with more cross-domain citation counts and measuring scholars' scientific impact more comprehensively. Furthermore, there still exists room for further modifications; e.g., the effects of the interplay and relationships between scholars on their scientific impact should be mined deeper. Our method is conducted only on literature from computer science area; the results obtained from more datasets on other disciplines could be examined, so that exploring other scientific disciplines for the same observed phenomena could further prove the effectiveness of our work.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this article.

Acknowledgments

The authors extend their appreciation to the Deanship of Scientific Research at King Saud University for funding this work through Research Group no. RG-1438-027.

References

- [1] F. Xia, W. Wang, T. M. Bekele, and H. Liu, "Big Scholarly Data: A Survey," *IEEE Transactions on Big Data*, vol. 3, no. 1, pp. 18–35, 2017.
- [2] Z. Ning, X. Wang, X. Kong, and W. Hou, "A Social-aware Group Formation Framework for Information Diffusion in Narrow-band Internet of Things," *IEEE Internet of Things Journal*, vol. PP, no. 99, pp. 1–1, 2017.
- [3] Y. Dong, R. A. Johnson, and N. V. Chawla, "Can Scientific Impact Be Predicted?" *IEEE Transactions on Big Data*, vol. 2, no. 1, pp. 18–30, 2016.
- [4] J. E. Hirsch, "An index to quantify an individual's scientific research output," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 102, no. 46, pp. 16569–16572, 2005.
- [5] L. Egghe, "Theory and practise of the g-index," *Scientometrics*, vol. 69, no. 1, pp. 131–152, 2006.
- [6] E. Garfield, "The history and meaning of the journal impact factor," *Journal of the American Medical Association*, vol. 295, no. 1, pp. 90–93, 2006.
- [7] C. Shi, Y. Li, J. Zhang, Y. Sun, and P. S. Yu, "A survey of heterogeneous information network analysis," *IEEE Transactions on Knowledge and Data Engineering*, vol. 29, no. 1, pp. 17–37, 2017.
- [8] L. Page, S. Brin, R. Motwani, and T. Winograd, *The pagerank citation ranking: bringing order to the web*, 1999.
- [9] J. M. Kleinberg, "Authoritative sources in a hyperlinked environment," *Journal of the ACM*, vol. 46, no. 5, pp. 604–632, 1999.
- [10] M. Nykl, K. Ježek, D. Fiala, and M. Dostal, "PageRank variants in the evaluation of citation networks," *Journal of Informetrics*, vol. 8, no. 3, pp. 683–692, 2014.
- [11] Z. Ning, X. Hu, Z. Chen et al., "A cooperative quality-aware service access system for social internet of vehicles," *IEEE Internet of Things Journal*, vol. PP, no. 99, pp. 1–1, 2017.
- [12] R. S. Burt, *Structural hole*, Harvard Business School Press, Cambridge, MA, USA, 1992.
- [13] T. Lou and J. Tang, "Mining structural hole spanners through information diffusion in social networks," in *Proceedings of the 22nd international conference on World Wide Web*, pp. 825–836, Rio de Janeiro, Brazil, May 2013.
- [14] X. Su, W. Wang, S. Yu, C. Zhang, T. M. Bekele, and F. Xia, "Can Academic Conferences Promote Research Collaboration?" in *Proceedings of the 16th ACM/IEEE-CS*, pp. 231–232, Newark, New Jersey, USA, June 2016.
- [15] L. Li and H. Tong, "The Child is Father of the Man," in *Proceedings of the 21th ACM SIGKDD International Conference*, pp. 655–664, Sydney, NSW, Australia, August 2015.
- [16] D. Wang, C. Song, and A.-L. Barabási, "Quantifying long-term scientific impact," *Science*, vol. 342, no. 6154, pp. 127–132, 2013.
- [17] F. Xia, X. Su, W. Wang, C. Zhang, Z. Ning, and I. Lee, "Bibliographic analysis of Nature based on Twitter and Facebook altmetrics data," *PLoS ONE*, vol. 11, no. 12, Article ID e0165997, 2016.
- [18] R. Sinatra, D. Wang, P. Deville, C. Song, and A.-L. Barabási, "Quantifying the evolution of individual scientific impact," *Science*, vol. 354, no. 6312, Article ID aaf5239, 2016.
- [19] A. Clauset, D. B. Larremore, and R. Sinatra, "Data-driven predictions in the science of science," *Science*, vol. 355, no. 6324, pp. 477–480, 2017.
- [20] R. K. Pan and S. Fortunato, "Author impact factor: Tracking the dynamics of individual scientific impact," *Scientific Reports*, vol. 4, article no. 4880, 2014.
- [21] S. Xiao, J. Yan, C. Li et al., "On modeling and predicting individual paper citation count over time," in *Proceedings of the 25th International Joint Conference on Artificial Intelligence, IJCAI 2016*, pp. 2676–2682, usa, July 2016.
- [22] X. Wan and F. Liu, "Are all literature citations equally important? Automatic citation strength estimation and its applications," *Journal of the Association for Information Science and Technology*, vol. 65, no. 9, pp. 1929–1938, 2014.
- [23] M. Valenzuela, V. Ha, and O. Etzioni, "Identifying meaningful citations," in *Proceedings of the in Workshops at the Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.
- [24] X. Bai, F. Xia, I. Lee, J. Zhang, and Z. Ning, "Identifying anomalous citations for objective evaluation of scholarly article impact," *PLoS ONE*, vol. 11, no. 9, Article ID e0162364, 2016.
- [25] R. Liang and X. Jiang, "Scientific ranking over heterogeneous academic hypernetwork," in *Proceedings of the in Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, pp. 20–26, 2016.
- [26] S. Wang, S. Xie, X. Zhang, Z. Li, P. S. Yu, and Y. He, "Coranking the future influence of multiobjects in bibliographic network

- through mutual reinforcement,” *ACM Transactions on Intelligent Systems and Technology*, vol. 7, no. 4, article no. 64, 2016.
- [27] Y. Li, C. Wu, X. Wang, and P. Luo, “A network-based and multi-parameter model for finding influential authors,” *Journal of Informetrics*, vol. 8, no. 3, pp. 791–799, 2014.
- [28] J. D. West, M. C. Jensen, R. J. Dandrea, G. J. Gordon, and C. T. Bergstrom, “Author-level eigenfactor metrics: Evaluating the influence of authors, institutions, and countries within the social science research network community,” *Journal of the Association for Information Science and Technology*, vol. 64, no. 4, pp. 787–801, 2013.
- [29] X. Cao, Y. Chen, and K. J. Ray Liu, “A data analytic approach to quantifying scientific impact,” *Journal of Informetrics*, vol. 10, no. 2, pp. 471–484, 2016.
- [30] J. Zhang, F. Xia, W. Wang et al., “Cocorank: A collaboration caliber-based method for finding academic rising stars,” in *Proceedings of the International Conference Companion on World Wide Web*, pp. 395–400, Montreal, Quebec, Canada, April 2016.
- [31] D. Yu, W. Wang, S. Zhang, W. Zhang, and R. Liu, “A multiple-link, mutually reinforced journal-ranking model to measure the prestige of journals,” *Scientometrics*, vol. 111, no. 1, pp. 521–542, 2017.
- [32] D. Fiala, L. Šubelj, S. Žitnik, and M. Bajec, “Do PageRank-based author rankings outperform simple citation counts?” *Journal of Informetrics*, vol. 9, no. 2, pp. 334–348, 2015.
- [33] T. Amjad, Y. Ding, A. Daud, J. Xu, and V. Malic, “Topic-based heterogeneous rank,” *Scientometrics*, vol. 104, no. 1, pp. 313–334, 2015.
- [34] Y. Wang, Y. Tong, and M. Zeng, “Ranking scientific articles by exploiting citations, authors, journals, and time information,” in *Proceedings of the in Twenty-Seventh AAAI Conference on Artificial Intelligence*, pp. 933–939, 2013.
- [35] T. Amjad, Y. Ding, J. Xu et al., “Standing on the shoulders of giants,” *Journal of Informetrics*, vol. 11, no. 1, pp. 307–323, 2017.
- [36] W. Wang, S. Yu, T. M. Bekele, X. Kong, and F. Xia, “Scientific collaboration patterns vary with scholars’ academic ages,” *Scientometrics*, vol. 112, no. 1, pp. 329–343, 2017.
- [37] J. Zhang, Z. Ning, X. Bai, W. Wang, S. Yu, and F. Xia, “Who are the Rising Stars in Academia?” in *Proceedings of the the 16th ACM/IEEE-CS*, pp. 211–212, Newark, New Jersey, USA, June 2016.

Research Article

Research of Deceptive Review Detection Based on Target Product Identification and Metapath Feature Weight Calculation

Ling Yuan , Dan Li , Shikang Wei , and Mingli Wang 

School of Computer Science, Huazhong University of Science and Technology, Wuhan 430074, China

Correspondence should be addressed to Dan Li; lidanhust@hust.edu.cn

Received 28 December 2017; Revised 20 March 2018; Accepted 10 April 2018; Published 11 June 2018

Academic Editor: Xiuzhen Zhang

Copyright © 2018 Ling Yuan et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

It is widespread that the consumers browse relevant reviews for reference before purchasing the products when online shopping. Some stores or users may write deceptive reviews to mislead consumers into making risky purchase decisions. Existing methods of deceptive review detection did not consider the valid product review sets and classification probability of feature weights. In this research, we propose a deceptive review detection algorithm based on the target product identification and the calculation of the Metapath feature weight, noted as *TM-DRD*. The review dataset of target product is modeled as a heterogeneous review information network with the feature nodes. The classification method of graph is used to detect the deceptive reviews, which can improve the efficiency and accuracy of deceptive review detection due to the sparsity, imbalance of deceptive reviews, and the absence of category probability of feature weight calculation. The *TM-DRD* algorithm we proposed is validated on the real review dataset *Yelp* and compared with the *SpEagle*, *NFC*, and *NetSpam* algorithm. The experiment results demonstrate that the *TM-DRD* algorithm performs better than the other method with regard to the accuracy and efficiency.

1. Introduction

With the rapid development of E-commerce, traditional concepts and methods of consumption are rapidly changing. People are increasingly inclined to consume online because it is simpler, faster, and more convenient. Many shopping sites or platforms offer their own online review platforms, such as *Yelp* and *Amazon*, allowing consumers to comment on products.

Product reviews are widely used in individuals and organizations. A survey by Cone, Inc. (<http://www.conecomm.com/contentmgr/showdetails.php/id/4008>), states that 67% of consumers will read the relevant comments before purchase, where 82% of these consumers conclude that product reviews will affect their final purchase decisions and about 80% of them will change their purchase intentions after reading negative reviews. Evaluation of the products or services quality will directly affect the buying behavior. If a product has a lot of praise, the user will show a greater tendency to purchase. Deceptive detection and prevention are complicated

by lack of standard online deception detection, a computationally efficient method for detecting deception in large online communities, and social media developers looking to prevent deception [1]. The deceptive reviews are fake reviews deliberately posted by a few illegal users. The reviews websites or platforms become the target of these deceptive users. Deceptive reviews control the viewpoint of target products and mislead consumers.

In recent years, there have been a large number of effective methods for detecting deceptive reviews [2], but there are still some problems to be solved in this field.

(1) *Method Based on the Review Texts*. The feature extraction of such methods has serious reliance on the field of review data. The scalability of the model is poor. Moreover, for different fields of the review data, the dataset needs to be regained and marked, while the deceptive review dataset is difficult to obtain. It has also become a major issue for deceptive review detection based on the review texts.

(2) *Method Based on Abnormal Behavior.* The main drawback of this kind of method is that most reviewers do not have the relevant information to conduct behavioral analysis, which results in limited ability to identify abnormal behavior. What is more, the professional deceptive users are good at hiding their abnormal behavior, making their behavior similar to the normal users.

In order to improve the efficiency and accuracy of deceptive review detection, this paper proposes a deceptive review detection algorithm based on the target product identification and the calculation of the metapath feature weight, noted as *TM-DRD*, involving two research contents.

(1) In order to identify the target product of deceptive review, we propose a method based on abnormal score, noted as *AS-TPI*. Firstly, we analyze the different states of deceptive reviews and then calculate the difference between the actual product rating scale and the standard score ratio. Finally, the distribution of the score in time is estimated by using the kernel density.

(2) We define the features separately from the reviews and reviewers, combine the target products and related review datasets identified by *AS-TPI*, and then construct the heterogeneous review information networks. We propose a method to calculate feature weights based on the metapath to calculate the deceptive degree probability of reviews to determine the final category of reviews, noted as *MFW-DDP*.

The related work is described in the Section 2. The preliminaries for the proposed *TM-DRD* algorithm are illustrated in Section 3. The proposed methodology is presented in Section 4. The experiments about the proposed algorithm are illustrated in Section 5. Section 6 concludes the whole paper.

2. Related Work

There are two directions of the current research on the deceptive review detection [3–5]: one is based on the reviews, and the other is based on the reviewers. For these two directions, there are the following research methods.

(1) *Method Based on the Content of Reviews.* The method detects the deceptive reviews based on the similarities and linguistic features of the reviews. It extracts relevant features from features of vocabulary, consistency of content, consistency of review style, and semantic consistency to identify deceptive reviews. By analyzing the tendencies of sentiment, semantics, we can find the deceptive reviews deviating from the normal reviews.

Ott et al. [6] used crowdsourcing platform (AMT) to construct datasets and used comprehension method of natural language to acquire linguistic features from multiple perspectives. They trained many kinds of classifiers and compared their performance. But the test results were not very well on real business datasets. Li et al. [7] created deceptive reviews datasets manually and used naive Bayesian machine learning algorithm for deceptive reviews detection. A two-sided cotraining semisupervised learning method was proposed to mark a large number of unlabelled reviews. And they used it as follow-up deceptive reviews test datasets. Rout et al. [8] also used semisupervised learning approaches to improve the

F-score metric in classification, and they incorporated new dimensions in the feature vector to obtain better results. Feng et al. [9, 10] proved that deep syntactic information of texts is very effective in deceptive reviews detection. They used probabilistic context-free syntax PCFG. The deep syntactic features of the reviews texts are generated by the generative rules of the PCFG syntax analysis tree and the SVM classifier is trained to identify the deceptive reviews. Li et al. [11] proposed a method of deceptive detection based on the LDA model named as TopicSpam, which can classify the reviews by detecting the probability of the deceptive index by detecting the slight difference between the distribution of the keywords of the real reviews and the deceptive reviews.

Due to the concealment, the behaviors of reviewers who publish deceptive reviews are getting closer and similar to those of normal users, and deceptive strategies they use are also getting better and more diversified.

(2) *Method Based on Behavior.* In this method, most of the features are extracted based on the metadata of the reviews (time of reviews, frequency of reviews, information of the first reviewers of the product, etc.), such as the research of [12–14]. They analyze the temporal or spatial information of reviews. If conditions permit, they can also use some privacy data of the site such as IP address, MAC address, and location reviews published, which are very useful to extract behavioral features. Then they mathematicize the features, construct user behavior models, and classify reviews by models.

Lim et al. [15] focused on the behavior of reviewers to find the deceptive reviews. They considered that it was better to study reviewers than reviews because the information obtained from the reviewers' behavior was far more than the information obtained from the reviews themselves. So they proposed a method to detect the deceptive reviewers based on the score of reviewers. They constructed a model from the multifaceted behaviors of reviewers, and designed a deceptive degree scoring function to calculate whether the reviewers are deceptive. Xie et al. [16] proposed a multi-time scale detection method and found time windows that concentratedly distributed deceptive reviews through time series analysis. They considered that the singleton review in such time windows is highly likely to be deceptive, where singleton review means that the reviewer of the review posted only this one review. Their method that makes use of features such as the release date of the review and the historical record of the reviewer is an unsupervised learning method. Mukherjee et al. [17] proposed an unsupervised model of hypothetical reviewers named ASM. They considered the distribution of different behaviors of deceptive reviewers and normal reviewers and set falsehood as an implicit variable and reviewers' behavior as an observation variable. They used a clustering algorithm to identify fake reviewers to identify deceptive reviews. Zhang and Lu [18] investigated the top Weibo accounts whose follower lists duplicate or nearly duplicate each other (hereafter called near-duplicates) and proposed a novel fake account detection method that is based on the very purpose of the existence of these accounts: they are created to follow their targets en masse, resulting in high-overlapping between the follower lists of their customers. The implementation is based on the estimation of Jaccard similarity

using random sampling. Unlike traditional fast algorithms for Jaccard similarity, they estimated the Jaccard similarities without the access to the entire data.

Compared with the method based on the content of the reviews, the behavior-based approach analyzes the characteristics of cheating behaviors from different perspectives and does not require a lot of textual analysis such as viewpoint mining and sentiment analysis. At present, deceptive reviews detection methods based on user behavior are analyzed from several common cheating behaviors. With the constant change of behavior of deceptive reviewers, new cheating behaviors need to be further extracted and analyzed to improve detection accuracy.

(3) *Method Based on the Relationship*. The method builds a relational model by studying the complex relationships among reviewers, critics, products, or stores. It uses the associations or some graph-based methods in the diagram to sort the reviews or mark the categories, with establishing a network diagram of relationships among the three.

Wang et al. [19] considered that it was not enough to only use behavior-based heuristic rules. Therefore, for the first time, a graph-based approach is proposed to detect the deceptive reviewers. This method can detect cheating behaviors that some original detection methods cannot detect. Li et al. [20] used a vector representation of products and reviewers related to reviews through the tensor decomposition method and combined it with the feature of bag bigram and then used SVM to detect the deceptive review. In their method, all reviewers and products related to reviews are characterized by a matrix, and then the tensor decomposition technique is used to translate each user and product into a corresponding vector representation. The advantage of this method is the vectorization of the global features, effectively improving the detection performance. There have been a large number of the deceptive reviewers who often work collaboratively to promote or demote target products, which severely harm the review system [21, 22]. Xu et al. [21] proposed a KNN-based approach based on the similarity of reviewers and the relevance of reviewer groups. They proposed a graph model of collusion reviewer based on Pairwise Markov Network, which was used to infer the classification of critics. Fei et al. [23] found that the reviewers and reviews appearing in sudden periods often showed the trend that the deceptive reviewers cooperate with each other and real reviewers are usually presented together. They established Markov random MRF network model for critics who appeared in different periods of emergency and proposed an evaluation method to evaluate the inference results. Their method has higher accuracy and recall rate for burst reviews detection. In the case of deceptive reviewers groups, Wang et al. [22] introduced a top-down computing framework to detect the deceptive reviewers groups by exploiting the topological structure of the underlying reviewer graph which reveals the coreview collusiveness. A novel instantiation is designed by modeling deceptive reviewers groups as biconnected graphs. Ye and Akoglu [24] proposed a two-stage approach to identify the deceptive reviewer groups and target products of deceptive reviews that they attack. They used GroupStrainer and a hash-clustering

algorithm based on similarity in the graph model to detect the deceptive reviewer groups. For big reviews dataset, Dhingra and Yadav [25] proposed a novel fuzzy modeling based solution to the problem and defined novel FSL deduction algorithm generating 81 fuzzy rules and Fuzzy Ranking Evaluation Algorithm (FREA) to determine the extent to which a group is suspicious and used Hadoop for storage and analyzation.

3. Preliminaries

3.1. Product Rating Difference Calculation. The original review dataset is statistically processed in the product scoring stage to obtain each product and its corresponding scoring dataset. Then it is used as input to a target product recognition algorithm based on the differences in the grade scoring.

In order to describe the target product identification algorithm based on the difference of the grade scores, we present two assumptions and the definitions of related concepts used in the algorithm.

Definition 1 (score distribution, D_p). Each product p corresponds to a score distribution $D_p = \{n_i, 1 \leq i \leq 5\}$, where n_i indicates the number of reviews with score i , as shown in

$$D_p = \{n_1, n_2, n_3, n_4, n_5\}. \quad (1)$$

For example, there are 10 reviews of product p with 1 point, 20 reviews with 2 points, 30 reviews with 3 points, 40 reviews of with 4 points, and 50 reviews with 5 points. The score distribution of product p is $\{10, 20, 30, 40, 50\}$.

Definition 2 (rating scale, $R_{p,i}$). Given a product p and a rating level i , $i \in [1, 5]$, we gather the reviewers set $R_{p,i}$ of the product p rating for i . For $\forall r \in R_{p,i}$, the proportion $s_{p,i}$ of the reviews with rating i is defined as the product rating scale, as shown in (2). The value range is $[0, 1]$.

$$s_{p,i} = \frac{|\{v_{r,p} \mid e_v = i\}|}{|\{v_{r,p}\}|}, \quad (2)$$

where $v_{r,p}$ is the review of reviewer r on product p and e_v is the score associated with review v .

The ratio range $[0, 1]$ is divided into 10 equidistant intervals, and the proportion corresponding to each equidistant interval in turn is $\varphi_1 = 10\%$, $\varphi_2 = 20\%$, \dots , $\varphi_{10} = 100\%$. The distribution of the score i of the product p in proportion is shown in

$$D_{p,i} = \{m_{i,j}, 1 \leq i \leq 5, 1 \leq j \leq 10\}, \quad (3)$$

where $m_{i,j}$ is the number of ratings. The proportion of i -level reviews falls within the range of $[\varphi_{j-1}, \varphi_j]$.

Definition 3 (standard rating scale, s_i). For all the products with a rating of i , $i \in [1, 5]$, we calculate the proportion s_i of reviews for all reviews with a rating of i . s_i is defined as the

standard rating scale. The range and division criteria for s_i are similar as above. Standard rating scale is defined as shown in

$$s_i = \frac{|\{v \mid e_v = i\}|}{|\{v\}|}, \quad (4)$$

where v is any review and e_v is the rating of the v .

We can calculate the proportional distribution of the number of scores for all products rated as i , defining it as the Standard Rating Scale distribution, as shown in

$$SD_i = \{sm_{i,j}, 1 \leq i \leq 5, 1 \leq j \leq 10\}. \quad (5)$$

Definition 4 (rating scale difference, $DIF_{p,i}$). The rating scale difference is the difference between the product rating scale and the standard rating scale. The rating scale difference in grade i on product p is defined as shown in (8).

$$S_i = \sum_{j=1}^{10} m_{i,j}, \quad (6)$$

$$SS_i = \sum_{j=1}^{10} sm_{i,j}, \quad (7)$$

$$DIF_{p,i} = \sum_{j=1}^{10} \left| \frac{m_{i,j}}{S_i} - \frac{sm_{i,j}}{SS_i} \right|. \quad (8)$$

Assumption 5. The criteria for a normal reviewer are fixed; that is, the same rating scale indicates the same tendencies to reviews on all products in its review, so the distribution of normal product ratings amount (the number of reviews or the number of reviewers) on each level should be consistent with a certain law.

Assumption 6. According to the majority voting principle, it is assumed that if there are three or more $DIF_{p,i}$ fallings within the range of nonconfidence intervals, the product is the target product.

3.2. Target Product Identification. The products involved in the real review data set are mainly divided into the following three groups:

- (1) Type one: such products are usually not popular products with a very small number of reviews. Their sales and commentary information are relatively small, such as products in some small self-employed stores. The impact of reviews for such products is small.
- (2) Type two: such products are usually popular products with a very large number of reviews but a very small number of deceptive reviews. These products generally come from shops with high reputation and high recognition, such as Tmall's official flagship store. The most reviews of these products are real reviews and therefore it is not enough to mislead consumers about the purchase decision.

- (3) Type three: such products are defined as target product. They are usually popular products with a very large number of reviews and a very large number of deceptive reviews. It is not easy to tell whether the review is deceptive or not. It is easy to mislead consumers to make objective and correct judgments about the products and make risky purchase decisions. What is more serious is disruption of the equitable and orderly nature of the E-commerce market. Therefore, it is of significance to conduct in-depth analysis and research on this type of products and related reviews. Target products identification with research significance from the mass data can reduce the scope of the review data involved, and the detection efficiency and accuracy can all be improved.

After identifying the target product in the original product scoring dataset, the remaining unidentified product and its scoring dataset are used as the input of the target product identification algorithm based on the kernel density estimation in this section to identify the target product.

For a target product that is staged attacked by a large number of good reviews or bad reviews in some time windows leads to the sudden increase or decrease of the average rating of the products, so that the average scores and the number of reviews show a positive or negative correlation.

Since the probability density curve estimated by the kernel density is smoother, we consider the review published time as the sample point for the density function curve estimation. Since the probability density function of the kernel density estimated by the smoothed kernel is also smooth, we can use the Gaussian kernel function here, as shown in

$$K(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2}. \quad (9)$$

Definition 7 (product review sequence V_p). The product review sequence $V_p = \{v_1, v_2, \dots, v_m\}$ is all the reviews of the product p , which are sorted in turn by review published time, where m is the total number of reviews of the product p , v_i is the i th reviews of the product p , and the range of i is $[1, m]$.

Definition 8 (product review time sequence T_p). The product review time sequence is $T_p = \{t_1, t_2, \dots, t_m\}$, where t_i is the time when the i th review is published.

Definition 9 (product time window I_i). The product time window is a time interval of a review. The time window is defined as shown in

$$I_i = (a_{i-1}, a_i], \quad a_i = i * \Delta t, \quad 1 \leq i \leq k, \quad (10)$$

where Δt is the size of specified time window, $T = t_m - t_1$ is the length of time, k is the number of time windows, $k = T/\Delta t = (t_m - t_1)/\Delta t$, a_{i-1} is the left boundary of time window I_i , and a_i is the right boundary.

Definition 10 (time window review collection H_i). The time window review collection refers to the review collection

whose published time falls within a certain time window, and it is defined as shown in

$$H_i = \{v_j \mid t_j \in (a_{i-1}, a_i], i \in [1, k]\}, \quad (11)$$

where v_j is the j th review of the product, and the corresponding publication time is t_j .

3.3. Metapath Feature Selection. The identified product-related review datasets are modeled as a heterogeneous review information network with feature nodes. In order to reflect the final impact of feature weight on the probability of deceptive review, the feature weight calculation algorithm is introduced into the calculation of the probability of the final deceptive degree of the review.

Definition 11 (heterogeneous information network G). A heterogeneous information network is a graph containing a types of nodes and b types of edges ($a > 1$ or $b > 1$), defined as $G = (N, E)$, where N is a set of all types of nodes and E is a collection of all types of edge. Any $v \in N$ or $\varepsilon \in E$ belongs to a particular type.

Definition 12 (network mode T_G). Given a heterogeneous information network graph $G = (N, E)$, we obtain a network pattern graph $T_G = (A, \Gamma)$, in which there exists a mapping relationship from heterogeneous information networks to network patterns $G = (N, E) \rightarrow T_G(A, \Gamma)$, involving the mapping relationship $\tau : N \rightarrow A$ and mapping relationship $\phi : E \rightarrow \Gamma$. The network pattern $T_G = (A, \Gamma)$ is a graph defined on a collection A of node types and a collection Γ of associated types that describes a new graph structure.

Definition 13 (metapath). The metapath is a path P in the network pattern diagram $T_G = (A, \Gamma)$. The corresponding metapaths of the two nodes A_1 and A_n in T_G are denoted as $A_1(\Gamma_1)A_2(\Gamma_2) \cdots A_{n-1}(\Gamma_{n-1})A_n$. The metapath extends the definition of associations to describe the association between two types of nodes that are not directly connected.

The features extracted from the research on the deceptive reviews are classified into three categories: related features of the review contents, relevant features of the reviewers, and related features of the network resources. The symbolic representations of related concepts and their meanings are illustrated in Table 1.

Features of the reviews include the following: the content features of review, the viewpoints features of review, and the metadata features of review. It is impossible to effectively distinguish the deceptive reviews from normal reviews simply by the features of language semantics, such as content features and viewpoints features, because the deceptive reviewers can mimic normal users' behavior so that they are not easily discoverable. Thus, more effective related features of reviewers are needed. The reviewer related features could be as follows: the feature of the reviewer and the feature of the reviewer's behavior.

With the comparative analysis, all the extracted features are classified according to four division strategies: the reviewers based on the behavior or semantic and the reviews based

TABLE 1: Symbol definition table.

Symbol	Definition
r	Reviewer
V_r	The collection of all the reviews published by Reviewer r
v	Review
v_i	The i th review
$V_{r,i}$	The collection of all the reviews published by Reviewer r on the i th day
e_v	The score of review v
$e_{r,p}$	The score of reviewer r on the product p
E_p	The collection of all the rating scores on the product p

TABLE 2: Features extraction in different strategy.

Features	Reviewers	Reviews
Based on behavior	MNRD	
	RPR	
	RNR	
	BST	RRD
	ERD	ETF
	BDS	
	RD	
Based on semantic	RWR	
	ACS	RPP
		ROW

MNRD: max number of reviews daily, RPR: ratio of positive reviews, RNR: ratio of negative reviews, BST: burstiness, ERD: entropy of ratings distribution, BDS: brand deviation score, RD: rating deviation, RWR: ratio of weekend reviews, RRD: review rating deviation, ETF: early time frame, ACS: average content similarity, RPP: ratio of 1st and 2nd person pronouns, and ROW: ratio of objective words.

on the behavior or semantic. Table 2 shows the distribution of these features of reviews and reviewers.

As the range of different features is inconsistent, which brings inconvenience to the measurement of the index, the above features need to be normalized, and the range of each feature is set to be limited to $[0, 1]$. The larger or smaller the value of different features indicates the abnormal performance.

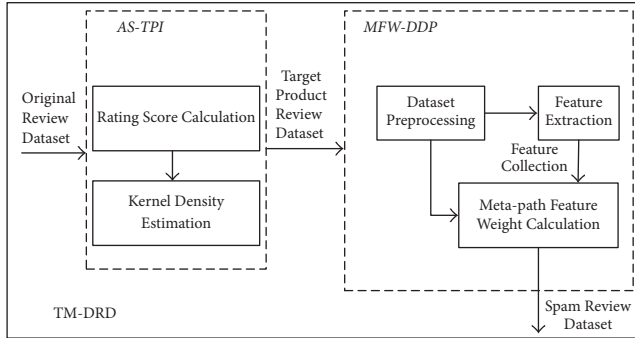
Theoretically there are infinite examples of metapaths in the network, but we can abandon long metapath instances by selecting the path length [26]. According to the small-world phenomenon [27] and the third-degree influence theory [28], it can be inferred that the metapath with a length greater than 3 reflects a very weak association, so we can consider only the metapath whose path length is not greater than 3. Therefore we select the metapaths as shown in Table 3.

4. Our Method

The research on deceptive review detection has mainly focused on improving the accuracy of the results without considering the validity of the test objects. Therefore, we propose a deceptive review detection algorithm based on the target product identification and the metapath feature weight

TABLE 3: Metapath results.

Symbol	Definition
V - V (RPP)	The reviews with the same ratio of 1st and 2nd person pronouns.
V - V (ROW)	The reviews with the same ratio of objective words
V - V (RRD)	The reviews with the same review rating deviation
V - V (ETF)	The reviews with the same early time frame
V - R - R - V (ACS)	The reviews published by the reviewers with the same average content similarity
V - R - R - V (MNRD)	The reviews published by the reviewers with the same max number of reviews daily
V - R - R - V (RPR)	The reviews published by the reviewers with the same ratio of positive reviews
V - R - R - V (RNR)	The reviews published by the reviewers with the same ratio of negative reviews
V - R - R - V (BST)	The reviews published by the reviewers with the same burstiness
V - R - R - V (ERD)	The reviews published by the reviewers with the same entropy of ratings distribution
V - R - R - V (BDS)	The reviews published by the reviewers with the same brand deviation score
V - R - R - V (RD)	The reviews published by the reviewers with the same rating deviation
V - R - R - V (RWR)	The reviews published by the reviewers with the same ratio of weekend reviews

FIGURE 1: Framework of *TM-DRD*.

calculation (*TM-DRD*) for the valid product review dataset. The overall framework is shown in Figure 1.

4.1. AS-TPI Method. In order to identify the target product of deceptive review, we propose a target product identification method based on abnormal score, noted as *AS-TPI*. The original review dataset is statistically processed in the product scoring stage to obtain each product and its corresponding scoring dataset as input to *AS-TPI*.

AS-TPI is divided into two parts. The first part is based on the rating score calculation, which statically identifies the product for the number distribution of reviews on each rating

Input: Product Set P , Review Set V .

Output: Target Product Set P_t

```

(1) for each rating score do
(2)   calculate  $SD_i$ 
(3)   for each product  $p$  in  $P$  do
(4)     calculate  $D_p, D_{p,i}, DIF_{p,i}, \mu_i, \delta_i$ 
(5)     if  $DIF_{p,i}$  not in the confidence interval then
(6)       Add( $DIF_{p,i}$ ) to  $DD_i$ 
(7)       Add( $DD_i$ ) to  $DD$ 
(8)   for each product  $p$  in  $P$  do
(9)     for each rating score do
(10)      if  $DIF_{p,i}$  in  $DD$  then
(11)        Count( $p$ )++
(12)   if Count( $p$ ) > 2 then
(13)     Add( $p$ ) to  $P_t$ 
(14) return  $P_t$ 
  
```

ALGORITHM 1: *StaticTargetProductDetection*(P, V).

level. The second part is based on the estimation of the kernel density to analyze the sudden abnormalities of reviews from the time dimension to dynamically identify the products.

Algorithm 1 is named as *StaticTargetProductDetection*, the number of reviews on each rating level of the product is counted to obtain D_p , then $R_{p,i}$ and s_i , according to the distribution of the number of reviews of the current product with the current rating scale. $DIF_{p,i}$ is calculated by comparing with the result of s_i . According to the Law of Large Numbers, $DIF_{p,i}$ follows a normal distribution. Finally, we set a confidence interval (a significance level) to find the grade difference index that does not satisfy the confidence interval in the normal distribution corresponding to the product grade difference. The pseudocode of static target product detection is shown in Algorithm 1.

In Algorithm 1, lines (2)–(4) calculate the rating score and other related parameters, lines (5)–(7) determine $DIF_{p,i}$ which does not meet the confidence interval, and add to the distribution of differences in the proportion of collection, lines (8)–(13) add p to the suspicious target product set where $DIF_{p,i}$ appear more than two times in the set, and line (14) returns target product set. The time complexity of the algorithm is $O(i * N)$, where i is the rating grades and N is the number of products in the review dataset to be detected.

Algorithm 2 is named as *DynamicTargetProduct-Detection*. In Algorithm 2, review sequence and other related parameters are calculated in lines (2)–(4); lines (5)–(6) calculate the set of extreme points of KDE and filter the extreme points and then add the time window which contains the extreme points to candidate burst time window set; lines (9)–(14) calculate the average score of each time window in the set of candidate time windows and then calculate the difference between the average of the ratings and the average of the overall score of the product. If the difference exceeds the threshold, the count of time windows increases by 1, and if count exceeds $k/2$, we add the product to the target product set. Line (15) returns the target product set.


```

Input: Product Set  $P$ , Review Set  $V$ .
Output: Target Product Set  $P_t$ 
(1) for each product  $p$  in  $P$  do
(2)   calculate  $V_p, T_p$ 
(3)   for each rating score do
(4)     calculate  $w_i$ 
(5)   calculate  $Xp$ 
(6)   Add( $Xp$ ) to  $Xp'$ 
(7)   for  $x_{p_j}$  in  $Xp'$  do
(8)     Add( $x_{p_j}$ ) to  $I_p$ 
(9)   for  $I_i$  in  $I_p$  do
(10)    calculate  $\mu_{p,i}$ 
(11)    if  $|\mu_{p,i} - \mu_p| > \tau$  then
(12)      Count( $p$ )++
(13)  if Count( $p$ ) >  $k/2$  then
(14)    Add( $p$ ) to  $P_t$ 
(15) return  $P_t$ 

```

ALGORITHM 2: DynamicTargetProductDetection(P).

The time complexity of Algorithm 2 is $O(m * N)$, where m is the maximum among the number of time windows, the number of extreme points, and the number of candidate time windows. N is the number of products in the review dataset to be detected.

4.2. MFW-DDP Method. With the above AS-TPI method, we can obtain the target product review dataset. Combining this dataset with a given feature list as input, we propose a method to calculate the metapath based feature weights to calculate the deceptive degree probability of reviews to determine the final category of reviews, noted as MFW-DDP. This method is mainly divided into four steps: feature-based prior probability calculation, feature-based network pattern creation, metapath generation, and classification marking.

Step 1 (feature-based prior probability calculation). The following equation is used to calculate the prior probability s_u of deceptive degree and initialize all the review nodes in the information network graph:

$$s_u = \frac{1}{L} \sum_{l=1}^L f(x_{lu}), \quad (12)$$

where $f(x_{lu})$ represents the a priori probability of the deceptive degree of the review u calculated from feature l .

Step 2 (feature-based network pattern creation). Given a set of feature lists F , constructing a heterogeneous review information network graph $G = (N, E)$, and according to graph G , we can obtain the network pattern $T_G = (A, \Gamma)$.

When the list of features is $\{ACS, MNRD, RPP, RRD\}$, the network pattern is shown in Figure 2. We can figure out that network pattern only contains one different type of node.

Step 3 (metapath generation). As shown in Figure 3, the two dotted lines, respectively, represent the instances of two

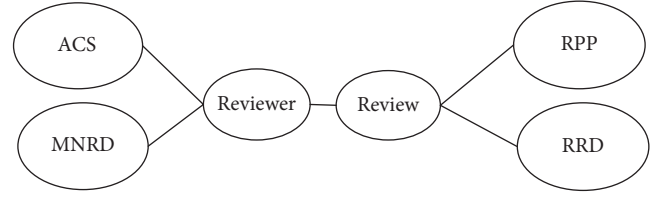
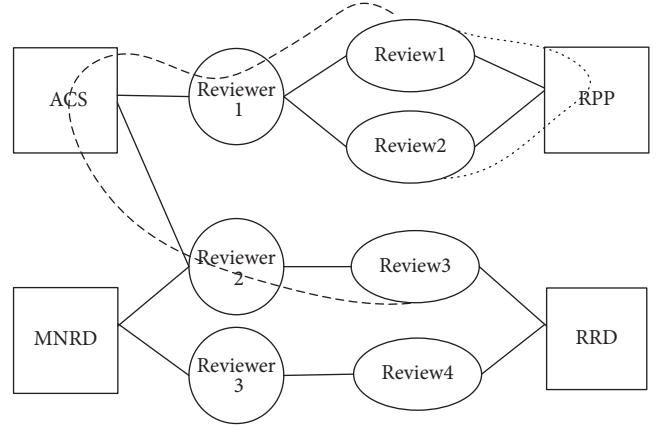
FIGURE 2: Network pattern based on the feature list $\{ACS, MNRD, RPP, RRD\}$.

FIGURE 3: Metapath generation example.

metapaths. If the *Review* node and another *Review* node are associated with the feature *RPP* and their *RPP* values are equal, a metapath is generated, the symbol of which is denoted as *Review-RPP-Review*. If the *Review* node and another *Review* node are associated with the feature *ACS* and their *ACS* values are equal, a metapath is generated with the symbol *Review-Reviewer-ACS-Reviewer-Review*.

Step 4 (classification marking). Classification marking includes two steps: feature weight calculation and classification marking. The weight calculation determines the importance of identifying each feature of the deceptive review. The classification marking calculates the final deceptive probability of each review. To help consumers seek credible information, most current work apply mainly qualitative approaches to investigate the credibility of reviews or reviewers [29]. We adopt the probability of deceptive degree for the review node to quantify the credibility of reviewers.

The weight is calculated as shown in (13). The classification marking is defined as (14). The probability of deceptive degree for the current review node is estimated according to (15).

$$W_{pi} = \frac{\sum_{u=1}^n \sum_{v=1}^n mp_{u,v}^{pi} \times s_u \times s_v}{\sum_{u=1}^n \sum_{v=1}^n mp_{u,v}^{pi}}, \quad (13)$$

$$P_{u,v} = 1 - \prod_{i=1}^L (1 - mp_{u,v}^{pi} \times W_{pi}), \quad (14)$$

Input: Review Set V , Reviewer Set R , Feature Set F
Output: Deceptive review degree probability set P , feature weight set W

- (1) **for** each reviews u in V **do**
- (2) *calculate* s_u
- (3) Define the network pattern $schema(A, \Gamma)$
- (4) **for** $u, v \in V$ **do**
- (5) **for** $p_l \in schema$ **do**
- (6) *calculate* mp_u^{pl}, mp_v^{pl}
- (7) **if** $mp_u^{pl} = mp_v^{pl}$ **then**
- (8) $mp_{u,v}^{pl} = mp_u^{pl}$
- (9) **Add** u, v to V'
- (10) **for** $p_l \in schema$ **do**
- (11) *calculate* w_{pl}
- (12) **for** $u, v \in V'$ **do**
- (13) *calculate* $P_{u,v}$
- (14) *calculate* P_u
- (15) **return** P, W

ALGORITHM 3: $TM-DRD(V, R, F)$.

$$P_u = \frac{\sum_{v=1}^n P_{u,v}}{n}. \quad (15)$$

According to the above calculation, we can obtain the deceptive probability set P of all the review nodes.

4.3. $TM-DRD$ Algorithm. With the result of target product identification method based on abnormal score ($AS-TPI$) and the calculation method of deceptive degree probability of reviews based on the metapath feature weights ($MFW-DDP$), we can determine the final category of reviews. Our proposed deceptive review detection algorithm based on the target product identification and the metapath feature weight calculation ($TM-DRD$) is shown in Algorithm 3.

In Algorithm 3, lines (1)-(2) calculate the a priori probability for each review. Line (3) defines the network pattern. Lines (4)–(9) calculate the probability of each feature associated value of the metapath corresponding to two review nodes. The weight of two review nodes associated with each feature is calculated in lines (10)-(11). The probability of the final deceptive degree of the review node is calculated in lines (12)–(14). Line (15) returns the degree probability of deceptive review set and the feature weight set.

The time complexity of Algorithm 3 is $O(|V| * |M|)$, where $|V|$ represents the number of review nodes in the heterogeneous review information network and $|M|$ represents the number of feature sets (constant).

5. Experiment

5.1. Experimental Setup. The experimental environment is described in Table 4. To verify the validity of the proposed algorithm, a real, reliable, and accurate dataset plays a crucial role in the deceptive review detection. Therefore, we try to test on the review datasets in real environment. In the experiment, we use the review datasets *YelpChi* and *YelpNYC*

TABLE 4: Experimental environment table.

Item	Content
CPU	Intel Core i5 3.30 GHz dual-core
RAM	2 GB
Hard disk	500 GB
Operating system	Microsoft Windows 7 32-bit
Development environment	Python 2.7.3
Development tools	Matlab R2014a + Eclipse
Database	MySQL5.6.26

TABLE 5: The distribution table of reviews, products, and reviewers in *Yelp*.

Dataset	Reviews number	Reviewers number	Products number
<i>YelpChi</i>	67395	38063	201
<i>YelpNYC</i>	359053	160225	923

from *Yelp*, a famous travel website, provided by [30]. The *YelpChi* [30] covers about 67,395 reviews, 38,063 reviewers, and 201 products for hotels and restaurants in the Chicago area from October 12, 2004, to October 8, 2012. The *YelpNYC* [30] covers about 359,053 restaurants related reviews, 160,225 reviewers, and 923 products in the New York City area from October 20, 2004, to July 1, 2015. The specific distribution of reviews, production, and reviewers is shown in Table 5. Six attributes extracted for structured processing are saved to the database. The reviews in this dataset contain the deceptive markups (fake or not) of each review. The annotation results are generated with the *Yelp* filtering algorithm [31].

5.2. Evaluation Index. In order to assess the performance of the target product identification and deceptive review detection methods, we should utilize the accepted assessment methods and evaluation criteria.

We adopt the widely used accuracy as an evaluation index to the behavior of $AS-TPI$. The accuracy λ is defined as the ratio of the number of target products M to the number of suspicious target products N identified by Algorithms 1 and 2, as shown in

$$\lambda = \frac{M}{N} \times 100\%. \quad (16)$$

There are two kinds of evaluation indexes to evaluate the recognition results of the algorithm comprehensively. The $TM-DRD$ algorithm would adopt the second one.

The first evaluation index is the classification model evaluation indicators: Precision rate, Recall rate, and accuracy computed from Precision and Recall rate. $F1$ value is the reconciled average of Precision and Recall rate. False positive FPR and true positive TPR rates characterize the recognition accuracy and recognition range. The second evaluation index is the ranking model to evaluate the performance of the algorithm, including the PR curve, the ROC curve, and the area covered by the curve, corresponding to Average Precision (AP) and Area under Curve (AUC), which indicate

the trade-off evaluation index of test results in the Precision and Recall rate, as shown in (17) and (18).

$$AP = \sum_{i=1}^n \frac{i}{I(i)}, \quad (17)$$

where n represents the number of reviews, i represents the position of the review in the sorted set of reviews, and I represents the position set of the review in the sorted set of reviews.

$$AUC = \sum_{i=2}^n (FPR(i) - FPR(i-1)) * (TPR(i)), \quad (18)$$

where $FPR(i)$ represents the false positive rate of the i th review and $TPR(i)$ represents the true positive rate of the i th review.

5.3. Experimental Results and Analysis

5.3.1. Target Product Identification Experiment. The experiment uses the *YelpNYC* [30] review dataset. The purpose of the experiment is to identify the target product attacked by a large number of deceptive reviews. The original dataset is filtered, the threshold value τ is set to 0.3, and the time window is two weeks. The target product identification method based on abnormal score was used for screening to obtain the collection of suspicious target products. We invite 3 online shopping experienced college students as judges to manually evaluate the collection. In order to reduce the impact of subjective factors or other random factors on the evaluation results, we consider the marking results of most evaluators as the final mark according to the voting principle.

Then, a time window Δt is set, and, for each time window size, $\tau \in \{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0\}$, each τ is used as a time window score mean difference parameter in the target product identification algorithm based on the nuclear density estimation. Differentiate the mean score of the review burst time window and calculate the collection of suspicious target products. Then we observe the influence of the change of τ on the target product recognition rate.

The marking results of 3 judges are shown in Table 6. According to the confirmation of the final marker, there are 35 true target products finally determined in the evaluation target products in the experiment; that is, the recognition rate was $\lambda = 35/42 * 100\% = 83.33\%$. It shows that the target product identification method based on abnormal score has high recognition accuracy. The target product-related review collection only accounted for 15.99% of the original review dataset. It shows that a large number of meaningless review data exist in original review dataset. If we detect deceptive review directly, it will lead to the decline in detection efficiency. Therefore, the target product identification method solves the overall sparseness and imbalance of deceptive reviews.

As shown in Figure 4, under the setting of time window size Δt of 7 days and 14 days, respectively, the recognition rate curve decreases with the increase of threshold parameter τ . The recognition rate drops to 0 until $\tau = 0.7$ and then remains

TABLE 6: Artificial premark results for the target product.

Judge	The number of premark target products
Judge 1	34
Judge 2	35
Judge 3	34

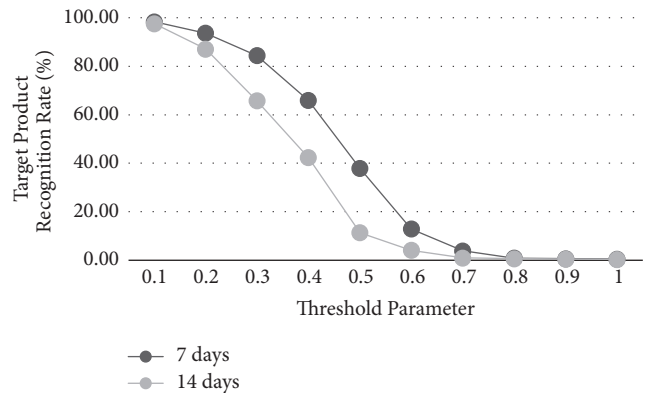


FIGURE 4: The influence of threshold parameters on the recognition rate of target products.

unchanged at 0. The target product recognition rate obtains the highest value when $\tau = 0.1$. It is usually short-term behavior that a large number of fake reviews are published by fake reviewers periodically, so when the smaller appropriate value of the time window is set, we can capture burst situation of reviews, so there is higher recognition rate when the time window is set to 7 days.

5.3.2. Comparative Experiment of Deceptive Review Detection Related Methods. The experiments in this section will compare the performance of the *TM-DRD* and the *NFC* [24], *SpEagle* [30], and *NetSpam* [32] on accuracy indices such as AP and AUC. We verify the impact of the target product review dataset and feature weight calculation on the detection efficiency of *TM-DRD* and the accuracy of the test results.

The experiment uses four review datasets: *YelpChi* [30], *YelpChiOP*, *YelpNYC* [30], and *YelpNYCOP*. The datasets *YelpChiOP* and *YelpNYCOP* are, respectively, related review datasets on the target product identified by the fusion algorithm based on the anomaly scores proposed in chapter 4 from the original data sets *YelpChi* and *YelpNYC* [30]. Next, we will compare the performance of *TM-DRD* and *NFC* [24], *SpEagle* [30], and *NetSpam* [32] in AP and AUC, respectively, on the above 4 review datasets. We analyze the impact of feature weights on the accuracy of deceptive review detection.

In order to verify the impact of feature weight on accuracy and find out whether there is a relationship between weight and accuracy, AP index is used here to measure the accuracy. The equation based on ranking difference set is adopted here, as shown in the following:

$$\rho = 1 - \frac{6 \sum_{i=1}^N d_i^2}{N(N^2 - 1)}, \quad (19)$$

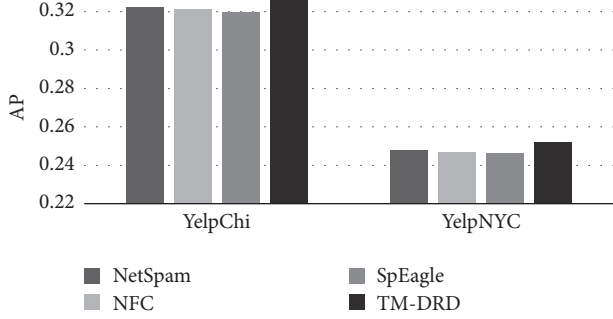


FIGURE 5: The AP for TM-DRD and SpEagle, NFC, and NetSpam in different datasets.

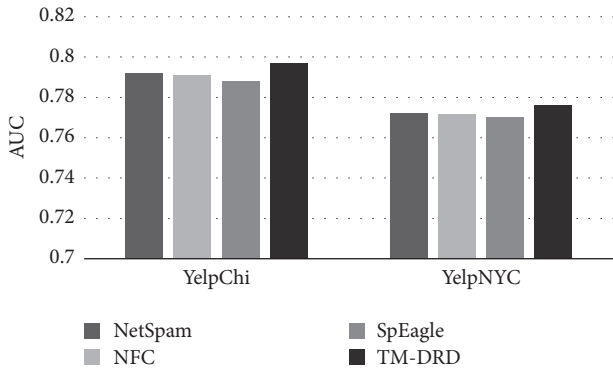


FIGURE 6: The AUC for TM-DRD and SpEagle, NFC, and NetSpam in different datasets.

where $d_i = x_i - y_i$ represents the i th element in the ranking differential set d , x_i represents the i th element in the isometric rank of the X variable, similarly, y_i represents the i th element in the ranking of Y variables, and N represents the number of elements in the X -variable set or Y -variable set. The two are equal, and here N is 13, the number of features.

We use *TM-DRD* and *NFC* [24], *SpEagle* [30], and *NetSpam* [32], respectively, to calculate the deceptive degree probability of each review in the experimental review datasets above. We sort all the reviews according to the deceptive probability in descending to obtain a list of reviews. Next, AP and AUC values are calculated according to (17) and (18), respectively. The experimental results are shown in Figures 5, 6, 8, and 9. We observe and analyze the test results in the performance of those two indicators. At the same time, experiments on the impact of the proportion of deceptive reviews in the datasets on the accuracy of the test results are carried out, as shown in Figure 7. Figure 10 shows the distribution of the features weight in the *YelpNYC*. The results show that behavior-based features are assigned higher weights than semantic-based features. The features in reviewer-behavior classification strategy UB in experimental data sets have higher weight and better performance. The feature list {RPP, ROW, RRD, ETF, ACS, MNRD, RPR, RNR, BST, ERD, BDS, RD, RWR} is obtained according to the definition order of the features in Section 3.3.

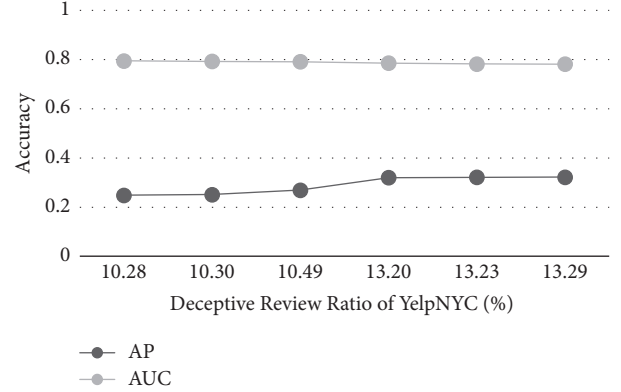


FIGURE 7: The AP and AUC of *TM-DRD* in different deceptive review ratios.

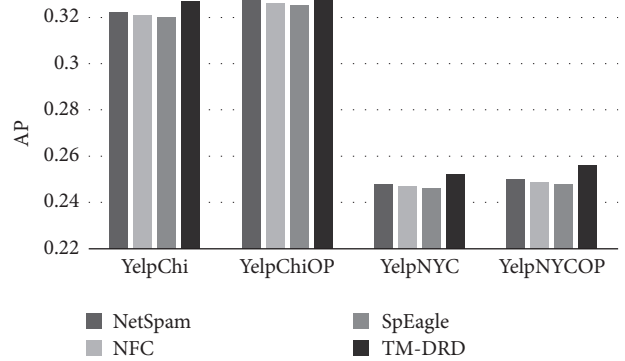


FIGURE 8: The AP for TM-DRD and SpEagle, NFC, and NetSpam in different datasets.

As shown in Figures 5 and 6, the detection results of the *TM-DRD* on the same review datasets are superior to others on the indicators of AP and AUC. The results of deceptive review detection on the *TM-DRD* algorithm on different review datasets are very different in the AP index. The difference between the detection results of *YelpChi* and *YelpNYC* [30] is more than 0.05 in the AP index, but the difference in the AUC index is far below 0.05. As shown in Figure 7, with the increasing proportion of deceptive review in the datasets, the AP index of *TM-DRD* algorithm is increasing, but the AUC index is almost unchanged.

Since the experimental data are all annotated, the proportion of deceptive review in the *YelpChi* and *YelpNYC* [30] is, respectively, calculated to be 13.2% and 10.3%. The proportion of deceptive review in the *YelpChi* [30] dataset is 13.23% and 13.29%, respectively. The ratio of deceptive review on restaurants and hotels in the *YelpNYC* [30] is 10.49% and 10.28%. As the proportion of deceptive review in the datasets increasing, the probability of review being detected as deceptive review increases. More and more reviews are identified as deceptive review, while the AUC values are almost unchanged. It shows that the AUC index has nothing to do with the proportion of deceptive review, it depends on the list of reviews after sorting.

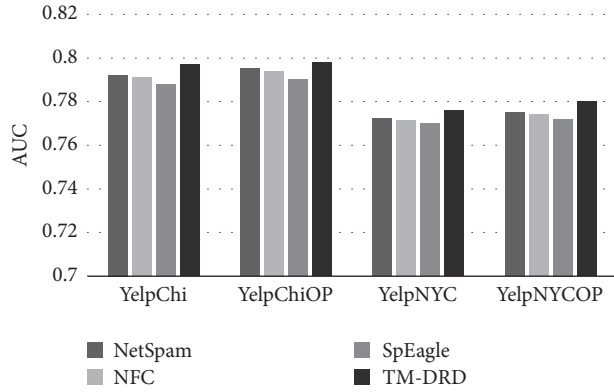


FIGURE 9: The AUC for TM-DRD and SpEagle, NFC, and NetSpam in different datasets.

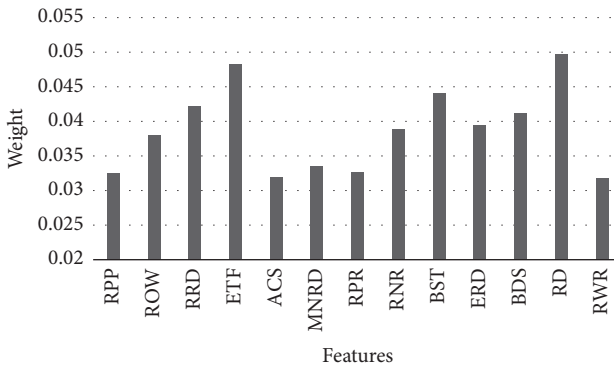


FIGURE 10: Features weight distribution of YelpNYC.

As shown in Figures 8 and 9, the performance of the AP and AUC indicators on the related review datasets *YelpChiOP* and *YelpNYCOP* are, respectively, better than the corresponding original review datasets *YelpChi* and *YelpNYC* [30]. The AP indicators and the AUC indicators improve on different review datasets.

As shown in Figure 11, the 13 levels of feature weights and their AP levels used in the experiment correspond to the coordinate points in the figure, respectively. From the figure, it can be seen that the overall trend of the accuracy rate increasing with the increase of weight level; that is, the higher the weight value, the higher the accuracy of the detection result. The feature weight is closely related to the accuracy of the final test result of the deceptive review detection. The feature weight calculated through the *TM-DRD* algorithm indicates the ability of the feature to distinguish the deceptive review, and the feature with the greater weight is more effective in the deceptive review detection. With the increase of weight, these features are accompanied by the corresponding increase of the test results on the AP, AUC, and other indicators, which shows that the feature weight calculation improves the accuracy of deceptive review detection test results.

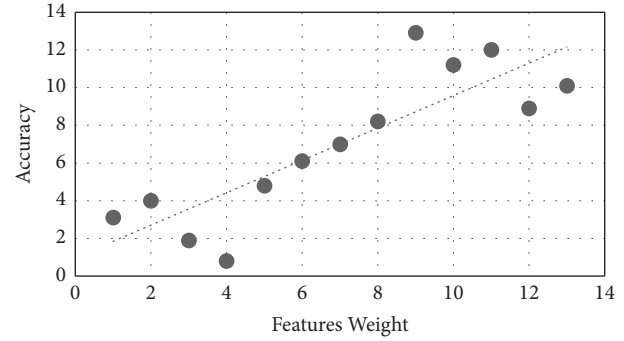


FIGURE 11: Relationship between features weight and accuracy.

6. Conclusion

In this paper, we analyze the existing research on the deception review detection and design a deceptive review detection algorithm based on target product identification and metapath feature weight calculation, *TM-DRD* algorithm. In this algorithm, we firstly analyze the different deceptive review states of the product type and then design the static target product detection algorithm based on the difference of the grade score and the dynamic target product detection algorithm based on the kernel density estimation for different states. Based on these proposed algorithms, we identify the target product. Then, we construct the related review datasets as a heterogeneous review information network and calculate the weight of the metapath feature of the target product. In the following, with the metapath based feature weights, we calculate the deceptive degree probability of reviews to determine the final category of reviews. Finally, we conduct several experiments to evaluate the accuracy and efficiency of the proposed *TM-DRD* algorithm. We analyze the experiment results, respectively, according to the target product identification and the deceptive review detection. In particular, comparative analysis of the performance of the proposed *TM-DRD* algorithm and the *NFC* [24], *SpEagle* [30], and *NetSpam* [32] on AP, AUC, and other evaluation indicators shows that the method of feature weight calculation is very helpful to improve the accuracy of the deceptive review detection.

Conflicts of Interest

All the authors do not have any possible conflicts of interest.

Acknowledgments

This work was supported by National Natural Science Fund of China under Grant 61502185 and the Fundamental Research Funds for the Central Universities (no. 2017KFYXJJ071).

References

- [1] M. Tsikerdekis and S. Zeadally, "Online deception in social media," *Communications of the ACM*, vol. 57, no. 9, pp. 72–80, 2014.

- [2] S. KC and A. Mukherjee, "On the Temporal Dynamics of Opinion Spamming," in *Proceedings of the the 25th International Conference on World Wide Web, WWW 2016*, pp. 369–379, Montreal, Canada, April 2016.
- [3] C. Xu, "Detecting collusive spammers in online review communities," in *Proceedings of the the sixth workshop on Ph.D. Students in Information and Knowledge Management, PIKM@CIKM 2013*, pp. 33–40, San Francisco, Calif, USA, November 2013.
- [4] H. Li, G. Fei, S. Wang et al., "Bimodal Distribution and Co-Bursting in Review Spam Detection," in *Proceedings of the the 26th International Conference*, pp. 1063–1072, Perth, Australia, April 2017.
- [5] M. Crawford, T. M. Khoshgoftaar, J. D. Prusa, A. N. Richter, and H. Al Najada, "Survey of review spam detection using machine learning techniques," *Journal of Big Data*, vol. 2, no. 1, article no. 23, 2015.
- [6] M. Ott, Y. Choi, C. Cardie, and J. T. Hancock, "Finding deceptive opinion spam by any stretch of the imagination," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT '11)*, vol. 1, pp. 309–319, Association for Computational Linguistics, Portland, Ore, USA, June 2011.
- [7] F. Li, M. Huang, Y. Yang et al., "Learning to identify review spam," in *Proceedings of the 22nd International Joint Conference on Artificial Intelligence*, pp. 2488–2493, AAAI Press, Barcelona, Spain, 2011.
- [8] J. K. Rout, A. Dalmia, K.-K. R. Choo, S. Bakshi, and S. K. Jena, "Revisiting semi-supervised learning for online deceptive review detection," *IEEE Access*, vol. 5, pp. 1319–1327, 2017.
- [9] S. Feng, R. Banerjee, and Y. Choi, "Syntactic stylometry for deception detection," in *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics, ACL 2012*, pp. 171–175, Jeju Island, Korea, July 2012.
- [10] S. Feng, L. Xing, A. Gogar, and Y. Choi, "Distributional footprints of deceptive product reviews," in *Proceedings of the 6th International AAAI Conference on Weblogs and Social Media, ICWSM 2012*, pp. 98–105, Dublin, Ireland, June 2012.
- [11] J. Li, C. Cardie, and S. Li, "TopicSpam: A topic-model-based approach for spam detection," in *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, ACL 2013*, pp. 217–221, bgr, August 2013.
- [12] T. Lappas, "Fake Reviews: The Malicious Perspective," in *Natural Language Processing and Information Systems*, vol. 7337 of *Lecture Notes in Computer Science*, pp. 23–34, Springer, Berlin, Germany, 2012.
- [13] H. Li, Z. Chen, and A. Mukherjee, "Analyzing and detecting opinion spam on a large-scale dataset via temporal and spatial patterns," in *Proceedings of the Ninth International Conference on Web and Social Media, ICWSM, 2015, University of Oxford*, pp. 634–637, Oxford, UK: the, 2015.
- [14] J. Ye, S. Kumar, and F. Akoglu, "Temporal opinion spam detection by multivariate indicative signals," in *Proceedings of the Tenth International Conference on Web and Social Media*, pp. 743–746, Cologne, Germany, 2016.
- [15] E. Lim, V. Nguyen, N. Jindal, B. Liu, and H. W. Lauw, "Detecting product review spammers using rating behaviors," in *Proceedings of the the 19th ACM international conference*, p. 939, Toronto, Canada, October 2010.
- [16] S. Xie, G. Wang, S. Lin, and P. S. Yu, "Review spam detection via temporal pattern discovery," in *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2012*, pp. 823–831, Beijing, China, August 2012.
- [17] A. Mukherjee, A. Kumar, B. Liu et al., "Spotting opinion spammers using behavioral footprints," in *Proceedings of the the 19th ACM SIGKDD international conference*, pp. 632–640, Chicago, Ill, USA, August 2013.
- [18] Y. Zhang and J. Lu, "Discover millions of fake followers in Weibo," *Social Network Analysis and Mining*, vol. 6, no. 1, article no. 16, 2016.
- [19] G. Wang, S. Xie, B. Liu, and P. S. Yu, "Review graph based online store review spammer detection," in *Proceedings of the 11th IEEE International Conference on Data Mining, ICDM 2011*, pp. 1242–1247, Vancouver, Canada, December 2011.
- [20] L. Li, W. Ren, B. Qin et al., "Learning Document Representation for Deceptive Opinion Spam Detection//Processing," in *Proceedings of the of 14th China National Conference on Chinese Computational Linguistics*, pp. 393–404, Guangzhou, China, 2015.
- [21] C. Xu, J. Zhang, K. Chang, and C. Long, "Uncovering collusive spammers in Chinese review websites," in *Proceedings of the 22nd ACM international conference*, pp. 979–988, San Francisco, California, USA, October 2013.
- [22] Z. Wang, S. Gu, X. Zhao, and X. Xu, "Graph-based review spammer group detection," *Knowledge & Information Systems*, vol. 3, no. 2017, pp. 1–27, 2017.
- [23] G. Fei, A. Mukherjee, B. Liu et al., "Exploiting burstiness in reviews for review spammer detection," in *Proceedings of the 7th International Conference on Weblogs and Social Media, ICWSM, 2013, The AAAI Press, Cambridge, Mass, USA, 2013*.
- [24] J. Ye and L. Akoglu, "Discovering Opinion Spammer Groups by Network Footprints," in *Machine Learning and Knowledge Discovery in Databases*, vol. 9284 of *Lecture Notes in Computer Science*, pp. 267–282, Springer International Publishing, Cham, 2015.
- [25] K. Dhingra and S. K. Yadav, "Spam analysis of big reviews dataset using Fuzzy Ranking Evaluation Algorithm and Hadoop," *International Journal of Machine Learning and Cybernetics*, 2017.
- [26] M. A. Hasan, "Link Prediction using Supervised Learning," *Proceedings of SDM Workshop on Link Analysis Counterterrorism & Security*, vol. 30, no. 9, pp. 798–805, 2006.
- [27] D. J. Watts and S. H. Strogatz, "Collective dynamics of small-world networks," *Nature*, pp. 440–442, 1998.
- [28] K. Walker Susan, "Connected: The Surprising Power of Our Social Networks and How They Shape Our Lives," *Journal of Family Theory Review*, vol. 3, no. 3, pp. 220–224, 2011.
- [29] Y. Wang, S. C. F. Chan, H. Va Leong, G. Ngai, and N. Au, "Multi-dimension reviewer credibility quantification across diverse travel communities," *Knowledge & Information Systems*, vol. 49, no. 3, pp. 1071–1096, 2016.
- [30] S. Rayana and L. Akoglu, "Collective opinion spam detection: Bridging review networks and metadata," in *Proceedings of the 21st ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD 2015*, pp. 985–994, aus, August 2015.
- [31] A. Mukherjee, V. Venkataraman, B. Liu, and N. Glance, "What yelp fake review filter might be doing?" in *Proceedings of the 7th International AAAI Conference on Weblogs and Social Media, ICWSM 2013*, pp. 409–418, usa, July 2013.
- [32] S. Shehnpoor, M. Salehi, R. Farahbakhsh, and N. Crespi, "NetSpam: A Network-Based Spam Detection Framework for Reviews in Online Social Media," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 7, pp. 1585–1595, 2017.