Complexity

Analysis and Applications of Complex Social Networks

Lead Guest Editor: Katarzyna Musial
Guest Editors: Piotr Brodka and Pasquale De Meo

Analysis and Applications of Complex Social Networks

Complexity

Analysis and Applications of Complex Social Networks

Lead Guest Editor: Katarzyna Musial
Guest Editors: Piotr Brodka and Pasquale De Meo

Editorial Board

Contents

Hindawi
Complexity
Volume 2017, Article ID 3014163, 2 pages
https://doi.org/10.1155/2017/3014163

Editorial
Analysis and Applications of Complex Social Networks

Katarzyna Musial,[1] Piotr Bródka,[2] and Pasquale De Meo[3]

[1]*University of Technology Sydney, Sydney, NSW, Australia*
[2]*Wroclaw University of Science and Technology, Wroclaw, Poland*
[3]*University of Messina, Messina, Italy*

Correspondence should be addressed to Katarzyna Musial; katarzyna.musial-gabrys@uts.edu.au

Received 22 November 2017; Accepted 28 November 2017; Published 27 December 2017

Social networks are everywhere and research aiming at analysing and understanding these structures is growing year by year as its outcomes enable us to understand different social phenomena including social structures evolution, communities, spread over networks, and dynamics of changes in networks. This huge interest in the analysis of large-scale social networks resulted in a lot of new approaches, methods, and techniques but with every advancement in this area, we uncover new challenges and new levels of complexity in the network universe that are far from being explored and addressed. The increasing complexity of tasks to be performed in terms of network analysis together with the volume, variety of social data about people and their interactions, and velocity with which this data is generated in the online world pose new requirements and challenges on researchers. One of them is: how to build accurate methods that would be able to cope with this vast amount of data. This issue is a result of an attempt to address these emerging challenges.

One of the goals of this special issue is to identify the areas where social network analysis can be applied and generate knowledge not accessible through other types of analysis. We also aimed at showing that analysis of large-scale, real-world social networks underpinned by fundamental research is the direction to take when it comes to the future of complex social network analysis. We emphasize that in the world of network science fundamental research and application-driven research are equally important and they need to go together to generate significant academic, societal, and commercial impact.

The variety of papers published in this special issue shows that there is a long list of topics that have not yet been comprehensively researched. These papers also show the future challenges and trends in analysis and applications of complex social networks. The articles in this issue cover topics starting from very practical and application-based ones such as (i) case study on evolution of collaborative systems, (ii) community detection and analysis, (iii) link dynamics, (iv) spreading processes including modelling behaviour of rumour maker and influence maximization problem, and (v) frangibility in social consensus system but it also presents some research more fundamental in its nature including optimisation of structural network properties and measuring complexity of networks.

Published papers show that although all of the presented topics have been researched for many years now, there is still space and need for new contributions. Challenges change their nature as we face vast amounts of heterogeneous data that are continuously generated. Network dynamics, communities, spread analysis, consensus formation, network complexity, and structural properties are topics that are trending in research community all over the world. Those are very hard problems to address because of their complexity originating from two sources (i) system, variety of connections, attributes of nodes and connections, nontrivial structure and dynamics of a system, and (ii) process, evolution driven by variety of factors including external ones that are very hard to capture, spreading over complex structure of multiple processes, or needed process adaptation connected with evolving structure. Thus, there is a continuous need to create

cross-disciplinary teams that would work on those challenges having a holistic view of the problem. So our work does not stop here, and we aim at continuing to bring together people from different fields to work on the topics covered within this special issue.

Acknowledgments

This special issue is an outcome of hard work of a number of people and could not happen without the support of our collaborators. This work was possible thanks to the researchers who provided their anonymous reviews. Finally, we are most grateful to the authors for their valuable contributions and for their willingness and efforts to improve their papers in accordance with the reviewers' suggestions and comments.

Katarzyna Musial
Piotr Bródka
Pasquale De Meo

Hindawi
Complexity
Volume 2017, Article ID 4152705, 10 pages
https://doi.org/10.1155/2017/4152705

Research Article

Predicting the Currency Market in Online Gaming via Lexicon-Based Analysis on Its Online Forum

Young Bin Kim,[1] Kyeongpil Kang,[2] Jaegul Choo,[2] Shin Jin Kang,[3] TaeHyeong Kim,[1] JaeHo Im,[2] Jong-Hyun Kim,[4] and Chang Hun Kim[2]

[1]*Interdisciplinary Program in Visual Information Processing, Korea University, Seoul, Republic of Korea*
[2]*Department of Computer Science and Engineering, Korea University, Seoul, Republic of Korea*
[3]*School of Games, Hongik University, Sejong, Republic of Korea*
[4]*Department of Software Application, Kangnam University, Yongin, Republic of Korea*

Correspondence should be addressed to Chang Hun Kim; chkim@korea.ac.kr

Received 6 May 2017; Revised 15 September 2017; Accepted 19 October 2017; Published 24 December 2017

Academic Editor: Katarzyna Musial

Transactions involving virtual currencies are becoming increasingly common, including those in online games. In response, predicting the market price of a virtual currency is an important task for all involved, but it has not yet attracted much attention from researchers. This paper presents user opinions from online forums in a massive multiplayer online game (MMOG) setting widely used around the world. We propose a method for predicting the next-day rise and fall of the currency used in an MMOG environment. Based on analysis of online forum users' opinions, we predict daily fluctuations in the price of a currency used in an MMOG setting. Focusing specifically on the World of Warcraft game, one of the most widely used MMOGs, we demonstrate the feasibility of predicting the fluctuation in value of virtual currencies used in this game community.

1. Introduction

Individuals can engage in countless interactions over the Internet owing to advancements in network technologies and unprecedented computing power. The massive multiplayer online game (MMOG) environment is becoming popular with an increasing number of users. In this MMOG environment, many users carry out economic activities with a variety of purposes [1, 2]. In the World of Warcraft (WoW) MMOG, users engage in transactions using a particular virtual currency called Gold. In Second Life—another well-known virtual world—a virtual currency called Linden dollars is used in transactions related to user-created items, such as houses and clothes [1].

The virtual economies based on these virtual currencies enable users to engage more fully in MMOGs, sometimes even achieving real economic value [3, 4]. Transactions between virtual currencies and real currencies have been gradually increasing [1, 3, 4]. Most MMOG-related virtual currencies are exchanged with real currencies on eBay or with crypto-currencies such as Bitcoin [1, 5]. As the number of users engaging in such transactions increases, the market size is expected to grow [5]. Many researchers have investigated the MMOG environment and related situations [1–7]. However, most of these researchers have focused on the transaction environment rather than on the currency value [1].

Real financial transactions have, of course, been extensively studied. Research on techniques for predicting the stock price, for example, dates to the commencement of stock trading [8]. More recently, stock market trends have been predicted using machine learning techniques, such as neural networks and support vector machines [9–13]. Data used for training some prediction models have been based on financial news [11, 12]. Several previous studies focused on the analysis of stock markets using web-based data [9, 10]. However, few researchers have attempted to valuate virtual currencies and predict their future values.

Numerous transactions involving virtual currencies occur between online game users, forming a large market

[1, 3, 4]. Thus, the ability to predict the next-day rise and fall of virtual currencies in conjunction with cash transactions has been considered important. The ability to predict the market prices of virtual currencies would help transaction agents engage in reasonable transactions and assist game developers in managing virtual environments as well as identifying and solving issues in virtual economies [1].

Virtual currencies for MMOGs exist in economic systems where an enormous amount of bidding competition is allowed, which is not the case for real currencies [14]. All economic agents are fully informed, to the extent that there is no information asymmetry [1, 14]. Suppliers and consumers of virtual currencies act primarily as rational profit maximizers. Fewer variables exist for moderating the value fluctuations of virtual currencies than is the case with real currencies; hence, it is possible to observe value fluctuations easily.

MMOG environments are sources of "Big Data" and thus are suitable for the study of many subjects. In the game Pardus [15], data acquisition makes it feasible to study social theories in large-scale virtual-world populations [16–22]. Various approaches to analyzing the structure and dynamic evolution of social networks in virtual worlds have been developed and have yielded significant findings [23, 24]. Related studies based on these findings have been conducted from a variety of perspectives [25–30].

Based on the above-mentioned studies, we have developed a method for predicting daily fluctuations in the value of a currency used in an MMOG environment via user opinion analysis. In this study, we demonstrated the application of this method for predicting daily fluctuations of the virtual currency used in the WoW game, the MMOG with the largest active user base.

The work by Kim et al. [1] parallels our research in terms of the objective; however, their predictions were based on trading data from only a small number of users who purchased virtual currencies in cash, which limits the quantity of available data. This makes it difficult to check the data on all daily transactions; therefore, the set price is not reliable. Tokens sold in WoW can earn in-game currency if sold by the developer via a qualified path; thus, experiments can be conducted based on more obvious data and using all transactions.

Furthermore, in the technical aspects, existing studies conducted sentiment analysis in a simple form, but the contribution of our study lies mainly in the application of our novel text mining approach based on a custom-built lexicon. That is, to extract a meaningful feature of a document or a time point (containing documents posted in it), we first build a set of keywords, or a lexicon, defining a particular concept or notion, which we call the process of *concept building*. Afterwards, we measure how strongly such a concept is manifested in a given document or a time point, by counting those keywords (and possibly other relevant keywords) occurring in it. Building a custom concept and its lexicon plays a crucial role in analyzing a domain-specific document corpus, and, in our paper, we aim at revealing the signals from textual data that can lead to price prediction. For example, if we use an off-the-shelf sentiment analysis method that uses standard lexicon for positive and negative keywords, then there would be many domain-specific keywords not captured by the method, that is, false negatives.

Instead, we build our own lexicons closely related to the price of a virtual currency in the game of WoW. In this game, users buy tokens with cash at a fixed price (for example, 20 US dollars in the North America (NA) region or 20 Euro in the Europe (EU) region) and sell them to other users in exchange for the virtual currency referred to as Gold. The token/Gold exchange rate changes according to the demand and supply, as shown in Figure 1.

In general, acquiring Gold in the game environment requires an extended period of time and effort, but such time and effort can be reduced if tokens are bought in cash and then sold to another user for Gold. Tokens that are bought with Gold are generally used to buy time available for using the virtual world. Such tokens have two advantages. First, tokens prevent the virtual currency from being illegally transacted in cash. Second, the value of tokens can be traced continuously. We aim to predict the next-day rise and fall of the token/Gold exchange rate (also referred to as the token price) based on user opinion data analyzed on consecutive days. The proposed method can predict fluctuations in the value of virtual currency in the MMOG environment and can be applied to the selling/buying of virtual currency, allowing developers to determine the numerical values and probabilities of multiple effects that would otherwise be difficult to identify.

2. Proposed Approach

As shown in Figure 2, the proposed approach is characterized as follows. The data extracted include users' opinions concerning the MMOG environment and the prices of the virtual currency used in the transactions. User opinions are sorted with respect to their scores of our custom-built concept and then the causation between the user opinion and the virtual currency price is determined by conducting a causality analysis. Based on the user opinions scored afterward and the machine learning model, we predict the rise and fall of the virtual currency price in the next day.

3. Data Crawling

Initially, data required to generate the prediction model is crawled. We gathered data from official WoW forums. People use these Internet forums to upload postings and exchange opinions regarding particular topics of common interest [1, 31–35]. Therefore, such online forums are good sources of information to gauge the daily reactions of many users to certain MMOGs. Communities or forums are widely used in MMOGs for information exchange [31]. According to a previous study [1], WoW links its forums to economic activities among users, which we found to be relevant to predicting the fluctuations in the current number of virtual-world users. Topics and relevant replies posted by users on general discussion boards in official WoW forums were crawled. We also crawled the time when each comment

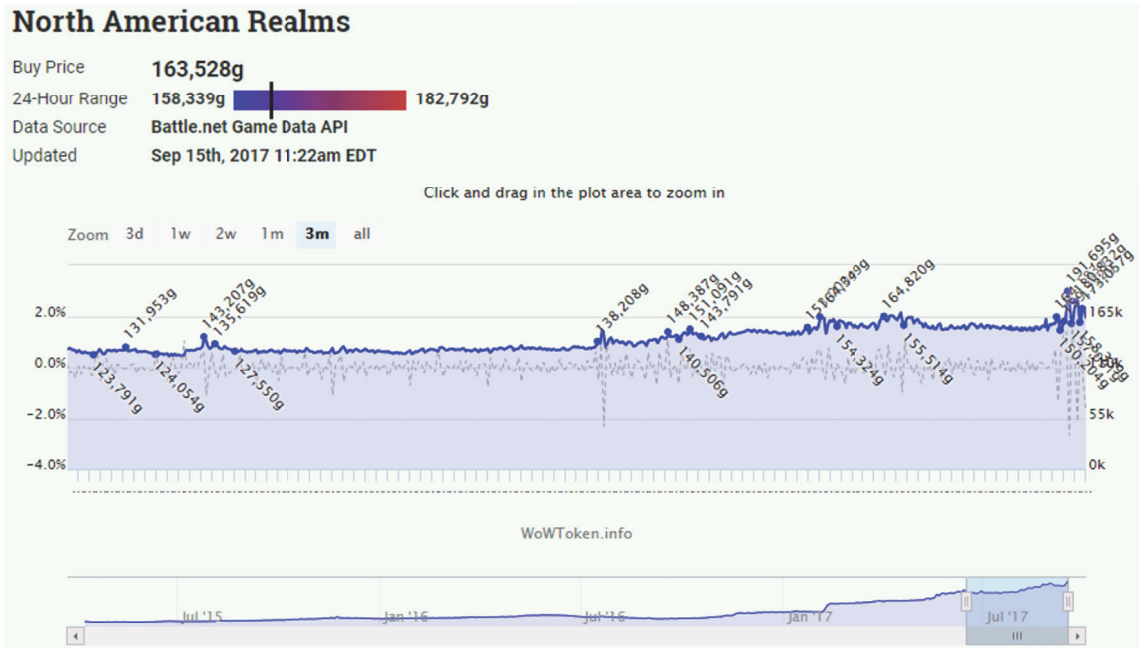


Figure 1: Token/Gold exchange rate example (from http://wowtoken.info).

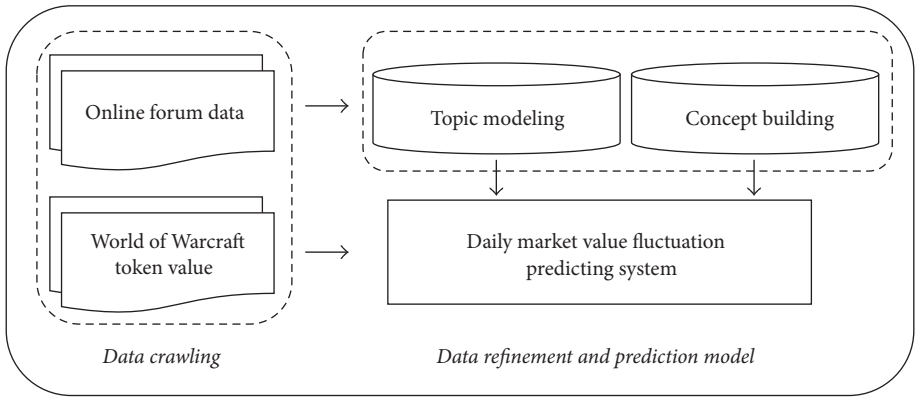


Figure 2: Overview of the proposed approach.

and reply to it were posted, the number of replies to each comment, and the number of views. Replies quoting previous comments and replies were crawled, excluding overlapping sentences. Each HTML page was crawled using Python regex to parse HTML tags and extract the number of topics, the number of replies, the dates on which the topics and replies were posted, and the URL of each topic from the general discussion boards. Based on the URLs of extracted topics and content, the replies to them were also extracted. The data were saved in json format, which was in turn converted to other formats (e.g., csv and xlsx) for different purposes. We collected data over a period of 460 days (April 23, 2015 to July 25, 2016). Within this period, the total number of topics collected was 166,651 (140,831 in the NA region and 25,820 in the EU region) and that of user replies was 2,931,748 (2,587,001 in the NA region and 344,747 in the EU region). We collected data in a manner that complied with the stipulated terms and conditions of use of the forums. The collected data did not include any personal information.

WoW players use real currencies (e.g., USD and EUR) to purchase tokens and sell them for the virtual currency—Gold—in the auction house within the gaming environment. Tokens have only been in use for a short time; moreover, websites and tools continue to trace and show the value of the tokens. We employed one of these sites to crawl the daily closing price of the token. The use of such data does not violate WoW's Terms of Use agreement or include users' personal or identifying information.

4. Analysis of User Opinion Data

Our goal is to build lexicons of concepts, that is, meaningful keywords for predicting the price of the currency used in WoW from the crawled data. To this end, we initially ran

topic modeling using the entire user comments to extract the representative keywords, the subset of which will then be used for building our initial lexicons. Afterwards, we retrieved relevant keywords to these initial keywords that we selected based on the similarity measure via a kernel density estimation technique. The main idea behind the kernel density estimation is to compute the similarity score of a given word to each keyword in our lexicon using a Gaussian kernel function in a word embedding space and take the average value of these similarity scores. Those highly relevant keywords would have high similarity values computed in this manner. For more details, refer to the subsequent section.

4.1. Data Preprocessing. To remove noise or unnecessary information, we applied several preprocessing strategies for all user comments. First, we removed the URL string, stop words such as auxiliary verbs, prepositions, and special characters. Then, we tokenized the strings into words and lemmatized each word. In addition, we used only words with a frequency higher than $f_{\min}$, to exclude words used very sparsely. In this study, we set $f_{\min}$ as 3.

4.2. Concept Building. Next, we built a lexicon representing a concept for our own purpose. The lexicon-based document analysis plays an important role in document analysis in various fields, such as economics, politics, and social sciences.

Our main goal is to find significantly relevant concepts existing in the entire document corpus by analyzing user comments. For example, a document in the WoW forum can be composed of concepts such as an item, a raid, and game contents. These concepts can be explained by those words corresponding to their lexicons, respectively. Hereby, we can quantitatively score how relevant a document is to each concept by analyzing how strongly the signal of the concept appears in it.

The lexicon building of a concept comprises two steps: (1) generation of a candidate word set that is potentially relevant to the concept and (2) manual refinement of it to finalize the lexicon. To extract candidate words for each concept, we constructed topic modeling with user comments. Topic modeling will be discussed later. Once representative keywords are generated for each topic in this manner, we used a union set of keywords from each topic as our candidate word set for concepts. Then, based on the prior knowledge of the features of virtual world [36–38] targeted by the authors who had a major related to game and the developers in the game companies making games in the MMOG genre, we selected words from candidate words and categorized them as some suitable lexicon for our desired concepts; each word is allowed to be assigned to multiple concepts. Thus, we can collect words that can explain the corresponding concept. Table 1 shows examples of these lexicons generated by this process.

4.3. Topic Modeling for Initial Lexicon Building. The topic modeling approach we used to extract representative keywords from a document corpus is nonnegative matrix factorization (NMF) [39], where the nonnegative constraint gives the interpretability on the resulting weights from factor matrices as the relevance score of a word or a document to each topic.

Table 1: Concept building example created and used in this study.

Concepts	Keywords
General	Game, like, people, time, player, blizzard, play, think, server, ...
Patch	Legion, new, game, change, going, spec, patch, expansion, ...
Raid	Get, gear, raid, dungeon, time, heroic, group, guild, ...
Movie	Movie, Warcraft, legion, time, see, story, good, well, ...
Leveling	Player, level, quest, character, new, ability, reward, ...
PVP (player versus player)	Click, gladiator, game, time, gear, pvp, play, damage, ...
Gear	Get, cloak, glove, gear, belt, robe, damage, gauntlet, ...

In detail, given a term-document matrix $A \in \mathfrak{R}^{m\times n}$, where m and n represent the size of the vocabulary and the number of documents, respectively, we normalized each column of this matrix to have a unit L2-norm. Given this matrix, NMF approximately factorizes it into two matrices $W \in \mathfrak{R}^{m\times d}$ and $H \in \mathfrak{R}^{n\times d}$, where d represents the number of topics, that is,

$$\min_{W,H} \left\| A - WH^T \right\|_F, \quad \text{where } W \geq 0,\ H \geq 0. \tag{1}$$

In this equation, the subscript F indicates the Frobenius norm, for example, $\|A\|_F = (\sum_{i=1}^{m}\sum_{j=1}^{n}|a_{ij}|^2)^{1/2}$ where $A \in \mathfrak{R}^{m\times n}$. NMF has nonnegative constraints, $W \geq 0$, $H \geq 0$, as shown in (1), which makes the element in W and H nonnegative, thus maintaining the interpretability. The columns in the resulting matrix, W, correspond to different topics, and the keywords correspond to the indices of the k largest value in each column function as the representative keywords of the topic.

In our analysis, we constructed a term-document matrix, A, from 140,831 threads in the US and 25,820 threads in the EU collected from the WoW forum. Each article has content and date features, such that we can calculate the scores per day by summing up the frequency of each word in sentences or articles generated in the corresponding day. In topic modeling, we set the number of topics, d, as 10. We also set the number of representative keywords for each topic, k, as 30.

4.4. Lexicon Expansion of Concepts and Relevance Score Calculation. In this subsection, we will describe details of how to expand concept keywords using kernel density estimation and to calculate the relevance scores with these concept keywords. Owing to the lack of expression resulting from the limited number of keywords a person could manage and the difficulty in determining the relevance to a concept in a user's

mind, we utilized kernel density estimation (KDE), a statistical metric to estimate the probability density function, which can be smoothened using multiple kernels, to infer what other words' concepts are. In other words, we selected some keywords for each concept, but the problem is that the number of keywords for each concept is relatively small compared to the number of the total vocabulary. To overcome this problem, we trained word embedding vector representations for all words using Word2Vec [40], which can provide semantically and syntactically meaningful vector representations. Afterwards, we calculated all distances between concept keywords and other words. Then, we computed the conditional probabilities or distributions for all words given each concept by utilizing KDE.

In particular, we adopted a Gaussian kernel adopts as follows [41]. For concept c, the conditional probability can be calculated by the distance function, d, which represents the distance between embedding vectors of words and vectors of word set in each concept, and the kernel, K, which ensures proper balance between the given word and others. The conditional probability of the embedding vector of the keyword, z, for a concept, c_i, that contains the embedding vector of concept keywords, $x_j \in c_i$, can be computed as follows:

$$p(z \mid c = c_i) = \frac{1}{\#(x \in c_i)} \sum_{x_j \in c_i} K\left(d\left(x_j, z\right)\right). \tag{2}$$

Here, we used Euclidean distance and Gaussian kernel and these equations are as follows.

$$d(x, y) = \|x - y\|_2 = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$
$$K(d) = \exp\left(-\frac{d^2}{\gamma}\right), \quad \text{where } \gamma = 2\sigma^2 \tag{3}$$

and we set the parameter, γ, as 1.

The conditional probability can also be considered as the relevance score to each class. Therefore, the score of the given sentence, s, for concept, c_i, is defined as

$$\text{score}_i(s) = \sum_{w_t \in s} p(w_t \mid c = c_i). \tag{4}$$

The range of the score is $[0, \infty)$ because the score of the sentence sums up all the scores of the words in the sentence, and all the probabilities are nonnegative. In practice, the sentence is not very long, and the score of each word is less than 1; therefore, the score of the sentence is not very large.

After scoring all the comments, we calculated scores for each day by summing up the articles generated in that day. Thus, we can obtain the scores for each day. Through this analysis, we found that the scores were rising in August 2015 and July 2016 when a new expansion pack and patch to change virtual world greatly was announced. Figure 3 shows the line graph of each score in each concept, as presented in Table 1 in some periods.

5. Causality Analysis

A Granger causality test [42] was conducted to assess the relation between standardized token prices and scores of each concept. The Granger causality test is based on the assumption that if variable X causes Y, then changes in X will consistently occur before changes in Y [1, 9]. We did not seek to test for actual causation but rather whether the time series of scores of each concept contained some predictive information regarding token prices.

Our time series for the selling price of tokens, denoted S_t, reflects daily changes in the price of tokens. We tested whether the time series of the collected data can predict changes in the token price by comparing the variance explained by two linear models. The first model uses only n lagged values of S_t (i.e., $S_{t-1}, \ldots, S_{t-n}$) for prediction, while the second model uses n lagged values of both S_t and the time series of scores of each concept, denoted by $X_{t-1}, \ldots, X_{t-n}$. We conducted the Granger causality test according to the models described in

$$S_t = \alpha + \sum_{i=1}^{n} \beta_i S_{t-i} + \epsilon_t$$
$$S_t = \alpha + \sum_{i=1}^{n} \beta_i S_{t-i} + \sum_{i=1}^{n} \gamma_i X_{t-i} + \epsilon_t. \tag{5}$$

Based on the results of the Granger causality tests, we rejected the null hypothesis that the time series of scores of each concept does not predict token prices—that is, $\beta \neq 0$—with a high level of confidence. By analyzing their relevance, we extracted the concepts with the highest Granger causality relations (p value < 0.05).

6. Configuration of the Prediction Model

Using the collected data and the analyzed and rated comment data, we built machine learning models for predicting the fluctuation in the WoW token price using gradient boosting trees, random forest, and support vector machines (SVM), which are widely used in binary classification problems, used to investigate sensitivity to different machine learning algorithms. We used R packages called *xgboost* and *cforest* for the implementation in gradient boosting and random forest. The implementation of SVM was based on libSVM [43] with the radial basis function (RBF) kernel. Using LibSVM, the learning data was cross-validated to search for optimum parameters for the RBF kernel. We created a setup to apply machine learning to data spanning over a period of 460 days.

As the first step, we standardized the data to improve its applicability to the learning model. The z-score, $Z_{\mathbb{E}_t} = (\mathbb{E} - \overline{x}(\mathbb{E}))/\sigma(\mathbb{E})$, where $\overline{x}(\mathbb{E})$ and $\sigma(\mathbb{E})$ represent the mean and the standard deviation for every date, respectively, of data for the previous 12 days ($t = 12$) was used. An example of applicable input data is shown in Table 2. As for the input nodes, based on the input data provided in Table 2, 10 input data points were represented as serial vectors to allocate neurons based on the cumulative number of days spent on learning; that is,

Figure 3: Scores of each concept for days.

Table 2: Example of a machine learning data set.

Data class	Date	KDE-based concept scoring data							Formal data		
		concept							Number of topics	Sum of replies	Sum of views
		General	PVP	Raid	Patch	Movie	Leveling	Gear			
Crawled and analyzed data	May 1, 2016	A	B	C	D	E	F	G	V	W	X
Input learning data	May 1, 2016	Z_{A_t}	Z_{B_t}	Z_{C_t}	Z_{D_t}	Z_{E_t}	Z_{F_t}	Z_{G_t}	Z_{V_t}	Z_{W_t}	Z_{X_t}

Table 3: Statistical significance (p values) of bivariate Granger causality test between WoW token price (NA Region) and concepts of forum opinions.

Time lag	WoW token price								
	General	PVP	Raid	Patch	Movie	Leveling	Gear	Number of topics	Number of replies
1 day	**0.0116**	**0.0032**	**0.0089**	**0.0074**	**0.0052**	**0.0073**	**0.0034**	**0.0016**	**0.0018**
2 days	**0.0286**	**0.0059**	**0.0138**	**0.0243**	**0.0144**	**0.0131**	**0.0049**	**0.0073**	**0.0169**
3 days	**0.0181**	**0.0035**	**0.0081**	**0.0091**	**0.0177**	**0.0075**	**0.0156**	**0.0163**	**0.0252**
4 days	**0.0494**	**0.0088**	**0.0163**	**0.0315**	**0.0174**	**0.0161**	**0.0332**	**0.0246**	**0.0391**
5 days	0.0886	**0.0138**	0.0603	**0.0363**	0.0657	**0.0346**	**0.0221**	**0.0286**	0.0683
6 days	0.1245	**0.0476**	0.0813	0.121	0.1109	0.0676	**0.0061**	**0.0341**	0.1244
7 days	0.2668	0.1325	0.2188	0.3087	0.2317	0.2106	**0.0322**	0.0665	0.4006

Table 4: Statistical significance (p values) of bivariate Granger causality test between WoW token price (EU Region) and concepts of forum opinions.

Time Lag	WoW token price								
	General	PVP	Raid	Patch	Movie	Leveling	Gear	Number of topics	Number of replies
1 day	0.4323	0.1714	0.4791	0.1193	0.1806	0.3695	0.1352	**0.0026**	0.2684
2 days	0.5641	0.2162	0.5267	0.1496	0.2411	0.4839	0.1117	**0.0031**	0.1178
3 days	0.1518	**0.0499**	0.1643	0.0647	0.0812	0.1301	0.0515	**0.0107**	0.1542
4 days	0.0924	**0.011**	0.1257	**0.0148**	**0.0214**	0.0722	**0.0132**	**0.0255**	0.2404
5 days	0.1384	**0.0176**	0.2236	**0.0303**	0.0524	0.1185	**0.0067**	**0.0221**	0.2501
6 days	0.2431	**0.0427**	0.3235	0.061	0.0896	0.2303	**0.0114**	**0.0143**	0.2077
7 days	0.2766	**0.0391**	0.2266	**0.0337**	0.114	0.2278	**0.0062**	**0.0076**	0.1791

20, 30, 50, and 700 neurons were allocated to cumulative 2, 3, 5, and 7 days.

7. Experimental Results

7.1. Results of Granger Causality Test. The Granger causality test was conducted on the Bitcoin transaction count and the price for time lag of 1 to 7 days. The time lag was omitted because it produced less significant results after 8 days. Tables 3 and 4 list the test results.

From the results, it was observed that in the NA region, most user opinions had an influence when the time lag was small and user opinions related to gear and PVP had causation when the time lag became larger. In the EU region, it was observed that user opinions related to gear, PVP, and patches had causation. This process was only used for verification. The entire data set was used to build the actual learning model for prediction.

7.2. Prediction Results. We built and applied the machine learning model based on the gathered and scoring data to predict the daily fluctuation of the WoW token price. From April 23, 2015, to July 25, 2016, tests are conducted with 10-fold cross validation. The accuracy rate, Matthews correlation coefficient (MCC), and F-measure were used to evaluate the performance of the proposed model.

Table 5 presents the prediction results. The most accurate prediction model for the WoW token price in the NA region (accuracy rate = 82.55%) is based on the gradient boosting and previous seven days' learning data. The most accurate prediction model for the WoW token price in the EU region (accuracy rate = 81.52%) is based on the gradient boosting and the previous twelve days' learning data. Table 5 presents the results relative to the different machine learning models and learning data structures. Cumulative learning data for seven days or longer resulted in negligible difference, and cumulative learning data for less than five days proved insufficient for learning and compromised the prediction accuracy.

8. Discussion and Conclusion

In this paper, we presented a method for predicting the value fluctuation of the virtual currency in an MMOG, a

Table 5: Experimental results of predicted WoW token prices fluctuation.

Learning method		Gradient boosting			Random forest			Support vector machine		
Region	Learning days	Accuracy (%)	*F*1-score	MCC	Accuracy (%)	*F*1-score	MCC	Accuracy (%)	*F*1-score	MCC
North America region	3 days	75.65%	0.7576	0.5731	69.13%	0.6984	0.4799	64.13%	0.6432	0.3445
	5 days	79.56%	0.7955	0.5921	76.08%	0.7784	0.5704	74.48%	0.7539	0.318
	7 days	**82.55%**	**0.8255**	**0.6372**	78.26%	0.793	0.5768	76.95%	0.7765	0.3106
	12 days	80.65%	0.8045	0.6009	78.69%	0.7858	0.579	78.69%	0.7996	0.3773
Europe region	3 days	69.56%	0.6957	0.3917	71.52%	0.7138	0.4375	75.43%	0.7966	−0.0219
	5 days	71.73%	0.7174	0.4346	73.91%	0.7367	0.4907	70.65%	0.6225	0.2742
	7 days	77.17%	0.7718	0.5429	77.39%	0.77	0.5723	71.3%	0.6346	0.2989
	12 days	**81.52%**	**0.815**	**0.6322**	77.6%	0.7791	0.5779	73.91%	0.6656	0.3416

subject that had previously been minimally investigated. Our results demonstrated that the value fluctuation of the virtual currency in the WoW MMOG environment can be predicted. This paper demonstrated that the proposed method could be applied to virtual currency selling/buying.

The results of refined user data were used in the predictions, which revealed that users' opinions could be effectively used in currency value fluctuation predictions. Granger causality test results showed that users' opinions affect the token value, regardless of the region of the world.

The proposed method provides developers with market price trends and enables them to determine the numerical values and probabilities of multiple effects that would otherwise be difficult to identify. These include users' economic activities in MMOG environments and the relevance of currency value fluctuations, which are conducive to adjusting the overall balance in the MMOG and thus improving the environment. Validated currency value fluctuation predictions would enable users to pursue profits, in a narrower sense, and to perceive the overall flows of a given virtual currency in a broader sense.

The proposed prediction system could be improved by taking the following aspects into consideration. First, a more sophisticated user characterization could yield more revealing findings. For example, a considerable number of reasons exist for users playing games [44], and these reasons are likely to be relevant to their transaction activities. An analysis of user motives and their relation to transaction activities could improve the understanding of economic activities in MMOG settings. In addition, data on factors, such as changes in the gaming environment, virtual currency system updates, and user comments, including chats, are worth analyzing to thoroughly understand the MMOG setting and increase prediction accuracy. Furthermore, if a sufficiently long time period of data could be gathered, multivariate time series analysis will ensure competitive results. Our plan is to enhance our prediction system in future research to incorporate these considerations with improved reliability and efficiency of the proposed method.

Conflicts of Interest

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

Acknowledgments

This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning (NRF-2015R1A1A1A05001196, NRF-2016R1E1A2A02946052, and NRF-2017R1A2B2005380), by an Institute for Information & Communications Technology Promotion (IITP) grant funded by the Korean government (MSIP; 2016-0-00285, High performance computing [HPC] based rendering solution development), and by Linewalks Inc.

References

[1] Y. B. Kim, S. H. Lee, S. J. Kang, M. J. Choi, J. Lee, and C. H. Kim, "Virtual world currency value fluctuation prediction system based on user sentiment analysis," *PLoS ONE*, vol. 10, no. 8, Article ID e0132944, 2015.

[2] D. H. Shin, "Understanding purchasing behaviors in a virtual economy: Consumer behavior involving virtual currency in Web 2.0 communities," *Interacting with Computers*, vol. 20, no. 4-5, pp. 433–446, 2008.

[3] Y. Guo and S. Barnes, "Why people buy virtual items in virtual worlds with real money," *ACM SIGMIS Database*, vol. 38, no. 4, pp. 69–76, 2007.

[4] S. Papagiannidis, M. Bourlakis, and F. Li, "Making real money in virtual worlds: MMORPGs and emerging business opportunities, challenges and ethical implications in metaverses," *Technological Forecasting & Social Change*, vol. 75, no. 5, pp. 610–622, 2008.

[5] S. Scarle, S. Arnab, I. Dunwell, P. Petridis, A. Protopsaltis, and S. de Freitas, "E-commerce transactions in a virtual environment: Virtual transactions," *Electronic Commerce Research*, vol. 12, no. 3, pp. 379–407, 2012.

[6] V. Lehdonvirta, "Virtual item sales as a revenue model: Identifying attributes that drive purchase decisions," *Electronic Commerce Research*, vol. 9, no. 1-2, pp. 97–113, 2009.

[7] D. Y. Wohn, "Spending real money: Purchasing patterns of virtual goods in an online social game," in *Proceedings of the 32nd Annual ACM Conference on Human Factors in Computing Systems, CHI 2014*, pp. 3359–3368, May 2014.

[8] T. Kimoto, K. Asakawa, M. Yoda, and M. Takeoka, "Stock market prediction system with modular neural networks," in *Proceedings of the 1990 International Joint Conference on Neural*

Networks (IJCNN '90), vol. 1, pp. 1–6, Washington, DC, USA, June 1990.

[9] 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.

[10] I. Bordino, S. Battiston, G. Caldarelli, M. Cristelli, A. Ukkonen, and I. Weber, "Web search queries can predict stock market volumes," *PLoS ONE*, vol. 7, no. 7, Article ID e40014, 2012.

[11] Y. Cohen-Charash, C. A. Scherbaum, J. D. Kammeyer-Mueller, and B. M. Staw, "Mood and the market: can press reports of investors' mood predict stock prices?" *PLoS ONE*, vol. 8, no. 8, Article ID e72031, 2013.

[12] R. P. Schumaker and H. Chen, "Textual analysis of stock market prediction using breaking financial news: The AZFin text system," *ACM Transactions on Information and System Security*, vol. 27, no. 2, article no. a12, 2009.

[13] Z. Yudong and W. Lenan, "Stock market prediction of S&P 500 via combination of improved BCO approach and BP neural network," *Expert Systems with Applications*, vol. 36, no. 5, pp. 8849–8854, 2009.

[14] E. Kosminsky, "World of Warcraft: the viability of massively multiplayer online role-playing games as platforms for modeling and evaluating perfect competition," *Journal For Virtual Worlds Research*, vol. 2, 2009.

[15] Pardus Massive Multiplayer Online Browser Game, https://www.pardus.at/.

[16] Y. B. Kim, N. Park, Q. Zhang, J. G. Kim, S. J. Kang, and C. H. Kim, "Predicting virtual world user population fluctuations with deep learning," *PLoS ONE*, vol. 11, no. 12, Article ID e0167153, 2016.

[17] O. Mryglod, B. Fuchs, M. Szell, Y. Holovatch, and S. Thurner, "Interevent time distributions of human multi-level activity in a virtual world," *Physica A: Statistical Mechanics and its Applications*, vol. 419, pp. 681–690, 2015.

[18] M. Szell, R. Lambiotte, and S. Thurner, "Multirelational organization of large-scale social networks in an online world," *Proceedings of the National Acadamy of Sciences of the United States of America*, vol. 107, no. 31, pp. 13636–13641, 2010.

[19] M. Szell, R. Sinatra, G. Petri, S. Thurner, and V. Latora, "Understanding mobility in a social petri dish," *Scientific Reports*, vol. 2, article no. 457, 2012.

[20] M. Szell and S. Thurner, "Measuring social dynamics in a massive multiplayer online game," *Social Networks*, vol. 32, no. 4, pp. 313–329, 2010.

[21] M. Szell and S. Thurner, "Social dynamics in a large-scale online game," *Advances in Complex Systems (ACS)*, vol. 15, no. 6, Article ID 1250064, 2012.

[22] S. Thurner, M. Szell, and R. Sinatra, "Emergence of good conduct, scaling and zipf laws in human behavioral sequences in an online world," *PLoS ONE*, vol. 7, no. 1, Article ID e29796, 2012.

[23] W. S. Bainbridge, "The scientific research potential of virtual worlds," *Science*, vol. 317, no. 5837, pp. 472–476, 2007.

[24] P. R. Messinger, E. Stroulia, and K. Lyons, "Virtual worlds—past, present, and future: new directions in social computing," *Decision Support Systems*, vol. 47, no. 3, pp. 204–228, 2009.

[25] B. Corominas-Murtra, B. Fuchs, and S. Thurner, "Detection of the elite structure in a virtual multiplex social system by means of a generalised K-core," *PLoS ONE*, vol. 9, no. 12, Article ID e112606, 2014.

[26] B. Fuchs and S. Thurner, "Behavioral and network origins of wealth inequality: Insights from a virtual world," *PLoS ONE*, vol. 9, no. 8, Article ID e103503, 2014.

[27] S.-J. Kang, Y. B. Kim, T. Park, and C.-H. Kim, "Automatic player behavior analysis system using trajectory data in a massive multiplayer online game," *Multimedia Tools and Applications*, vol. 66, no. 3, pp. 383–404, 2013.

[28] P. Klimek and S. Thurner, "Triadic closure dynamics drives scaling laws in social multiplex networks," *New Journal of Physics*, vol. 15, article 063008, 2013.

[29] M. Szell and S. Thurner, "How women organize social networks different from men," *Scientific Reports*, vol. 3, article no. 1214, 2013.

[30] W.-J. Xie, M.-X. Li, Z.-Q. Jiang, and W.-X. Zhou, "Triadic motifs in the dependence networks of virtual societies," *Scientific Reports*, vol. 4, article no. 5244, 2014.

[31] Y. S. Hau and Y.-G. Kim, "Why would online gamers share their innovation-conducive knowledge in the online game user community? Integrating individual motivations and social capital perspectives," *Computers in Human Behavior*, vol. 27, no. 2, pp. 956–970, 2011.

[32] P. Panzarasa, T. Opsahl, and K. M. Carley, "Patterns and dynamics of users' behavior and interaction: Network analysis of an online community," *Journal of the Association for Information Science and Technology*, vol. 60, no. 5, pp. 911–932, 2009.

[33] C. C. Sing and M. S. Khine, "An analysis of interaction and participation patterns in online community," *Journal of Educational Technologyand Society*, vol. 9, article 250, 2006.

[34] Y. B. Kim, J. G. Kim, W. Kim et al., "Predicting fluctuations in cryptocurrency transactions based on user comments and replies," *PLoS ONE*, vol. 11, no. 8, Article ID e0161197, 2016.

[35] Y. B. Kim, J. Lee, N. Park et al., "When Bitcoin encounters information in an online forum: Using text mining to analyse user opinions and predict value fluctuation," *PLoS ONE*, vol. 12, no. 5, p. e0177630, 2017.

[36] V. H.-H. Chen, H. B.-L. Duh, and H. Renyi, "The Changing Dynamic of Social Interaction in World of Warcraft: The Impacts of Game Feature Change," in *Proceedings of the 2008 International Conference on Advances in Computer Entertainment Technology, ACE 2008*, pp. 356–359, December 2008.

[37] A. Golub, "Being in the World (of Warcraft): Raiding, realism, and knowledge production in a massively multiplayer online game," *Anthropological Quarterly*, vol. 83, no. 1, pp. 17–46, 2010.

[38] C. A. Paul, "Welfare epics? The rhetoric of rewards in world of warcraft," *Games and Culture*, vol. 5, no. 2, pp. 158–176, 2010.

[39] J. Choo, C. Lee, C. K. Reddy, and H. Park, "UTOPIAN: User-driven topic modeling based on interactive nonnegative matrix factorization," *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 12, pp. 1992–2001, 2013.

[40] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," arXiv preprint https://arxiv.org/abs/1301.3781.

[41] D. Park, S. Kim, J. Lee, J. Choo, N. Diakopoulos, and N. Elmqvist, "ConceptVector: text visual analytics via interactive lexicon building using word embedding," *IEEE Transactions on Visualization and Computer Graphics*, 2017.

[42] C. Hiemstra and J. D. Jones, "Testing for linear and nonlinear Granger causality in the stock price-volume relation," *The Journal of Finance*, vol. 49, no. 5, pp. 1639–1664, 1994.

[43] C. Chang and C. Lin, "LIBSVM: a Library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 3, article 27, 2011.

[44] J. Billieux, M. Van der Linden, S. Achab et al., "Why do you play World of Warcraft? An in-depth exploration of self-reported motivations to play online and in-game behaviours in the virtual world of Azeroth," *Computers in Human Behavior*, vol. 29, no. 1, pp. 103–109, 2013.

Hindawi
Complexity
Volume 2017, Article ID 3719428, 10 pages
https://doi.org/10.1155/2017/3719428

Research Article

Social Network Community Detection Using Agglomerative Spectral Clustering

Ulzii-Utas Narantsatsralt and Sanggil Kang

Department of Computer Engineering, Inha University, Incheon, Republic of Korea

Correspondence should be addressed to Sanggil Kang; sgkang@inha.ac.kr

Received 18 April 2017; Revised 24 July 2017; Accepted 23 August 2017; Published 7 November 2017

Academic Editor: Katarzyna Musial

Community detection has become an increasingly popular tool for analyzing and researching complex networks. Many methods have been proposed for accurate community detection, and one of them is spectral clustering. Most spectral clustering algorithms have been implemented on artificial networks, and accuracy of the community detection is still unsatisfactory. Therefore, this paper proposes an agglomerative spectral clustering method with conductance and edge weights. In this method, the most similar nodes are agglomerated based on eigenvector space and edge weights. In addition, the conductance is used to identify densely connected clusters while agglomerating. The proposed method shows improved performance in related works and proves to be efficient for real life complex networks from experiments.

1. Introduction

In recent years, community detection in a network has become one of the main topics of fields, such as biology, computer science, physics, and applied mathematics [1–3]. In a network, $G(V, E)$, where V is a set of nodes and E is the edges (relation) between the nodes, a community is a group of nodes with tightly connected edges with each other and the nodes of community show similar characteristics. For example, in social network, people in a community show similar interest to a trend in a community, for example, buying the same products in online marketing. In a biology network, proteins in a community show similar specific functions, and, in the World Wide Web, sites clustered together show the same topic in their web page. Scientists in many fields made significant contributions to detecting communities by a number of different methods, such as graph partitioning [4, 5], hierarchical clustering [6, 7], and spectral clustering [8, 9].

In graph partitioning, a network is divided into clusters in such a way that the number of edges connecting the clusters is minimum, that is, the edges of a cluster are denser inside than outside (also referred as conductance [10]). In addition, the number of lowest sized clusters needs to be specified. Girvan and Newman [3] introduced a popular graph partitioning algorithm. Girvan and Newman [3] use modularity (also referred as conductance) to cluster communities but the method is slower than other community detection algorithms [11, 12]. Later, Djidjev [13] proposed a computationally faster version of the algorithm. However, the definition of conductance is not always definite, and the definition can be false in some cases [5]. Therefore, graph partitioning still needs further inference. A number of methods have been proposed to solve this problem. One of the famous methods is introduced by Newman [5]. They use spectral clustering algorithm with modularity maximization, in which the modularity function is implemented for only possible clusters of network and the result proved that spectral clustering with conductance can efficiently cluster communities.

Hierarchical clustering is used for complex networks because they often have a hierarchical structure [14]. Hierarchical clustering consists of a division [15] and agglomeration stage [16, 17]. In the division stage, a network is

deemed to be one cluster in the beginning and the network is then divided into clusters in each iteration, where the most dissimilar nodes are separated. In the agglomeration stage, similar nodes are agglomerated together until the termination criteria are met or the clusters agglomerate into a single community. However, hierarchical clustering needs a well-defined similarity function and the clustering can be inaccurate if all nodes are similar to each other.

However, the problem of similar nodes and similarity function can be solved by projecting the nodes into high dimensional feature space using spectral clustering because the projected eigenvectors significantly distinguish the similar nodes into more distanced positions in feature space. The reason for using eigenvector space instead of using original point is that the properties of original clustering are made more distinct by the eigenvector space. In spectral clustering, original points are transformed into a set of points in eigenvector space and clustering is done by analyzing eigenvector space. One technique for clustering eigenvector space is to use k-means algorithm [18] where similar nodes are clustered together. However, traditional spectral clustering has a problem with model selection which depends on heuristics. The problem can be solved using weighted kernel spectral clustering (KSC) with primal and dual representations [19–21]. KSC [21] focuses on the principle that projections of similar nodes are clustered together in eigenvector space. In another work of KSC, an agglomeration technique is introduced to the KSC which is called agglomerative hierarchical kernel spectral clustering (AH-KSC) [22]. AH-KSC uses eigenvector space to find distance between nodes and it agglomerates close distanced nodes. The main purpose of AH-KSC is to get hierarchical clustering but accuracy of AH-KSC does not improve significantly from KSC because AH-KSC allows indirectly connected nodes to be agglomerated together and also there are no termination criteria for satisfied community. The problems of KSC and AH-KSC are choosing eigenvector, improving accuracy of detected communities, and using only data generated by hand which usually do not show same characteristics as real life networks.

The above-mentioned methods focus on decreasing the computation time or improving the accuracy of community detection. Methods for improving the computational time have been well studied and it can be solved by advances in technology and techniques [23–25], such as parallel computing and GPU programming. Improving the accuracy of community detection has been challenging task because networks are usually structured with great complexity with millions of nodes and edges. Hence, this paper proposes an agglomerative spectral clustering method with conductance and edge weights to improve the accuracy of community detection. The characteristics of the proposed method are well suited for accurate community detection in complex networks because the eigenvector space from spectral clustering provides well distinct points that are used for the similarity function in agglomeration. The conductance is used for the sensitive termination criteria of agglomeration and the edge weight is a major factor for evaluating a more accurate similarity. In addition, performance of the proposed method was compared with that of AH-KSC and KSC using real life social network data with a ground-truth, which are the LiveJournal and Orkut network. This method can help improve the community detection performance from previous works [21, 22].

The remainder of this paper is as follows: Section 2 introduces the problem statement and background, which helps understand the proposed method. The core algorithm of the proposed method is explained in Section 3. The experiment is outlined in Section 4 and Section 5 reports the conclusions.

2. Fundamental Concepts

2.1. Problem Statement. In KSC, the data are divided into training, validating, and test sets. In the training stage, the eigenvector space of the training data is signed, which is used for clustering the nodes in a network. The sign of the eigenvector points in the same cluster is identical. In the validating stage, model selection is performed to identify the clustering parameters. The eigenvector space of the test data is used to evaluate the clusters obtained from the training data using the hamming distance function. The problem with the KSC is that clustering depends on encoding/decoding eigenvectors space. The encoded values are all signed in KSC [21] and distinction between the two elements of the eigenvector is only "1" or "0" so that similar eigenvector points become noisy. For example, if in eigenvector space, $e_1^{(l)} = 0.001$ and $e_2^{(l)} = -0.001$, the two values can be binarized as "1" and "0," respectively. Although, $e_1^{(l)}$ and $e_2^{(l)}$ are projected in similar feature space, the results of encoding show a different outcome. This problem can be solved by agglomerative hierarchical KSC (AH-KSC). In AH-KSC [22], instead of signing eigenvector space, the space is used as the data points to obtain the distance between nodes in a network and close distanced nodes are agglomerated together until there are only k clusters or less.

KSC and AH-KSC still have certain disadvantages. Both methods calculate the kernel matrix Ω by counting the number of edges connecting the common neighbors between two nodes, $i \wedge j$: $\Omega_{ij} = \sum_{k,l \in N_{ij}} A_{kl}$, where N_{ij} is a set of common neighbors of $i \wedge j$, k and l are common neighbors, and A_{kl} is adjacency matrix of the graph. However, the common neighbors between nodes can cause indirectly connected nodes to be clustered together so that the nodes in different clusters can be clustered. To solve this problem, the adjacency matrix is used as a kernel matrix so agglomerated nodes can be connected directly. In addition, KSC and AH-KSC use only the first $k - 1$ eigenvectors for encoding/decoding but the remaining eigenvectors can still provide correlated information for clustering. To take this into consideration, in this study, all eigenvector space was used to evaluate the similarity between nodes. Furthermore, in AH-KSC, there were no termination criteria for agglomerating clusters. Therefore, the conductance was used as termination criterion during the agglomeration of satisfied clusters.

2.2. Background. In general, KSC is described by a primal-dual formulation. Given a network, $G(V, E)$, where V denotes the vertices and E the edges, and the training data $V_{\text{tr}} = \{x_{\text{tr}}\}_{i=1}^{N_{\text{tr}}}$, the primal problem [21] is

$$\min_{w^l, e^l, b_l} \frac{1}{2}\sum_{l=1}^{k-1} w^{(l)^t} w^{(l)} - \frac{1}{2}\sum_{l=1}^{k-1} \gamma_l e^{(l)^T} D_\Omega^{-1} e^l \quad \text{subject} \quad e^{(l)} = \Phi w^{(l)} + b_l 1_{N_{\text{tr}}}, \quad l = 1, \ldots, k-1, \tag{1}$$

where $e^{(l)} = [e_1^{(l)}, \ldots, e_{N_{\text{tr}}}^{(l)}]$ is the projection, which is the mapped points of training data in feature space with respect to the direction, $w^{(l)}$, l indicates the number of score variables, which is needed to encode the k clusters, D_Ω^{-1} is the inverse matrix of the degree matrix of the kernel matrix, Ω, Φ is the $N_{\text{tr}} \times d_h$ feature matrix, where $\Phi = [\varphi(x_1)^T; \ldots; \varphi(x_{N_{\text{tr}}})^T]$, γ_l is the regularization constant, and $1_{N_{\text{tr}}}$ is the $N_{\text{tr}} \times N_{\text{tr}}$ matrix of ones. The primal form of the data point is expressed as

$$e_i^{(l)} = w^{(l)^T} \varphi(x_i) + b_l, \quad i = 1, \ldots, N_{\text{tr}}, \tag{2}$$

where $\varphi : R^n \to R^{d_h}$ is the map to high dimensional feature space, where n is the number of nodes in graph, G, d_h is the number of eigenvectors, and b_l is the bias term. The dual problem related to the primal problem is

$$D_\Omega^{-1} M_D \Omega \alpha^l = \lambda_l \alpha^{(l)}, \tag{3}$$

where Ω is the kernel matrix with ijth entry, $\Omega_{ij} = K(x_i, x_j) = \varphi(x_i)^T \varphi(x_i)$; D_Ω is the diagonal matrix of Ω with elements of $d_{ii} = \sum_j \Omega_{ij}$; M_D is a centering matrix defined as $M_D = I_{N_{\text{tr}}} - (1/1_{N_{\text{tr}}}^T D_\Omega^{-1} 1_{N_{\text{tr}}})(1_{N_{\text{tr}}} 1_{N_{\text{tr}}}^T D_\Omega^{-1})$, where $I_{N_{\text{tr}}}$ is the $N_{\text{tr}} \times N_{\text{tr}}$ identity matrix; $\alpha^{(l)}$ is the dual variable; and the kernel function K is the similarity function of the graph. The parameters, such as the number of community k, are estimated using the training data, V_{tr}, and validating data, V_{va}. In addition, all the nodes are clustered in the training and validating stage. The eigenvector space is used to find unique code-word for all clusters A_p, $p = 1, \ldots, k$. The codebook, $C = \{c_p\}_{p=1}^k$, can be obtained from rows of the binarized eigenvector matrix. Finally, the eigenvector space of the test data, V_{te}, is decoded using the hamming distance [21] and the clustered result is evaluated. Therefore, eigenvector space is used to derive the similarity among nodes, which will be explained in detail in the following section.

3. Proposed Community Detection Algorithm

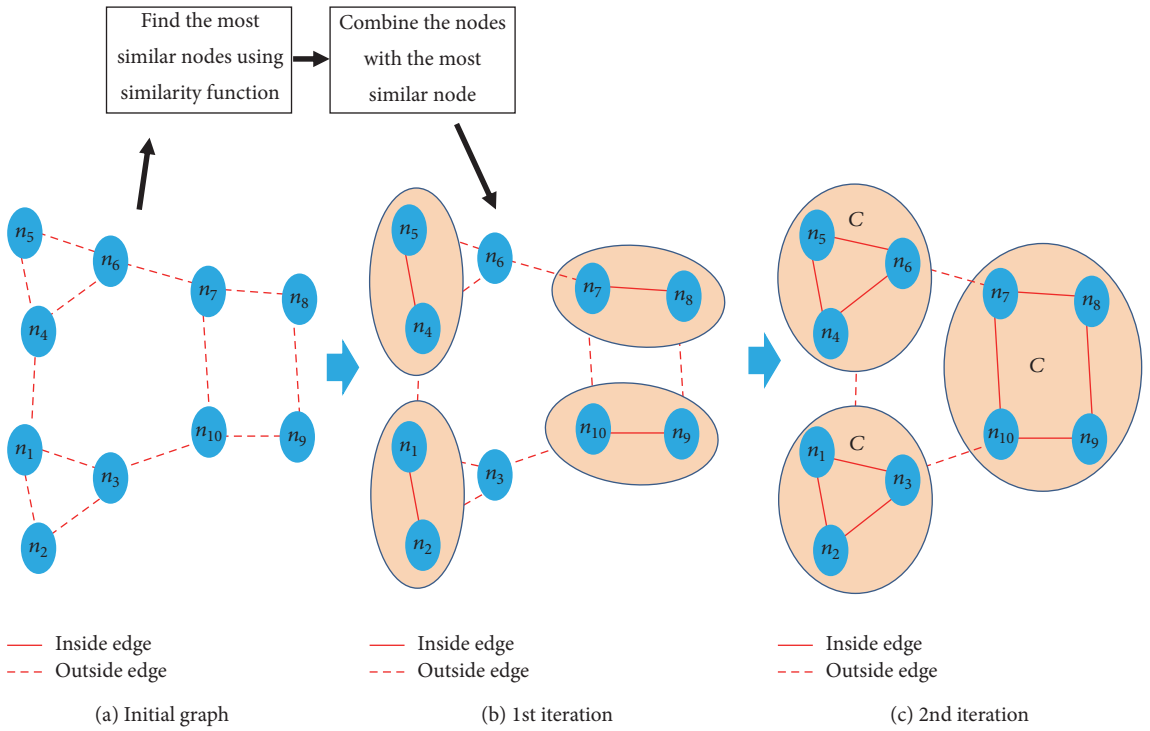
This section presents details of agglomerative spectral clustering with the conductivity method. The eigenvector space is used to find the similarity among nodes and agglomerate the most similar nodes to make a new combined node in a network graph. The new combined node is added to the graph after agglomeration and the changed graph is iterated until the termination criteria are satisfied.

To agglomerate two nodes, a similarity function is modified from the correlation distance function among the nodes as follows:

$$(i, j) = \text{CorDis} = 1 - p = 1 - \frac{\sum_{n=1}^N \left(x_n^i * x_n^j\right) - (1/N)\sum_{n=1}^N x_n^i \sum_{n=1}^N x_n^j}{\sqrt{\sum_{n=1}^N \left(x_n^i\right)^2 - (1/n)\left(\sum_{n=1}^N x_n^i\right)^2}\sqrt{\sum_{n=1}^N \left(x_n^j\right)^2 - (1/n)\left(\sum_{n=1}^N x_n^j\right)^2}}, \tag{4}$$

where (i, j) is the similarity between nodes i and j in the range of [0,2] with 0 being perfect similarity and 2 being perfect dissimilarity. x_n^i is the value of the eigenvector, $e_i^{(n)}$, in eigenvector space, ith row, and nth column.

The eigenvector space is not enough to fully express the similarity among agglomerated nodes because the nodes connected to each other are projected into a similar place in feature space and it is very difficult to distinguish similar projections. On the other hand, these similar projections can be distinguished using the disparity of the edge connections between the nodes. Agglomerated nodes can have more than one connecting edge with each other. Therefore, more tightly connected nodes have more similarity. For example, in Figure 1, similar nodes are combined in the 1st iteration and node n_6 has two connections to the agglomerated node of n_4 and n_5 and has one connection to the agglomerated node of n_7 and n_8, so that n_6 is more likely to agglomerate with n_5 and n_4. In the 2nd iteration, new agglomerated nodes are used to find new eigenvector space and agglomerate similar nodes. In doing so, the number of edges in the graph is unchanged and some nodes have more than one edge between them. As mentioned in the example of node n_6, the edges between two nodes are used as a mean to give a similarity score between nodes to improve the accuracy of the algorithm. On the other hand, the number of edges between nodes can be varied too much and the value of the similarity function in (4) is too different. Therefore, the similarity function will be overemphasized on a number of edges; that is, disregard the eigenvector space score. In the present study, a sigmoid function is used to normalize the edge values to solve the above-mentioned problem.



(a) Initial graph (b) 1st iteration (c) 2nd iteration

Figure 1: Example of agglomerating nodes.

Equation (4) is modified, as expressed in

$$(i, j) = \frac{\left(1 - \left(\left(\sum_{n=1}^{N}\left(x_n^i * x_n^j\right) - (1/N)\sum_{n=1}^{N} x_n^i \sum_{n=1}^{N} x_n^j\right) / \sqrt{\sum_{n=1}^{N}\left(x_n^i\right)^2 - (1/n)\left(\sum_{n=1}^{N} x_n^i\right)^2}\sqrt{\sum_{n=1}^{N}\left(x_n^j\right)^2 - (1/n)\left(\sum_{n=1}^{N} x_n^j\right)^2}\right)\right)}{0.5/\left(1 + e^{((-5/E_{\max}) * E(i,j))}\right)}, \quad (5)$$

where $E_{\max}$ is the maximum number of edges and $E(i, j)$ is the number of edges between nodes $i \wedge j$. The vertical value of the sigmoid graph is deemed to be the edge similarity score and starts from 0.5 to 1 while the horizontal value, which is the number of edges, ranges from 0 to the maximum number of edges. Equation (5) is used to find the most similar node of node i from the other nodes in graph G. At the first iteration, the first node becomes a candidate and if there is a more similar node than the candidate, the candidate is then replaced with it. The process continues until the similarity of all nodes is evaluated. Thus, the most similar node to node i is determined by

$$n_{\text{ms}}^i = \min_k (i, k), \quad (6)$$

where $k \in N$, N is all the nodes in graph G, and n_{ms}^i is the most similar node of node i.

Furthermore, to obtain a more accurate clustering result, this study considers the definition of a good community, which is "density of the edge connection should be higher inside than outside" [10]. The similar nodes are agglomerated together in every iteration and the agglomerated nodes become a clustered community after a few iterations. If the cluster is connected tightly inside and sparsely connected to outside, there is no need for further agglomeration because the cluster is sufficiently satisfied to be a good community and agglomeration for this community is terminated. In addition, two communities are agglomerated when they are tightly connected to each other. For example, in Figure 1(c), where the inside edges are straight lines and the outside edges are dotted lines, the graph is clustered into three agglomerated communities, such as C_1, C_2, and C_3. In the case of community C_1, it has three inside edges and two outside edges connected to both C_2 and C_3 so that C_1 has a denser connection inside than outside. Consequently, no further agglomeration is needed. To consider the ratio of the inside and outside edges into (5), the two possible cases are

divided when node j is a candidate as the most similar node for node i:

(1) $N_i < N_j$

(2) $N_i > N_j$

where N_i is the number of nodes inside node i and N_j is the number of nodes inside node j.

In the first case, the number of inside edges of node i is at most equal to the number of outside edges connecting to node j. However in the next case, the number of inside edges of node i is more than the number of outside edges connecting to node j. Therefore, to agglomerate only tightly connected nodes together, (5) can be modified using the inside and outside edges:

$$n_{\text{ms}}^{i} = \min_{k}(i,k) \longrightarrow \begin{Bmatrix} E_i < E_{ij} * \mu, N_i < N_j \\ E_j < E_{ij} * \mu, N_i > N_j \end{Bmatrix}, \quad (7)$$

where E_i is the inside edges of node i, E_j is the inside edges of node j, E_{ij} is the edges connecting node $i \wedge j$, and μ is the community density parameter.

After finding the similarity between nodes using (7), agglomeration of the most similar nodes starts. In every iteration, the most similar node for each node was found and if the most similar node of n_i is n_j, the opposite is not definite for n_j. Therefore, only the case in which both nodes choose each other is agglomerated as the most similar node. Thus, the agglomerated node n_{ag} is

$$n_{\text{ag}} = n_i n_j \longrightarrow \begin{Bmatrix} n_{\text{ms}}^{i} = j \\ n_{\text{ms}}^{j} = i \end{Bmatrix}, \quad (8)$$

where $k \in N$ and N is all the nodes in graph G.

The termination condition is met when all agglomerated nodes are connected more tightly inside than outside as seen in Pseudocode 1.

Input: Graph G, Nodes V, Edges E, density parameter μ
Output: Hierarchically clustered communities
(1) Find the eigenvectors $\alpha^{l} = [\alpha_1^{(l)}, \ldots, \alpha_{N_{\text{tr}}}^{(l)}]$ of $D_{\Omega}^{-1} M_D \Omega$
(2) Find the similarities of each node i and j using eigenvectors with Eq. (5)
(3) Compute the most similar node i using Eq. (7)
(4) Agglomerate the node i and j if the two nodes chose each other as the most similar node
(5) Re-initialize the graph with the agglomerated nodes and start the next iteration
(6) Agglomerate the nodes into hierarchical clusters when the iteration is finished

Pseudocode 1

4. Experiment

This section presents the results of the proposed method and compares the data with that of conventional community detection works [21, 22] by varying the value of parameters. LiveJournal and Orkut are used for evaluation as the ground-truth social network. LiveJournal is a blogging and social networking site that has been around since 1999. LiveJournal data has 4 million nodes and 35 million edges. The LiveJournal ground-truth data has 287,512 communities. In order to show the change of detected community by varying the density parameter for different networks, we also use Orkut network because the density difference of two networks can clearly emphasize the importance of choosing optimal density parameter. Orkut is a free online social network where users form friendship with each other. Orkut data has 3 million nodes and 117 million edges. The Orkut ground-truth data has 6,288,363 communities. The network is massive and complex, which makes more difficult clustering task. The dataset is available at https://snap.stanford.edu/data/.

The evaluation is done using the measurement metrics, such as precision (P), recall (R), and F-score.

$$P = \frac{T_p}{T_p + F_p}, \quad (9)$$

where T_p is the number of nodes that are correctly clustered and F_p is the number of nodes that are falsely clustered.

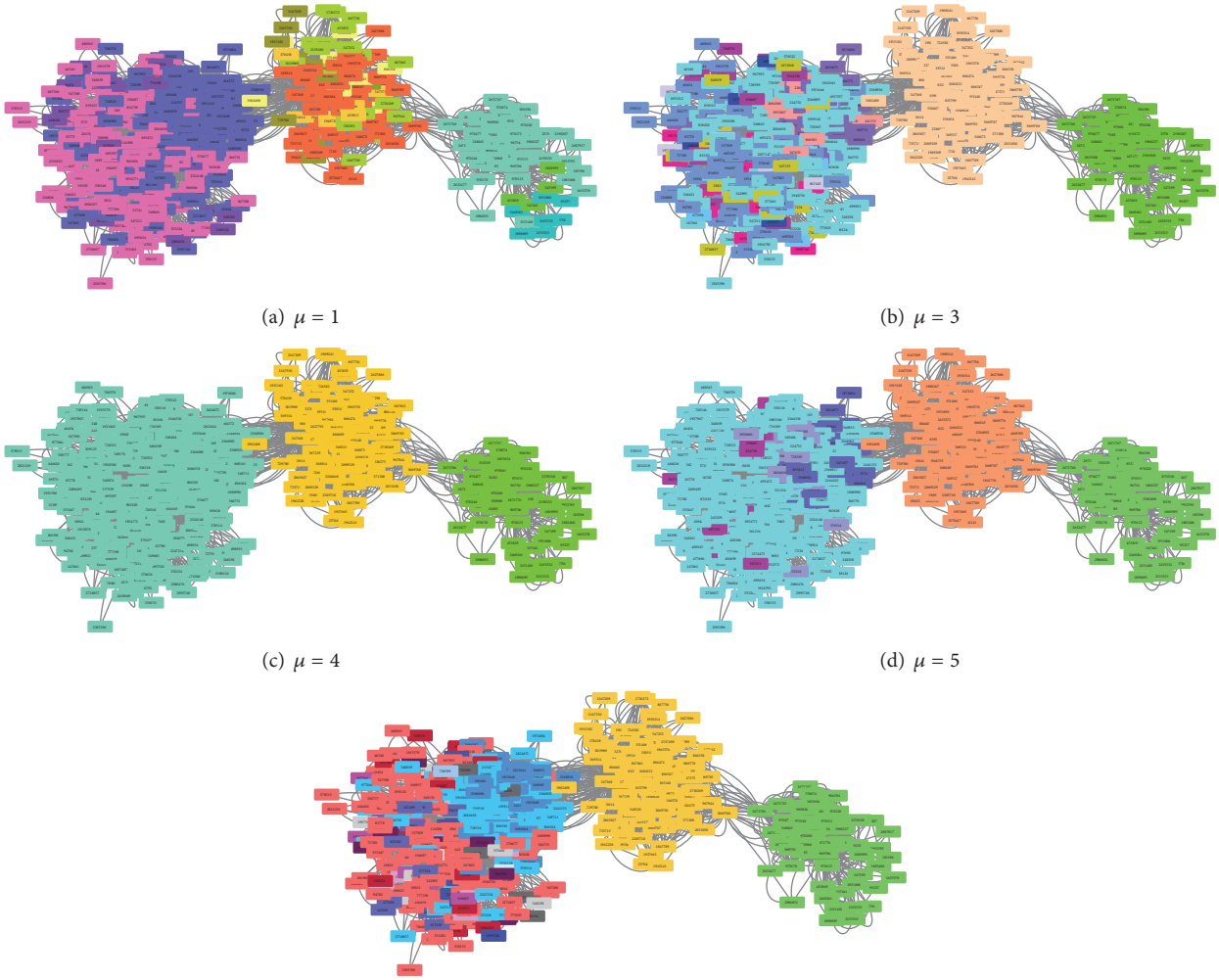
$$R = \frac{T_p}{T_p + F_n}, \quad (10)$$

where F_n is the number of nodes that are supposed to be clustered but failed to do so.

$$F\text{score} = \frac{2 * P * R}{P + R}, \quad (11)$$

where Fscore is the harmonic mean of precision and recall.

The results of clustering are shown by varying the value of μ, which is the community density parameter in Figure 2, where there are 3 communities. The right side community is the smallest with 61 nodes, the middle community has 109 nodes, and the left side community is the largest community with 331 communities. An optimal value was evaluated from the training data by a trial-and-error method. Beginning with $\mu = 1$, as shown in Figure 2(a), each community is clustered into small sized clusters internally. The clustering performance is improved by increasing the value of μ up to $\mu = 3$. The middle and right side communities are clustered successfully but the largest community on the left side failed to cluster because the left side community is very complex with many edges. In this case, the clustering performance can be improved by relaxing μ. The three communities are successfully clustered when $\mu = 4$, as shown in Figure 2(c). If the value of μ continues to increase, the clustering performance becomes worse than $\mu = 4$ because the clustering criteria are too relaxed as μ increases. With $\mu =$



(a) $\mu = 1$ (b) $\mu = 3$ (c) $\mu = 4$ (d) $\mu = 5$ (e) $\mu = 7$

FIGURE 2: Varying the results of the density parameter μ to obtain the optimal value.

5, the left side community is clustered into 4 communities, as shown in Figure 2(d), and when the value of μ reaches 7, the left side community is separated into smaller communities, as shown in Figure 2(e).

Figure 3 shows comparison by varying values of density parameter μ of Orkut network. Unlike LiveJournal network, Orkut is a more densely clustered network where the ratio of node and edges of LiveJournal is 1 : 8.6 whereas Orkut is 1 : 38.1. Therefore, the density parameter of Orkut requires being more strict compared to LiveJournal because the clusters are all densely clustered with each other. If the density parameter is not strict, it will allow the densely clustered nodes to be agglomerated together. First column of Figure 3 shows the ground-truth community which is colored in yellow and the following columns are detected networks by varying density parameters ranging from 0.1 to 4. As shown in the first row of Figure 3, the detected community with density parameter 0.1 shows well desired result but the accuracy has sharply dropped when we increased the density parameter because the detected community's size has continuously increased. The second row of Figure 3 shows different characteristics compared to the first row where the seed node has agglomerated into different cluster due to the relaxed density parameter. The third row network has similar characteristics to the second row network which shows that relaxed density parameter could lead to less densely clustered community. Fourth row network has similar results to the first row which shows that if we allow relaxed density parameter, the network will continue to expand. The optimal result of community detection has been obtained with 0.1 in Orkut network while the optimal value is 4 in LiveJournal network. Therefore, the experiment result shows that the density parameter is closely related to density of the network where the denser network requires stricter density parameter. It means that when evaluating the optimal value of density parameter, the density of the network should be considered.

Figure 4 shows the results of an analysis of the agglomeration process of the proposed method with $\mu = 4$ and AH-KSC as the number of iterations increase. Figure 4(a) shows the early stage agglomeration result of middle community

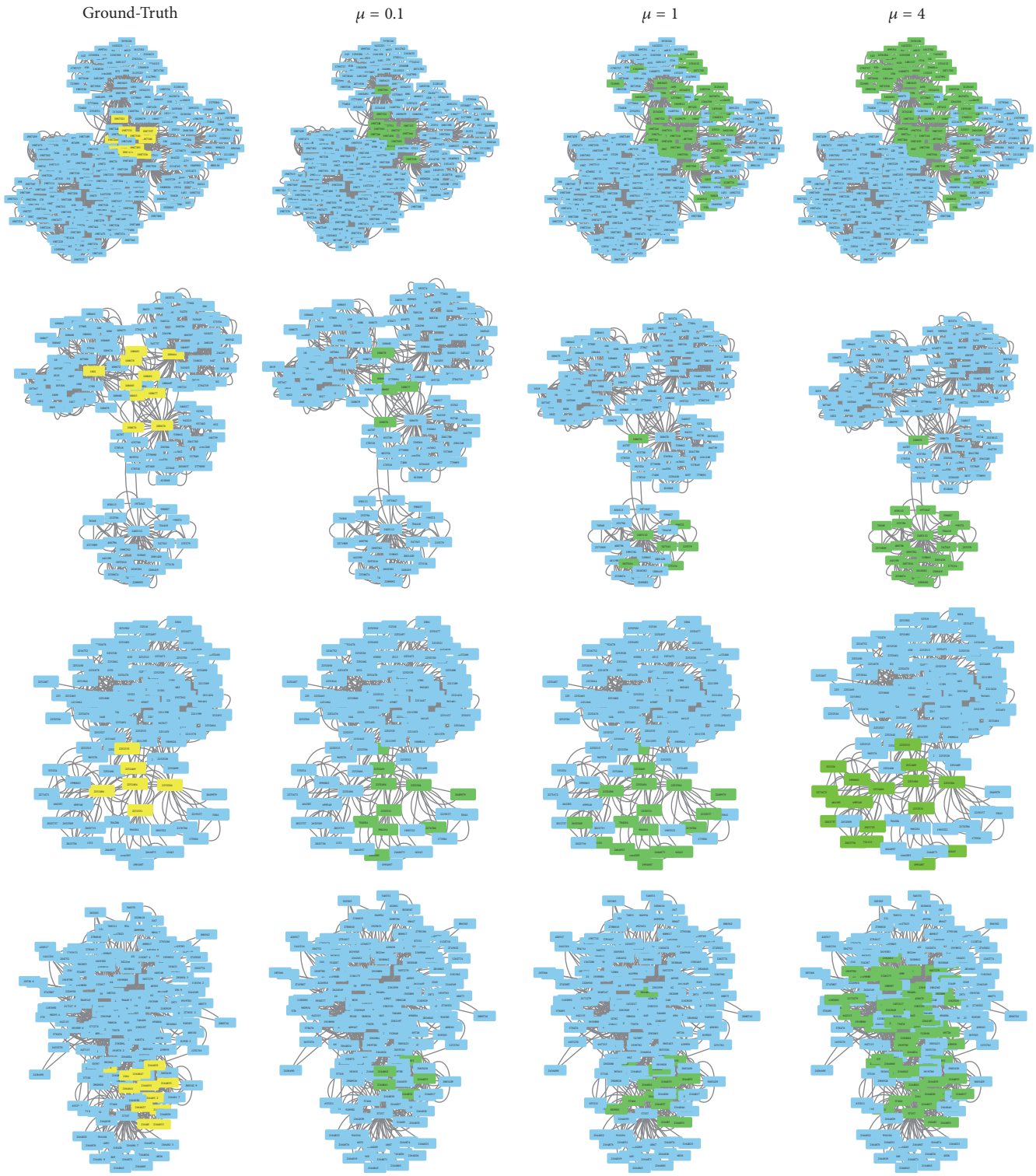
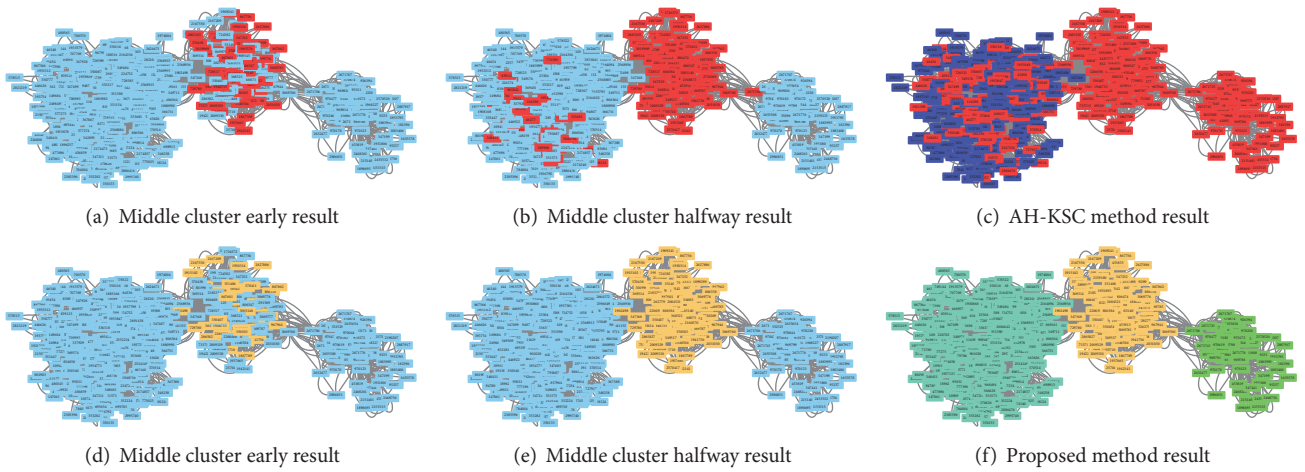


Figure 3: Comparison by varying density parameters on Orkut network.



(a) Middle cluster early result (b) Middle cluster halfway result (c) AH-KSC method result

(d) Middle cluster early result (e) Middle cluster halfway result (f) Proposed method result

Figure 4: Analysis of agglomeration process. (a, b, c) AH-KSC method. (d, e, f) Proposed method.

(at the 17th iteration). The early stage of AH-KSC was successfully clustering colored in red but in the halfway stage (at 26th iteration), the middle community was agglomerated together with some nodes that were included in the left side community, even though there were no direct connections to the nodes. In the late stage of agglomeration process (at the 30th iteration), the middle community was clustered with the right side community even though the right side community was the satisfied community, as shown in Figure 4(c). Figure 4(d) is the early stage of the proposed method. Like AH-KSC, the agglomeration process of the middle community is well clustered (at the 23rd iteration). In the halfway stage of agglomeration (at the 26th iteration), the middle community was clustered successfully because only directly connected nodes are agglomerated according to (5), where the number of edges between the nodes is added to the similarity function. At the late stage (at the 39th iteration), the right side community is clustered accurately because the ratio of the inner and outer edge connection is applied to (7) so that the smallest community on the right side has stopped agglomerating. Figure 4(f) shows the final result of the algorithm.

This study compared the accuracy of detection of the proposed method with AH-KSC and KSC using the ground-truth LiveJournal network. To show the comparison conveniently, only four parts of the network are used because the network is too large, that is, more than 4 million nodes. As shown in Figure 5, there are four subnetworks with different structures. The 1st network has 292 nodes and 1858 edges, and ground-truth community, to which the seed node belongs, has 24 nodes. The second network has 356 nodes and 33616 edges with a ground-truth community of 52 nodes. The third network has 652 nodes and 63044 edges, and the ground-truth community has 22 nodes. The last network has 119 nodes and 866 edges with a ground-truth community of 15 nodes. The 2nd and 3rd networks are so complex that it is difficult to detect communities while the 1st and 4th networks are well structured, that is, average in difficulty. In Figure 5, the light yellow colored node groups in the first column are the ground-truth community, the green colored node groups in the second column are the detected community from the proposed method, and the red colored node groups are the result of detected community from AH-KSC and KSC, respectively. From the observation of the experiment, AH-KSC agglomerates the neighbor nodes successfully in the early stages of agglomeration, as mentioned above, but it failed to terminate the agglomeration, as shown in the first and last networks in Figure 4 due to the lack of termination criteria. In addition, when the networks are too complex, such as the 2nd and 3rd networks, clustering is not done efficiently. KSC also produces similar results to AH-KSC but it clusters better than AH-KSC for the case in which the network is well organized, as with the 4th network. AH-KSC provides a better result than KSC when the network is very complex, such as the 2nd and 3rd network.

Table 1: Overall accuracy of community detection for the proposed method, AH-KSC, and KSC.

	Proposed method	AH-KSC	KSC
Precision	0.64	0.57	0.55
Recall	0.95	0.70	0.82
F-score	0.75	0.61	0.62

Table 1 shows overall accuracy of community detection using LiveJournal ground-truth network with respect to the precision, recall, and F-score for the proposed method, AH-KSC, and KSC. For AH-KSC, the average precision, recall, and F-score were 0.57, 0.7, and 0.61, respectively. For KSC, the average precision, recall, and F-score were 0.55, 0.82, and 0.62, respectively. For the proposed method, the average precision, recall, and F-score were 0.64, 0.95, and 0.75, respectively. The overall precision of AH-KSC and KSC were similar with a 2 percent difference but KSC has higher performance in the overall recall with more than 12 percent, which means the KSC detected more true positive nodes than AH-KSC from the network. The average F-score for AH-KSC and KSC is similar with only a 1% difference. The proposed

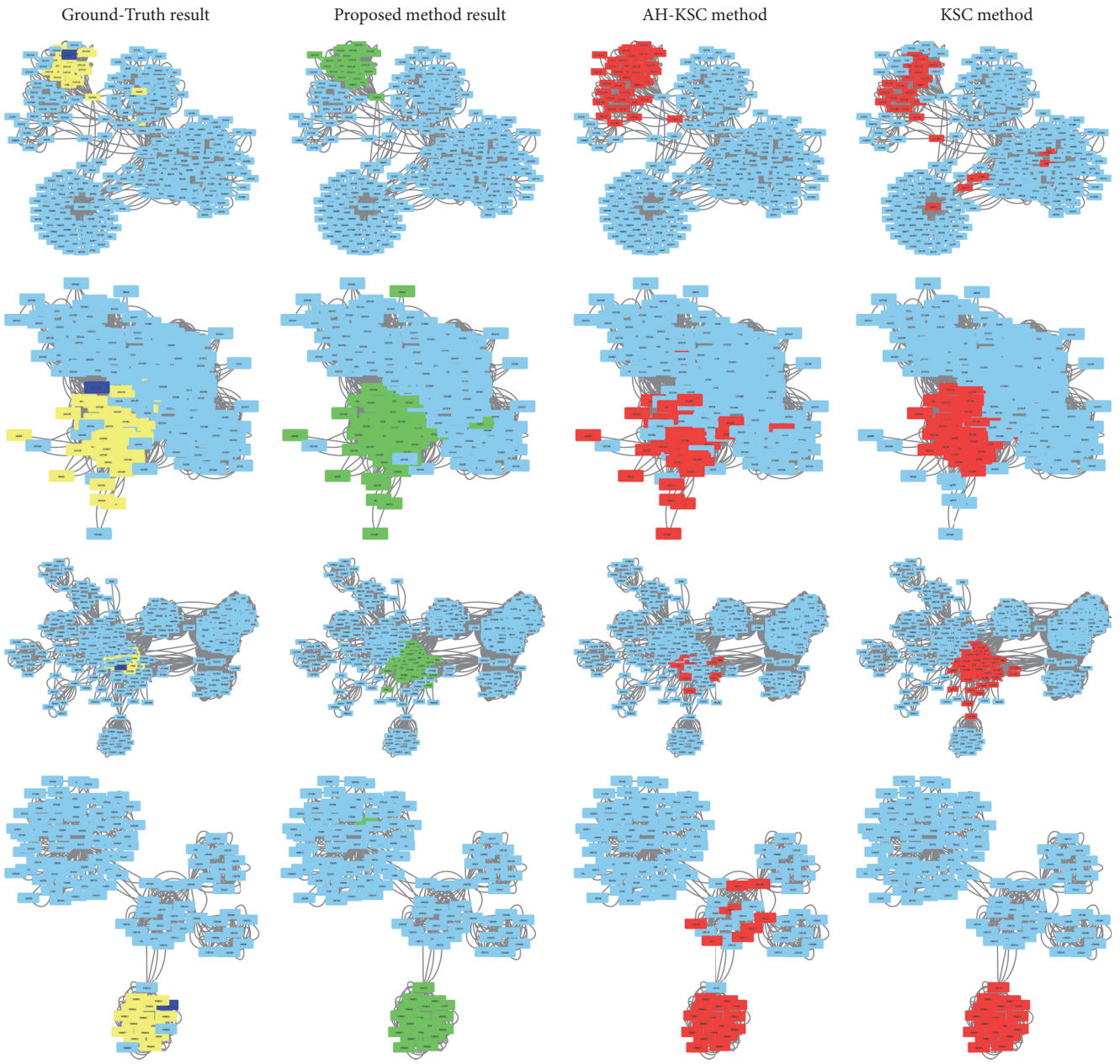


Figure 5: Example of detected community comparison.

method outperformed AH-KSC and KSC in all evaluation metrics. In average precision, the proposed method improved 7 to 9% compared to AH-KSC and KSC. In the average recall, the proposed method had the highest improvement with 25 to 13%. The average F-score of the proposed method is improved by 14%.

5. Conclusion

This paper introduced an agglomerative spectral clustering with conductance and edge weight for detecting communities. The proposed method projects the original points into eigenvector feature space in the first stage. In the second stage, the eigenvector space and the number of edges between nodes are used to evaluate the similarity between nodes. Each node finds candidate for the most similar nodes. The third stage finds the conductance between the node and its candidate. If only the conductance improves, the nodes are agglomerated. The three-stage process is iterated until the network requires no further agglomeration. The time complexity of the proposed method is increased compared to AH-KSC because we check the conductance of each agglomerated node but it is necessary for more accurate detection. From the analysis of the experiment, the proposed method outperformed the AH-KSC and KSC using a real life network, LiveJournal.

The two contributions of this method can be summarized as follows. One is the improvement accuracy compared to related works. The other is that the proposed method is feasible for practical situations because the performance of the method is well suited to real life social networks. On the other hand, the eigenvector space is calculated in every iteration so that the computation time is slower than that of KSC. Our future work will focus on improving the time complexity with a method such as parallel computing.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2016R1D1A1B03932447).

References

[1] S. Fortunato, "Community detection in graphs," *Physics Reports. A Review Section of Physics Letters*, vol. 486, no. 3-5, pp. 75–174, 2010.

[2] J. Leskovec, K. J. Lang, and M. Mahoney, "Empirical comparison of algorithms for network community detection," in *Proceedings of the 19th International World Wide Web Conference (WWW '10)*, pp. 631–640, ACM, New York, NY, USA, April 2010.

[3] M. Girvan and M. E. Newman, "Community structure in social and biological networks," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 99, no. 12, pp. 7821–7826, 2002.

[4] M. E. Newman, "Community detection and graph partitioning," *EPL (Europhysics Letters)*, vol. 103, no. 2, Article ID 28003, 2013.

[5] M. E. Newman, "Spectral methods for community detection and graph partitioning," *Physical Review E*, vol. 88, no. 4, 2013.

[6] S. C. Johnson, "Hierarchical clustering schemes," *Psychometrika*, vol. 32, no. 3, pp. 241–254, 1967.

[7] Y. Zhao and G. Karypis, "Hierarchical clustering algorithms for document datasets," *Data Mining and Knowledge Discovery*, vol. 10, no. 2, pp. 141–168, 2005.

[8] U. von Luxburg, "A tutorial on spectral clustering," *Statistics and Computing*, vol. 17, no. 4, pp. 395–416, 2007.

[9] Y. Ng. Andrew, M. I. Jordan, and Weiss Y., "On spectral clustering analysis and an algorithm," in *Proceedings of the Advances in Neural Information Processing Systems*, British Columbia, Canada, 2001.

[10] J. Yang and J. Leskovec, "Defining and evaluating network communities based on ground-truth," in *Proceedings of the 12th IEEE International Conference on Data Mining, ICDM 2012*, pp. 745–754, bel, December 2012.

[11] G. Karypis and V. Kumar, "A fast and high quality multilevel scheme for partitioning irregular graphs," *SIAM Journal on Scientific Computing*, vol. 20, no. 1, pp. 359–392, 1998.

[12] B. W. Kernighan and S. Lin, "An efficient heuristic procedure for partitioning graphs," *The Bell System Technical Journal*, vol. 49, no. 1, pp. 291–307, 1970.

[13] H. N. Djidjev, "A scalable multilevel algorithm for graph clustering and community structure detection," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 4936, pp. 117–128, 2006.

[14] A. Clauset, C. Moore, and M. E. J. Newman, "Hierarchical structure and the prediction of missing links in networks," *Nature*, vol. 453, no. 7191, pp. 98–101, 2008.

[15] M. C. Ramos, "Divisive and hierarchical clustering techniques to analyse variability of rainfall distribution patterns in a Mediterranean region," *Atmospheric Research*, vol. 57, no. 2, pp. 123–138, 2001.

[16] L. Lin, T. Luo, J. Fu, Z. Ji, and D. Xiao, "A new community detection based on agglomeration mechanism," in *Proceedings of the IEEE 2nd International Conference on Computing, Control and Industrial Engineering, CCIE 2011*, pp. 352–355, chn, August 2011.

[17] M. Leng, J. Wang, P. Wang, and X. Chen, "Hierarchical Agglomeration Community Detection Algorithm via Community Similarity Measures," *TELKOMNIKA Indonesian Journal of Electrical Engineering*, vol. 10, no. 6, pp. 1510–1518, 2012.

[18] I. S. Dhillon, Y. Guan, and B. Kulis, "Kernel k-means, spectral clustering and normalized cuts," in *Proceedings of the KDD-2004 - Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 551–556, Seattle, Wash, USA, August 2004.

[19] C. Alzate and J. A. K. Suykens, "Multiway spectral clustering with out-of-sample extensions through weighted kernel PCA," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 2, pp. 335–347, 2010.

[20] R. Langone, R. Mall, C. Alzate, and J. A. K. Suykens, "Kernel spectral clustering and applications," *Unsupervised Learning Algorithms*, pp. 135–161, 2016.

[21] R. Langone, C. Alzate, and J. A. K. Suykens, "Kernel spectral clustering for community detection in complex networks," in *Proceedings of the 2012 Annual International Joint Conference on Neural Networks, IJCNN 2012, Part of the 2012 IEEE World Congress on Computational Intelligence, WCCI 2012*, Queensland, Australia, June 2012.

[22] R. Mall, R. Langone, and J. A. K. Suykens, "Agglomerative hierarchical kernel spectral data clustering," in *Proceedings of the 5th IEEE Symposium on Computational Intelligence and Data Mining, CIDM 2014*, pp. 9–16, usa, December 2014.

[23] B. Hendrickson and T. G. Kolda, "Graph partitioning models for parallel computing," *Parallel Computing*, vol. 26, no. 12, pp. 1519–1534, 2000.

[24] C. F. Olson, "Parallel algorithms for hierarchical clustering," *Parallel Computing*, vol. 21, no. 8, pp. 1313–1325, 1995.

[25] A. R. Brodtkorb, T. R. Hagen, and M. L. Sætra, "Graphics processing unit (GPU) programming strategies and trends in GPU computing," *Journal of Parallel and Distributed Computing*, vol. 73, no. 1, pp. 4–13, 2013.

Hindawi
Complexity
Volume 2017, Article ID 3250301, 12 pages
https://doi.org/10.1155/2017/3250301

Research Article

On Measuring the Complexity of Networks: Kolmogorov Complexity versus Entropy

Mikołaj Morzy,[1] Tomasz Kajdanowicz,[2] and Przemysław Kazienko[2]

[1]*Institute of Computing Science, Poznan University of Technology, Piotrowo 2, 60-965 Poznań, Poland*
[2]*Department of Computational Intelligence, ENGINE-The European Centre for Data Science, Wrocław University of Science and Technology, Wybrzeże Wyspiańskiego 27, 50-370 Wrocław, Poland*

Correspondence should be addressed to Mikołaj Morzy; mikolaj.morzy@put.poznan.pl

Received 6 April 2017; Revised 27 July 2017; Accepted 13 August 2017; Published 1 November 2017

Academic Editor: Pasquale De Meo

One of the most popular methods of estimating the complexity of networks is to measure the entropy of network invariants, such as adjacency matrices or degree sequences. Unfortunately, entropy and all entropy-based information-theoretic measures have several vulnerabilities. These measures neither are independent of a particular representation of the network nor can capture the properties of the generative process, which produces the network. Instead, we advocate the use of the algorithmic entropy as the basis for complexity definition for networks. Algorithmic entropy (also known as Kolmogorov complexity or K-complexity for short) evaluates the complexity of the description required for a lossless recreation of the network. This measure is not affected by a particular choice of network features and it does not depend on the method of network representation. We perform experiments on Shannon entropy and K-complexity for gradually evolving networks. The results of these experiments point to K-complexity as the more robust and reliable measure of network complexity. The original contribution of the paper includes the introduction of several new entropy-deceiving networks and the empirical comparison of entropy and K-complexity as fundamental quantities for constructing complexity measures for networks.

1. Introduction

Networks are becoming increasingly more important in contemporary information science due to the fact that they provide a holistic model for representing many real-world phenomena. The abundance of data on interactions within complex systems allows network science to describe, model, simulate, and predict behaviors and states of such complex systems. It is thus important to characterize networks in terms of their complexity, in order to adjust analytical methods to particular networks. The measure of network complexity is essential for numerous applications. For instance, the level of network complexity can determine the course of various processes happening within the network, such as information diffusion, failure propagation, actions related to control, or resilience preservation. Network complexity has been successfully used to investigate the structure of software libraries [1], to compute the properties of chemical structures [2], to assess the quality of business processes [3–5], and to provide general characterizations of networks [6, 7].

Complex networks are ubiquitous in many areas of science, such as mathematics, biology, chemistry, systems engineering, physics, sociology, and computer science, to name a few. Yet the very notion of network complexity lacks a strict and agreed-upon definition. In general, a network is considered "complex" if it exhibits many features such as small diameter, high clustering coefficient, anticorrelation of node degrees, presence of network motifs, and modularity structures [8]. These features are common in real-world networks, but they rarely appear in artificial random networks. Finding a good metric with which one can estimate the complexity of a network is not a trivial task. A good complexity measure should not depend solely on the number of vertices and edges, but it must take into consideration topological characteristics of the network. In addition, complexity is not synonymous with randomness

or unexpectedness. As has been pointed out [8], within the spectrum of possible networks, from the most ordered (cliques, paths, and stars) to the most disordered (random networks), complex networks occupy the very center of this spectrum. Finally, a good complexity measure should not depend on a particular network representation and should yield consistent results for various representations of the same network (adjacency matrix, Laplacian matrix, and degree sequence). Unfortunately, as current research suggests, finding a good complexity measure applicable to a wide variety of networks is very challenging [9–11].

Among many possible measures which can be used to define the complexity of networks, the entropy of various network invariants has been by far the most popular choice. Network invariants considered for defining entropy-based complexity measures include number of vertices, number of neighbors, number of neighbors at a given distance [12], distance between vertices [13], energy of network matrices such as Randić matrix [14] or Laplacian matrix [15], and degree sequences. There are multiple definitions of entropies, usually broadly categorized into three families: thermodynamic entropies, statistical entropies, and information-theoretic entropies. In the field of computer science, information-theoretic measures are the most prevalent, and they include Shannon entropy [16], Kolmogorov-Sinai entropy [17], and Rényi entropy [18]. These entropies are based on the concept of the information content of a system and they measure the amount of information required to transmit the description of an object. The underlying assumption of using information-theoretic definitions of entropy is that uncertainty (as measured by entropy) is a nondecreasing function of the amount of available information. In other words, systems in which little information is available are characterized by low entropy and therefore are considered to be "simple." The first idea to use entropy to quantify the complexity of networks comes from Mowshowitz [19].

Despite the ubiquitousness of general-purpose entropy definitions, many researchers have developed specialized entropy definitions aimed at describing the structure of networks [10]. Notable examples of such definitions include the proposal by Ji et al. to measure the unexpectedness of a particular network by comparing it to the number of possible network configurations available for a given set of parameters [20]. This concept is clearly inspired by algorithmic entropy, which defines the complexity of a system not in terms of its information content, but in terms of its generative process. A different approach to measure the entropy of networks has been introduced by Dehmer under the form of information functional [21]. Information functional can be also used to quantify network entropy in terms of k-neighborhoods of vertices [12, 13] or independent sets of vertices [22]. Yet another approach to network entropy has been proposed by Körner, who advocates the use of stable sets of vertices as the basis to compute network entropy [23]. Several comprehensive surveys of network entropy applications are also available [9, 11].

Within the realm of information science, the complexity of a system is most often associated with the number of possible interactions between elements of the system. Complex systems evolve over time, they are sensitive to even minor perturbations at the initial steps of development and often involve nontrivial relationships between constituent elements. Systems exhibiting high degree of interconnectedness in their structure and/or behavior are commonly thought to be difficult to describe and predict, and, as a consequence, such systems are considered to be "complex." Another possible interpretation of the term "complex" relates to the size of the system. In the case of networks, one might consider to use the number of vertices and edges to estimate the complexity of a network. However, the size of the network is not a good indicator of its complexity, because networks which have well-defined structures and behaviors are, in general, computationally simple.

In this work, we do not introduce a new complexity measure or propose new informational functional and network invariants, on which an entropy-based complexity measure could be defined. Rather, we follow the observations formulated in [24] and we present the criticism of the entropy as the guiding principle of complexity measure construction. Thus, we do not use any specific formal definition of complexity, but we provide additional arguments why entropy may be easily deceived when trying to evaluate the complexity of a network. Our main hypothesis is that algorithmic entropy, also known as Kolmogorov complexity, is superior to traditional Shannon entropy due to the fact that algorithmic entropy is more robust, less dependent on the network representation, and better aligned with intuitive human understanding of complexity.

The organization of the paper is the following. In Section 2, we introduce basic definitions related to entropy and we formulate arguments against the use of entropy as the complexity measure of networks. Section 2.3 presents several examples of entropy-deceiving networks, which provide both motivation and anecdotal evidence for our hypothesis. In Section 3, we introduce Kolmogorov complexity and we show how this measure can be applied to networks, despite its high computational cost. The results of the experimental comparison of entropy and Kolmogorov complexity are presented in Section 4. The paper concludes in Section 5 with a brief summary and future work agenda.

2. Entropy as the Measure of Network Complexity

2.1. Basic Definitions. Let us introduce basic definitions and notation used throughout the remainder of this paper. A *network* is an ordered pair $G = \langle V, E \rangle$, where $V = \{v_1, \ldots, v_N\}$ is the set of *vertices* and $E = \{(v_i, v_j) \in V \times V\}$ is the set of *edges*. The *degree* $d(v_i)$ of the vertex v_i is the number of vertices adjacent to it, $d(v_i) = |\{v_j : (v_i, v_j) \in E\}|$. A given network can be represented in many ways, for instance, using an *adjacency matrix* defined as

$$A_{N\times N}[i, j] = \begin{cases} 1 & \text{if } (v_i, v_j) \in E \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

An alternative to the adjacency matrix is the Laplacian matrix of the network defined as

$$L_{N\times N}[i,j] = \begin{cases} d(v_i) & \text{if } i = j \\ -1 & \text{if } i \neq j,\ (v_i, v_j) \in E \\ 0 & \text{otherwise.} \end{cases} \quad (2)$$

Other popular representations of networks include the *degree list* defined as $D = \langle d(v_1), d(v_2), \ldots, d(v_n)\rangle$ and the *degree distribution* defined as

$$P(d_i) = p\left(d\left(v_j\right) = d_i\right) = \frac{\left|\left\{v_j \in V : d\left(v_j\right) = d_i\right\}\right|}{N}. \quad (3)$$

Although there are numerous different definitions of entropy, in this work we are focusing on the definition most commonly used in information sciences, the Shannon entropy [16]. This measure represents the amount of information required to provide the statistical description of the network. Given any discrete random variable X with n possible outcomes, the Shannon entropy $H(X)$ of the variable X is defined as the function of the probability p of all outcomes of X:

$$H(X) = -\sum_{i=1}^{n} p(x_i) \log_b p(x_i). \quad (4)$$

Depending on the selected base of the logarithm, the entropy is expressed in bits ($b = 2$), nats ($b = e$), or dits ($b = 10$) (bits are also known as Shannon, and dits are also known as Hartley). The above definition applies to discrete random variables; for random variables with continuous probability distributions differential entropy is used, usually along with the limiting density of discrete points. Given a variable X with n possible discrete outcomes such that in the limit $n \to \infty$ the density of X approaches the invariant measure $m(x)$, the continuous entropy is given by

$$\lim_{n\to\infty} H(X) = -\int p(x) \frac{p(x)}{m(x)}\, dx. \quad (5)$$

In this work, we are interested in measuring the entropy of various network invariants. These invariants can be regarded as discrete random variables with the number of possible outcomes bound by the size of the available alphabet, either binary (in the case of adjacency matrices) or decimal (in the case of other invariants). Consider the 3-regular graph presented in Figure 1. This graph can be described using the following adjacency matrix:

$$A_{10\times 10} = \begin{bmatrix} 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \end{bmatrix}. \quad (6)$$

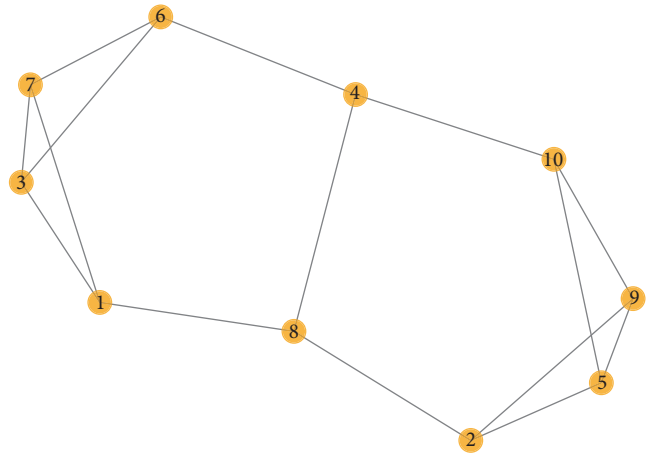


Figure 1: Three-regular graph with 10 vertices.

This matrix, in turn, can be flattened to a vector (either row-wise or column-wise), and this vector can be treated as a random variable with two possible outcomes, 0 and 1. Counting the number of occurrences of these outcomes, we arrive at the random variable $X = \{x_0 = 0.7, x_1 = 0.3\}$ and its entropy $H(X) = 0.88$. Alternatively, this graph can be described using the degree list $D = \langle 3, 3, 3, 3, 3, 3, 3, 3, 3, 3\rangle$ which can be treated as the random variable with the entropy $H(D) = 0$. Yet another possible random variable that can be derived from this graph is the degree distribution $PD = \{d_0 = 0, d_1 = 0, d_2 = 0, d_3 = 1\}$ with the entropy $H(PD) = 0$. In summary, any network invariant can be used to extract a random variable and compute its entropy.

Thus, in the remainder of the paper, whenever mentioning entropy, we will refer to the entropy of a discrete random variable. In general, the higher the randomness, the greater the entropy. The value of entropy is maximal for a random variable with the uniform distribution and the minimum value of entropy is attained by a constant random variable. This kind of entropy will be further explored in this paper in order to reveal its weaknesses.

As an alternative to Shannon entropy, we advocate the use of Kolmogorov complexity. We postpone the discussion of Kolmogorov complexity to Section 3, where we provide both its definition and the method to approximate this incomputable measure. For the sake of brevity, in the remainder of this paper, we will use the term "entropy" to refer to Shannon entropy and the term "K-complexity" to refer to Kolmogorov complexity.

2.2. Why Is Entropy a Bad Measure of Network Complexity. Zenil et al. [24] argue that entropy is not appropriate to measure the true complexity of a network and they present several examples of networks which should not qualify as complex (using the colloquial understanding of the term), yet which attain maximum entropy of various network invariants. We follow the line of argumentation of Zenil et al., and we present more examples of entropy-deceiving networks. Our main aim is to show that it is relatively easy

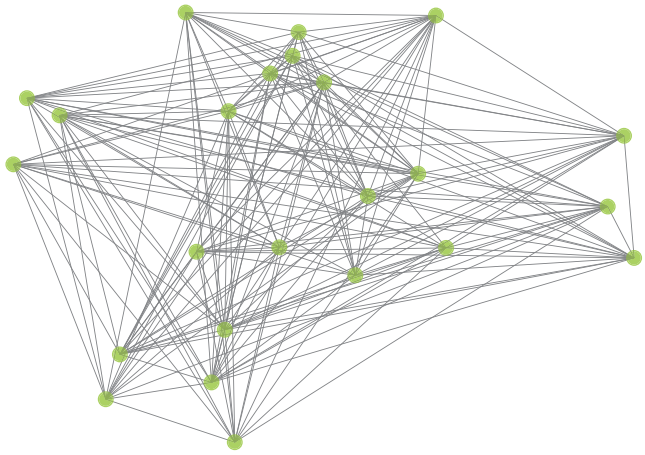


Figure 2: Block network composed of eight of the same 3-node blocks.

to construct a network which achieves high values of entropy of various network invariants. Examples presented in this section outline the main problem with using entropy as the basis for complexity measure construction: namely, that entropy is not aligned with intuitive human understanding of complexity. Statistical randomness, as measured by entropy, does not imply complexity in a useful, operational way.

The main reason why entropy and other entropy-related information-theoretic measures fail to correctly describe the complexity of a network is the fact that these measures are not independent of the network representation. As a matter of fact, this remark applies equally to all computable measures of network complexity. It is quite easy to present examples of two equivalent lossless descriptions of the same network having very different entropy values, as we will show in Section 2.3. In this paper, we experiment with four different representations of networks: adjacency matrices, Laplacian matrices, degree lists, and degree distributions. We show empirically that the choice of a particular representation of the network strongly influences the resulting entropy estimation.

Another property which makes entropy a questionable measure of network complexity is the fact that entropy cannot be applied to several network features at the same time, but it operates on a single feature, for example, degree and betweenness. In theory, one could devise a function which would be a composition of individual features, but high complexity of the composition does not imply high complexity of all its components and vice versa. This requirement to select a particular feature and compute its probability distribution disqualifies entropy as a universal and independent measure of complexity.

In addition, an often forgotten aspect of entropy is the fact that measuring entropy requires making an arbitrary choice regarding the aggregation level of the variable, for which entropy is computed. Consider the network presented in Figure 2. At the first glance, this network seems to be fairly random. The density of the network is 0.35 and its entropy computed over adjacency matrix is 0.92 bits. However, this network has been generated using a very simple procedure. We begin with the initial matrix:

$$M_{3\times 3} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix}. \tag{7}$$

Next, we create 64 copies of this matrix, and each of these copies is randomly transposed. Finally, we bind all these matrices together to form a square matrix $M_{24\times 24}$ and we use it as the adjacency matrix to create the network. So, if we were to coalesce the adjacency matrix into 3×3 blocks, the entropy of the adjacency matrix would be 0, since all constituent blocks are the same. It would mean that the network is actually deterministic and its complexity is minimal. On the other hand, it should be noted that this shortcoming of entropy can be circumvented by using the *entropy rate* (*n*-gram entropy) instead, because entropy rate calculates the entropy for all possible levels of granularity of a variable. Given a random variable $X = \langle x_1, x_2, \ldots, x_n \rangle$, let $p(x_i, x_{i+1}, \ldots, x_{i+l})$ denote the joint probability over l consecutive values of X. Entropy rate $H_l(X)$ of a sequence of l consecutive values of X is defined as

$$H_l(X) = -\sum_{x_1 \in X} \cdots \sum_{x_l \in X} p(x_1, \ldots, x_l) \log_2 p(x_1, \ldots, x_l). \tag{8}$$

Entropy rate of the variable X is simply the limit of the above estimation for $l \to \infty$.

2.3. Entropy-Deceiving Networks. In this section, we present four different examples of entropy-deceiving networks, similar to the idea coined in [24]. Each of these networks has a simple generative procedure and should not (intuitively) be treated as complex. However, if the entropy was used to construct a complexity measure, these networks would have been qualified as complex. The examples given in this section disregard any specific definition of complexity; their aim is to outline main shortcomings of entropy as the basis for any complexity measure construction.

2.3.1. Degree Sequence Network. Degree sequence network is an example of a network which has an interesting property: there are exactly two vertices for each degree value $1, 2, \ldots, N/2$; $N = |V|$.

The procedure to generate degree sequence network is very simple. First, we create a linked list of all N vertices, for which $d(v_1) = d(v_N) = 1$ and $\forall i \neq 1, i \neq N, d(v_i) = 2$. It is a circle without one edge (v_1, v_N). Next, starting with vertex v_3, we follow a simple rule:

for $i = 3$ **to** $N/2$ **do**
  for $j = 1$ **to** $(i - 2)$ **do**
    add_edge$(v_i, v_{N/2+j})$
  end for
end for

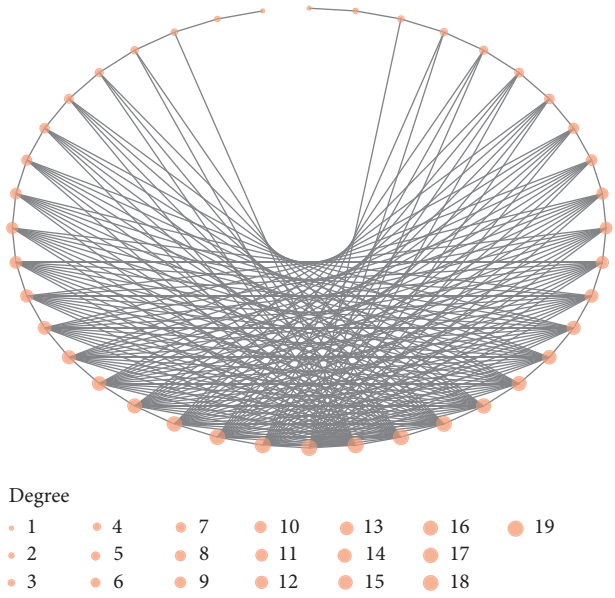


Figure 3: Degree sequence network.

The resulting network is presented in Figure 3. It is very regular, with a uniform distribution of vertex degrees, due to its straightforward generation procedure. However, if one would examine the entropy of the degree sequence, this entropy would be maximal for a given number N of vertices, suggesting far greater randomness of such network. This example shows that entropy of the degree sequence (and the entropy of the degree distribution) can be very misleading when trying to evaluate the true complexity of a network.

2.3.2. Copeland-Erdös Network. The Copeland-Erdös network is a network which seems to be completely random, despite the fact that the procedure of its generation is deterministic. The Copeland-Erdös constant is a constant which is produced by concatenating "0" with the sequence of consecutive prime numbers [25]. When prime numbers are expressed in base 10, the Copeland-Erdös constant is a normal number; that is, its infinite sequence of digits is uniformly distributed (the normality of the Copeland-Erdös constant in bases other than 10 is not proven). This fact allows us to devise the following simple generative procedure for a network. Given the number of vertices N, take the first N^2 digits of the Copeland-Erdös constant and represent them as the matrix of the size $N \times N$. Next, binarize each value in the matrix using the function $f(x) = x\text{div}5$ (integer division) and use it as the adjacency matrix to create a network. Since each digit in the matrix is approximately equally likely, the resulting binary matrix will have approximately the same number of 0's and 1's. An example of the Copeland-Erdös network is presented in Figure 4. The entropy of the adjacency matrix is maximal for a given number of N vertices; furthermore, the network may seem to be random and complex, but its generative procedure, as we can see, is very simple.

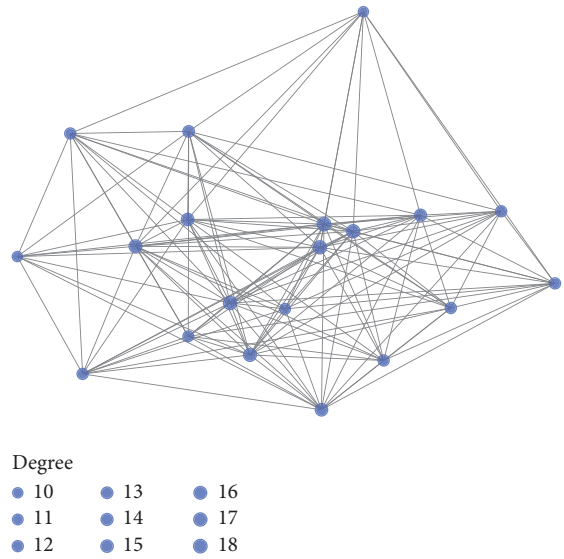


Figure 4: Copeland-Erdös network.

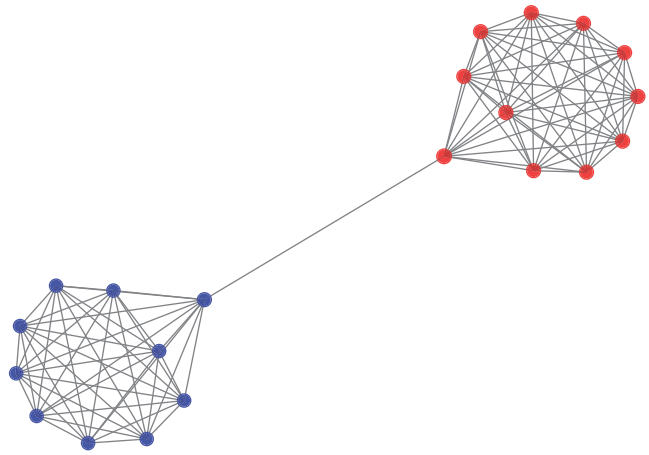


Figure 5: 2-Clique network.

2.3.3. 2-Clique Network. 2-Clique network is an artificial example of a network in which the entropy of the adjacency matrix is maximal. The procedure to generate this network is as follows. We begin with two connected vertices labeled *red* and *blue*. We add *red* and *blue* vertices alternatingly, each time connecting the newly added vertex with all other vertices of the same color. As a result, two cliques appear (see Figure 5). Since there are as many *red* vertices as there are *blue* vertices, the adjacency matrix contains the same number of 0's and 1's (not considering the 1 representing the bridge edge between cliques). So, entropy of the adjacency matrix is close to maximal, although the structure of the network is trivial.

2.3.4. Ouroboros Network. Ouroboros (Ouroboros is an ancient symbol of a serpent eating its own tail, appearing first in Egyptian iconography and then gaining notoriety in later magical traditions) network is another example of an entropy-deceiving network. The procedure to generate this

network is very simple: for a given number N of vertices, we create two closed rings, each consisting of $N/2$ vertices, and we connect corresponding vertices of the two rings. Finally, we break a single edge in one ring and we put a single vertex at the end of the broken edge. The result of this procedure can be seen in Figure 6. Interestingly, even though almost all vertices in this network have equal degree of 3, each vertex has different betweenness. Thus, the entropy of the betweenness sequence is maximal, suggesting a very complex pattern of communication pathways though the network. Obviously, this network is very simple from the communication point of view and should not be considered complex.

3. K-Complexity as the Measure of Network Complexity

We strongly believe that Kolmogorov complexity (K-complexity) is a much more reliable and robust basis for constructing the complexity measure for compound objects, such as networks. Although inherently incomputable, K-complexity can be easily approximated to a degree which allows for the practical use of K-complexity in real-world applications, for instance, in machine learning [26, 27], computer network management [28], and general computation theory (proving lower bounds of various Turing machines, combinatorics, formal languages, and inductive inference) [29].

Let us now introduce the formal framework for K-complexity and its approximation. Note that entropy is defined for any random variable, whereas K-complexity is defined for strings of characters only. K-complexity $K_T(s)$ of a string s is formally defined as

$$K_T(s) = \min\{|P|, T(P) = s\}, \tag{9}$$

where P is a program which produces the string s when run on a universal Turing machine T and $|P|$ is the length of the program P, that is, the number of bits required to represent P. Unfortunately, K-complexity is incomputable [30], or more precisely, it is upper semicomputable (only the upper bound of the value of K-complexity can be computed for a given string s). One way for approximating the true value of $K_T(s)$ is to use the notion of algorithmic probability introduced by Solomonoff and Levin [31, 32]. Algorithmic probability $p^a(s)$ of a string s is defined as the expected probability that a random program P running on a universal Turing machine T with the binary alphabet produces the string s upon halting:

$$p^a(s) = \sum_{P:T(P)=s} \frac{1}{2^{|P|}}. \tag{10}$$

Of course there are $2^{|P|}$ possible programs of the length $|P|$, and the summation is performed over all possible programs without limiting their length, which makes algorithmic probability $p^a(s)$ a semimeasure which itself is incomputable. Nevertheless, algorithmic probability can be used to calculate K-complexity using the Coding Theorem [31] which states that algorithmic probability approximates K-complexity up to a constant c:

$$\left|-\log_2 p^a(s) - K_T(s)\right| \leq c. \tag{11}$$

The consequence of the Coding Theorem is that it associates the frequency of occurrence of the string s with its complexity. In other words, if a particular string s can be generated by many different programs, it is considered "simple." On the other hand, if a very specific program is required to produce the given string s, this string can be regarded as "complex." The Coding Theorem also implies that K-complexity of a string s can be approximated from its frequency using the formula:

$$K_T(s) \approx -\log_2 p^a(s). \tag{12}$$

This formula has inspired the Algorithmic Nature Lab group (https://www.algorithmicnaturelab.org) to develop the CTM (Coding Theorem Method), a method to approximate K-complexity by counting output frequencies of small Turing machines. Clearly, algorithmic probability of the string s cannot be computed exactly, because the formula for algorithmic probability requires finding all possible programs that produce the string s. Nonetheless, for a limited subset of Turing machines it is possible to count the number of machines that produce the given string s, and this is the trick behind the CTM. In broad terms, the CTM for a string s consists in computing the following function:

$$\begin{aligned} \mathrm{CTM}(s) &= D(n, m, s) \\ &= \frac{|\{T \in \mathcal{T}(n,m) : T(P) = s\}|}{|\{T \in \mathcal{T}(n,m) : T(P) : \text{halts}\}|}, \end{aligned} \tag{13}$$

where $\mathcal{T}(n, m)$ is the space of all universal Turing machines with n states and m symbols. Function $D(n, m, s)$ computes the ratio of all halting machines with n states and m symbols which produce the string s and its value is determined with the help of known values of the famous Busy Beaver function [33]. The Algorithmic Nature Lab group has gathered statistics on almost 5 million short strings (maximum length is 12 characters) produced by Turing machines with alphabets ranging from 2 to 9 symbols, and based on these statistics the CTM can approximate the algorithmic probability of a given string. Detailed description of the CTM can be found in [34]. Since the function $D(n, m, s)$ is an approximation of the true algorithmic probability $p^a(s)$, it can also be used to approximate K-complexity of the string s.

The CTM can be applied only to short strings consisting of 12 characters or less. For larger strings and matrices, the BDM (Block Decomposition Method) should be used. The BDM requires the decomposition of the string s into (possibly overlapping) blocks $\{b_1, b_2, \ldots, b_k\}$. Given a long string s, the BDM computes its algorithmic probability as

$$\mathrm{BDM}(s) = \sum_{i=1}^{k} \mathrm{CTM}(b_i) + \log_2 |b_i|, \tag{14}$$

where $\mathrm{CTM}(b_i)$ is the algorithmic complexity of the block b_i and $|b_i|$ denotes the number of times the block b_i appears in s. Detailed description of the BDM can be found in [35].

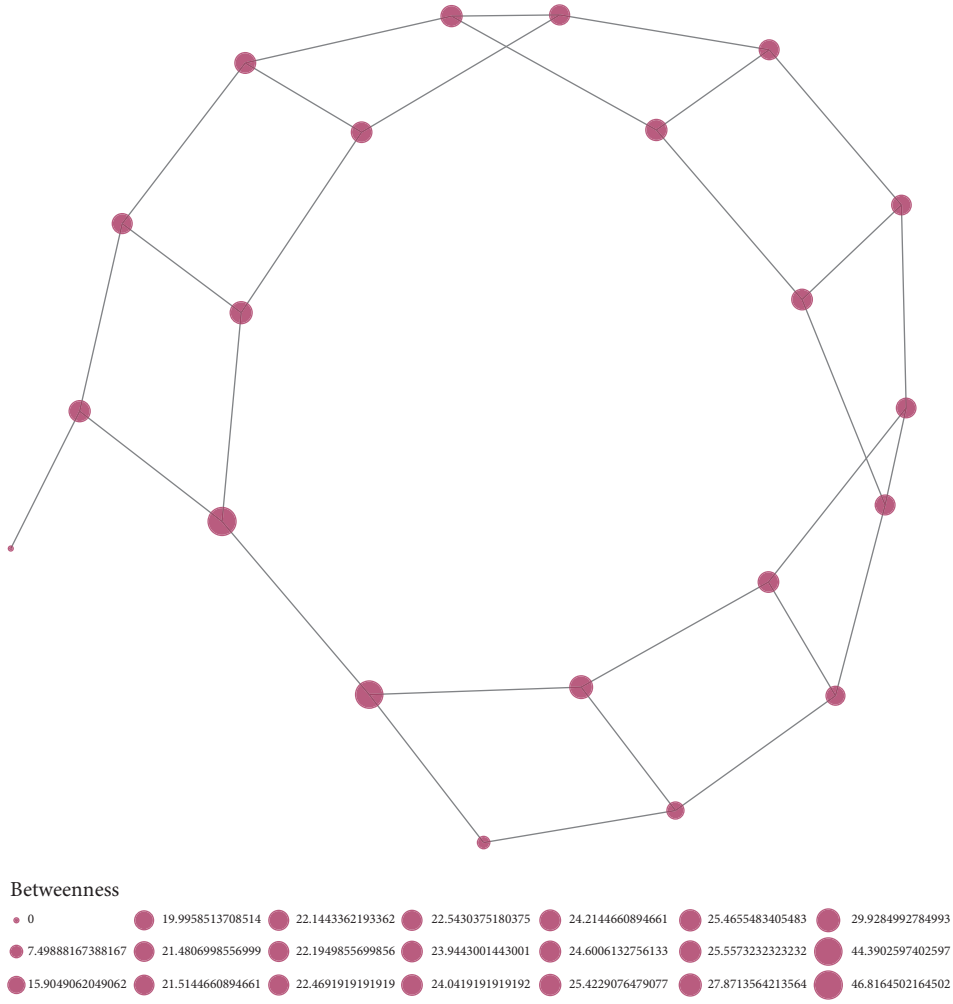


Figure 6: Ouroboros network.

Obviously, any representation of a nontrivial network requires far more than 12 characters. Consider once again the 3-regular graph presented in Figure 1. The Laplacian matrix representation of this graph is the following:

$$L_{10\times 10} = \begin{bmatrix} 3 & 0 & -1 & 0 & -1 & 0 & 0 & 0 & -1 & 0 \\ 0 & 3 & 0 & 0 & -1 & 0 & 0 & -1 & -1 & 0 \\ -1 & 0 & 3 & -1 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 3 & 0 & -1 & -1 & 0 & 0 & 0 \\ -1 & -1 & 0 & 0 & 3 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & -1 & -1 & 0 & 3 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 3 & -1 & 0 & -1 \\ 0 & -1 & 0 & 0 & 0 & -1 & -1 & 3 & 0 & 0 \\ -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & -1 \\ 0 & 0 & 0 & 0 & -1 & 0 & -1 & 0 & -1 & 3 \end{bmatrix}. \quad (15)$$

If we treat each row of the Laplacian matrix as a separate block, the string representation of the Laplacian matrix becomes $s = \{b_1 = 3010100010, b_2 = 0300100110, \ldots, b_{10} = 0000101013\}$ (for the sake of simplicity, we have replaced the symbol "−1" with the symbol "1"). This input can be fed into the BDM, producing the final estimation of the algorithmic probability (and, consequently, the estimation of the K-complexity) of the string representation of the Laplacian matrix. In our experiments, whenever reporting the values of K-complexity of the string s, we actually report the value of BDM(s) as the approximation of the true K-complexity.

4. Experiments

4.1. Gradual Change of Networks. As we have stated before, the aim of this research is not to propose a new complexity measure for networks, but to compare the usefulness and robustness of entropy versus K-complexity as the underlying foundations for complexity measures. Let us recall what properties are expected from a good and reliable complexity measure for networks. Firstly, the measure should not

depend on the particular network representation but should yield more or less consistent results for all possible lossless representations of a network. Secondly, the measure should not equate complexity with randomness. Thirdly, the measure should take into consideration topological properties of a network and not be limited to simple counting of the number of vertices and edges. Of course, statistical properties of a given network will vary significantly between different network invariants, but at the base level of network representation the quantity used to define the complexity measure should fulfill the above requirements. The main question that we are aiming to answer in this study is whether there are qualitative differences between entropy and K-complexity with regard to the above-mentioned requirements when measuring various types of networks.

In order to answer this question we have to measure how a change in the underlying network structure affects the observed values of entropy and K-complexity. To this end, we have devised two scenarios. In the first scenario, the network gradually transforms from the perfectly ordered state to a completely random state. The second transformation brings the network from the perfectly ordered state to a state which can be understood as semiordered, albeit in a different way. The following sections present both scenarios in detail.

4.1.1. From Watts-Strogatz Small-World Model to Erdös-Rényi Random Network Model. A small-world network model introduced by Watts and Strogatz [36] is based on the process, which transforms a fully ordered network with no random edge rewiring into a random network. According to the small-world model, vertices of the network are placed on a regular k-dimensional grid and each vertex is connected to exactly m of its nearest neighbors, producing a regular lattice of vertices with equal degrees. Then, with a small probability p, each edge is randomly rewired. If $p = 0$, no rewiring occurs and the network is fully ordered. All vertices have the same degree, the same betweenness, and the entropy of the adjacency matrix depends only on the density of edges. When $p \geq 0$, edge rewiring is applied to edges and this process distorts the degree distribution of vertices.

On the other end of the network spectrum is the Erdös-Rényi random network model [37], in which there is no inherent pattern of connectivity between vertices. The random network emerges by selecting all possible pairs of vertices and creating, for each pair, an edge with the probability p. Alternatively, one can generate all possible networks consisting of n vertices and m edges and then randomly pick one of these networks. The construction of the random network implies the highest degree of randomness, and there is no other way of describing a particular instance of such network other than by explicitly providing its adjacency matrix or the Laplacian matrix.

In our first experiment, we observe the behavior of entropy and K-complexity being applied to gradually changing networks. We begin with a regular small-world network generated for $p = 0$. Next, we iteratively increase the value of p by 0.01 in each step, until $p = 1$. We retain the network between iterations, so conceptually it is one network undergoing the transition. Also, we only consider rewiring of edges which have not been rewired during preceding iterations, so every edge is rewired at most once. For $p = 0$, the network forms a regular lattice of vertices, and for $p = 1$ the network is fully random with all edges rewired. While randomly rewiring edges, we do not impose any preference on the selection of the target vertex of the edge being currently rewired; that is, each vertex has a uniform probability of being selected as the target vertex of rewiring.

4.1.2. From Watts-Strogatz Small-World Model to Barabási-Albert Preferential Attachment Model. Another popular model of artificial network generation has been introduced by Barabási and Albert [38]. This network model is based on the phenomenon of preferential attachment, according to which vertices appear consecutively in the network and tend to join existing vertices with a strong preference for high degree vertices. The probability of selecting vertex v_i as the target of a newly created edge is proportional to v_i's degree $d(v_i)$. Scale-free networks have many interesting properties [39, 40], but from our point of view the most interesting aspect of scale-free networks is the fact that they represent a particular type of semiorder. The behavior of low-degree vertices is chaotic and random, and individual vertices are difficult to distinguish, but the structure of high-degree vertices (so-called *hubs*) imposes a well-defined topology on the network. High-degree vertices serve as bridges which facilitate communication between remote parts of the network, and their degrees are highly predictable. In other words, although a vast majority of vertices behave randomly, the order appears as soon as high-degree vertices emerge in the network.

In our second experiment, we start from a small-world network and we increment the edge rewiring probability p in each step. This time, however, we do not select the new target vertex randomly, but we use the preferential attachment principle. In the early steps, this process is still random as the differences in vertex degrees are relatively small, but at a certain point the scale-free structure emerges and as more rewiring occurs (for $p \to 1$), the network starts organizing around a subset of high-degree hubs. The intuition is that a good measure of network complexity should be able to distinguish between the initial phase of increasing the randomness of the network and the second phase where the semiorder appears.

4.2. Results and Discussion. We experiment only on artificially generated networks, using three popular network models: Erdös-Rényi random network model, Watts-Strogatz small-world network model, and Barabási-Albert scale-free network model. We have purposefully left out empirical networks from consideration, due to a possible bias which might have been introduced. Unfortunately, for empirical networks, we do not have a good method of approximating the algorithmic probability of a network. All we could do is to compare empirical distributions of network properties (such as degree, betweenness, and local clustering coefficient) with distributions from known generative models. In our

previous work [41], we have shown that this approach can lead to severe approximation errors as distributions of network properties strongly depend on values of model parameters (such as edge rewiring probability in the small-world model, or power-law coefficient in the scale-free model). Without a universal method of estimating the algorithmic probability of empirical networks, it is pointless to compare entropy and K-complexity of such networks since no baseline can be established and the results would not yield themselves to interpretation.

In our experiments we have used the `acss` R package [42] which implements the Coding Theorem Method [34, 43] and the Block Decomposition Method [35].

Let us now present the results of the first experiment. In this experiment, the edge rewiring probability p changes from 0 to 1 by 0.01 in each iteration. In each iteration, we generate 50 instances of the network consisting of $N = 100$ vertices, and for each generated network instance, we compute the following measures:

(i) Entropy and K-complexity of the adjacency matrix

(ii) Entropy and K-complexity of the Laplacian matrix

(iii) Entropy and K-complexity of the degree list

(iv) Entropy and K-complexity of the degree distribution

We repeat the experiments described in Section 4.1 for each of the 50 networks, performing the gradual change of each of these networks, and for each value of the edge rewiring probability p we average the results over all 50 networks. Since entropy and K-complexity are expressed in different units, we normalize both measures to allow for side-by-side comparison. The normalization procedure works as follows. For a given string of characters s with the length $l = |s|$, we generate two strings. The first string $s_{\min}$ consists of l repeated 0's and it represents the least complex string of the length l. The second string $s_{\max}$ is a concatenation of l uniformly selected digits and it represents the most complex string of the length l. Each value of entropy and K-complexity is normalized with respect to minimum and maximum value of entropy and K-complexity possible for a string of equal length. This allows us not only to compare entropy and K-complexity between different representations of networks, but also to compare entropy to K-complexity directly. The results of our experiments are presented in Figure 7.

We observe that traditional entropy of the adjacency matrix remains constant. This is obvious, the rewiring of edges does not change the density of the network (the number of edges in the original small-world network and the final random network or scale-free network is exactly the same), so entropy of the adjacency matrix is the same for each value of the edge rewiring probability p. On the other hand, K-complexity of the adjacency matrix slowly increases. It should be noted that the change of K-complexity is small when analyzed in absolute values. Nevertheless, K-complexity consistently increases as networks diverge from the order of the small-world model toward the chaos of random network model. A very similar result can be observed for networks represented using Laplacian matrices. Again, entropy fails to signal any change in network's complexity because the density of networks remains constant throughout the transition, and the very slight change of entropy for $p \in \langle 0, 0.25\rangle$ is caused by the change of the degree list which forms the main diagonal of the Laplacian matrix. The result for the degree list is more surprising. K-complexity of the degree list slightly increases as networks lose their ordering but remains close to 0.4. At the same time, entropy increases quickly as the edge rewiring probability p approaches 1. The pattern of entropy growth is very similar for both the transition to random network and the transition to scale-free network, with the latter characterized counterintuitively by larger entropy. In addition, the absolute value of entropy for the degree list is several times larger than for the remaining network representations (the adjacency matrix and the Laplacian matrix). Finally, both entropy and K-complexity behave similarly for networks described using degree distributions. We note that both measures correctly identify the decrease of apparent complexity as networks approach the scale-free model (when semiorder emerges) and signal increasing complexity as networks become more and more random. It is tempting to conclude from the results of the last experiment that the degree distribution is the best representation when network complexity is concerned. However, one should not forget that the degree distribution and the degree list are not lossless representations of networks, so the algorithmic complexity of degree distribution only estimates how difficult it is to recreate that distribution and not the entire network.

Given the requirements formulated at the beginning of this section and the results of the experimental evaluation, we conclude that K-complexity is a more feasible measure for constructing intuitive complexity definitions. K-complexity captures small topological changes in the evolving networks, where entropy cannot detect these changes due to the fact that network density remains constant. Also, K-complexity produces less variance in absolute values across different network representations, and entropy returns drastically different estimates depending on the particular network representation.

5. Conclusions

Entropy has been commonly used as the basis for modeling the complexity of networks. In this paper, we show why entropy may be a wrong choice for measuring network complexity. Entropy equates complexity with randomness and requires preselecting the network feature of interest. As we have shown, it is relatively easy to construct a simple network which maximizes entropy of the adjacency matrix, the degree sequence, or the betweenness distribution. On the other hand, K-complexity equates the complexity with the length of the computational description of the network. This measure is much harder to deceive and it provides a more robust and reliable description of the network. When networks gradually transform from the highly ordered to highly disordered states, K-complexity captures this transition, at least with respect to adjacency matrices and Laplacian matrices.

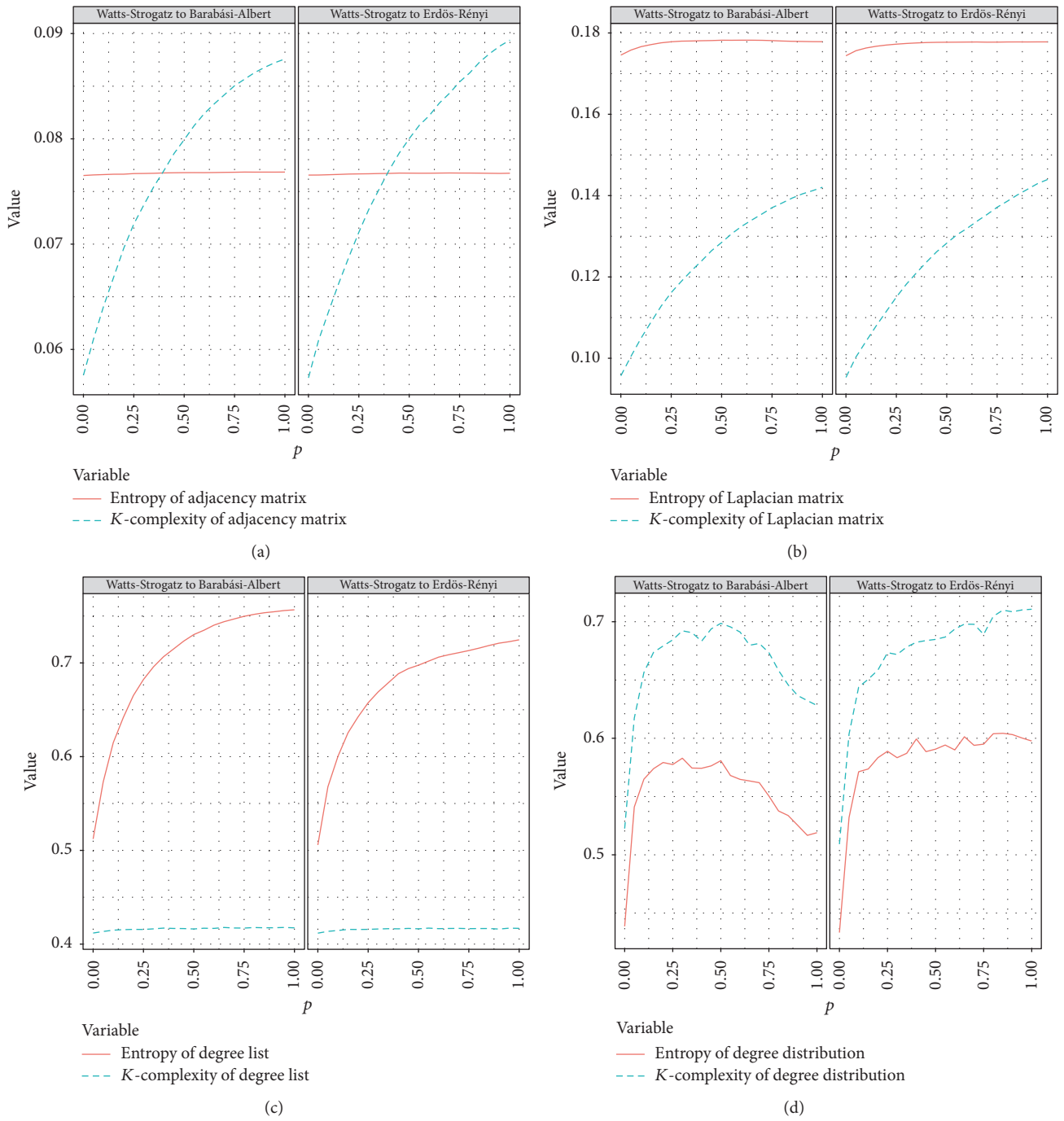


FIGURE 7: Entropy and K-complexity of (a) adjacency matrix, (b) Laplacian matrix, (c) degree list, and (d) degree distribution under gradual transition from Watts-Strogatz model to Erdös-Rényi and Barabási-Albert models.

In this paper, we have used traditional methods to describe a network: the adjacency matrix, the Laplacian matrix, the degree list, and the degree distribution. We have limited the scope of experiments to three most popular generative network models: random networks, small-world networks, and scale-free networks. However, it is possible to describe networks more succinctly, using universal network generators. In the near future, we plan to present a new method of computing algorithmic complexity of networks without having to estimate K-complexity, but rather following the minimum description length principle. Also, extending the experiments to the realm of empirical networks could prove to be informative and interesting. We also intend to investigate network representations based on various energies (Randić energy, Laplacian energy, and adjacency matrix energy) and to research the relationships between network energy and K-complexity.

Conflicts of Interest

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

Acknowledgments

The authors would like to thank Adrian Szymczak for helping to devise the degree sequence network. This work was supported by the National Science Centre, Poland, Decisions nos. DEC-2016/23/B/ST6/03962, DEC-2016/21/B/ST6/01463, and DEC-2016/21/D/ST6/02948; European Union's Horizon 2020 Research and Innovation Program under the Marie Skłodowska-Curie Grant Agreement no. 691152 (RENOIR); and the Polish Ministry of Science and Higher Education Fund for supporting internationally cofinanced projects in 2016–2019 (Agreement no. 3628/H2020/2016/2).

References

[1] L. Todd Veldhuizen, "Software libraries and their reuse: Entropy, kolmogorov complexity, and zipf's law," *Library-Centric Software Design (LCSD'05)*, p. 11, 2005.

[2] D. Bonchev and G. A. Buck, "Quantitative measures of network complexity," in *Complexity in chemistry, biology, and ecology*, Math. Comput. Chem., pp. 191–235, Springer, New York, 2005.

[3] J. Cardoso, J. Mendling, G. Neumann, and H. A. Reijers, "A discourse on complexity of process models," in *Business Process Management Workshops*, vol. 4103 of *Lecture Notes in Computer Science*, pp. 117–128, Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.

[4] J. Cardoso, "Complexity analysis of BPEL Web processes," *Software Process Improvement and Practice*, vol. 12, no. 1, pp. 35–49, 2007.

[5] A. Latva-Koivisto, 2001, Finding a complexity measure for business process models.

[6] G. M. Constantine, "Graph complexity and the Laplacian matrix in blocked experiments," *Linear and Multilinear Algebra*, vol. 28, no. 1-2, pp. 49–56, 1990.

[7] D. L. Neel and M. E. Orrison, "The linear complexity of a graph," *Electronic Journal of Combinatorics*, vol. 13, no. 1, Research Paper 9, 19 pages, 2006.

[8] J. Kim and T. Wilhelm, "What is a complex graph?" *Physica A. Statistical Mechanics and its Applications*, vol. 387, no. 11, pp. 2637–2652, 2008.

[9] M. Dehmer, F. Emmert-Streib, Z. Chen, X. Li, and Y. Shi John Wiley & Sons, Mathematical foundations and applications of graph entropy, 2016.

[10] M. Dehmer and A. Mowshowitz, "A history of graph entropy measures," *Information Sciences. An International Journal*, vol. 181, no. 1, pp. 57–78, 2011.

[11] A. Mowshowitz and M. Dehmer, "Entropy and the complexity of graphs revisited," *Entropy. An International and Interdisciplinary Journal of Entropy and Information Studies*, vol. 14, no. 3, pp. 559–570, 2012.

[12] S. Cao, M. Dehmer, and Y. Shi, "Extremality of degree-based graph entropies," *Information Sciences. An International Journal*, vol. 278, pp. 22–33, 2014.

[13] Z. Chen, M. Dehmer, and Y. Shi, "A note on distance-based graph entropies," *Entropy. An International and Interdisciplinary Journal of Entropy and Information Studies*, vol. 16, no. 10, pp. 5416–5427, 2014.

[14] K. C. Das and S. Sorgun, "On randic energy of graphs," *MATCH Commun. Math. Comput. Chem*, vol. 72, no. 1, pp. 227–238, 2014.

[15] I. Gutman and B. Zhou, "Laplacian energy of a graph," *Linear Algebra and its Applications*, vol. 414, no. 1, pp. 29–37, 2006.

[16] C. E. Shannon, *The Mathematical Theory of Communication*, The University of Illinois Press, Urbana, Ill, USA, 1949.

[17] A. N. Kolmogorov, "Three approaches to the quantitative definition of information," *International Journal of Computer Mathematics. Section A. Programming Theory and Methods. Section B. Computational Methods*, vol. 2, pp. 157–168, 1968.

[18] A. Rényi et al., "On measures of entropy and information," in *Proceedings of the 4th Berkeley Symposium on Mathematical Statistics and Probability*, pp. 547–561, University of California Press, 1961.

[19] A. Mowshowitz, "Entropy and the complexity of graphs: I. An index of the relative complexity of a graph," *The Bulletin of Mathematical Biophysics*, vol. 30, no. 1, pp. 175–204, 1968.

[20] I. Ji, W. Bing-Hong, W. Wen-Xu, and Z. Tao, "Network entropy based on topology configuration and its computation to random networks," *Chinese Physics Letters*, vol. 25, no. 11, p. 4177, 2008.

[21] M. Dehmer, "Information processing in complex networks: graph entropy and information functionals," *Applied Mathematics and Computation*, vol. 201, no. 1-2, pp. 82–94, 2008.

[22] S. Cao, M. Dehmer, and Z. Kang, "Network entropies based on independent sets and matchings," *Applied Mathematics and Computation*, vol. 307, pp. 265–270, 2017.

[23] J. Körner, "Fredman-komlós bounds and information theory," *SIAM Journal on Algebraic Discrete Methods*, vol. 7, no. 4, pp. 560–570, 1986.

[24] H. Zenil, N. A. Kiani, and J. Tegnér, "Low-algorithmic-complexity entropy-deceiving graphs," *Physical Review E*, vol. 96, no. 1, 2017.

[25] N. Sloane, "The on-line encyclopedia of integer sequences," https://oeis.org/A033308.

[26] C. Faloutsos and V. Megalooikonomou, "On data mining, compression, and Kolmogorov complexity," *Data Mining and Knowledge Discovery*, vol. 15, no. 1, pp. 3–20, 2007.

[27] J. Schmidhuber, "Discovering neural nets with low Kolmogorov complexity and high generalization capability," *Neural Networks*, vol. 10, no. 5, pp. 857–873, 1997.

[28] A. Kulkarni and S. Bush, "Detecting distributed denial-of-service attacks using Kolmogorov Complexity metrics," *Journal of Network and Systems Management*, vol. 14, no. 1, pp. 69–80, 2006.

[29] M. Li and P. M. B. Vitányii, "Kolmogorov complexity and its applications," in *Algorithms and Complexity*, Texts and Monographs in Computer Science, pp. 1–187, Springer-Verlag, New York, 2014.

[30] G. J. Chaitin, "On the length of programs for computing finite binary sequences," *Journal of the Association for Computing Machinery*, vol. 13, pp. 547–569, 1966.

[31] L. A. Levin, "Laws on the conservation (zero increase) of information, and questions on the foundations of probability theory," *Akademiya Nauk SSSR. Institut Problem Peredachi Informatsii Akademii Nauk SSSR. Problemy Peredachi Informatsii*, vol. 10, no. 3, pp. 30–35, 1974.

[32] R. J. Solomonoff, "A formal theory of inductive inference. Part I," *Information and Control*, vol. 7, no. 1, pp. 1–22, 1964.

[33] T. Rado, "On non-computable functions," *The Bell System Technical Journal*, vol. 41, pp. 877–884, 1962.

[34] F. Soler-Toscano, H. Zenil, J.-P. Delahaye, and N. Gauvrit, "Calculating Kolmogorov complexity from the output frequency distributions of small turing machines," *PLoS ONE*, vol. 9, no. 5, Article ID e96223, 2014.

[35] H. Zenil, F. Soler-Toscano, N. A. Kiani, S. Hernández-Orozco, and A. Rueda-Toicen, *A decomposition method for global evaluation of shannon entropy and local estimations of algorithmic complexity*, 2016.

[36] D. J. Watts and S. H. Strogatz, "Collective dynamics of "small-world" networks," *Nature*, vol. 393, no. 6684, pp. 440–442, 1998.

[37] P. Erdös and A. Rényi, "On the evolution of random graphs," *Publications of the Mathematical Institute of the Hungarian Academy of Sciences*, vol. 5, pp. 17–61, 1960.

[38] A.-L. Barabási and R. Albert, "Emergence of scaling in random networks," *American Association for the Advancement of Science*, vol. 286, no. 5439, pp. 509–512, 1999.

[39] A.-L. Barabási, "Scale-Free Networks: a Decade and beyond," *American Association for the Advancement of Science. Science*, vol. 325, no. 5939, pp. 412-413, 2009.

[40] M. E. Newman, "The structure and function of complex networks," *SIAM Review*, vol. 45, no. 2, pp. 167–256, 2003.

[41] T. Kajdanowicz and M. Morzy, "Using Kolmogorov Complexity with Graph and Vertex Entropy to Measure Similarity of Empirical Graphs with Theoretical Graph Models," in *Proceedings of the 2nd International Electronic Conference on Entropy and Its Applications*, p. C003, Sciforum.net.

[42] N. Gauvrit, H. Singmann, F. Soler-Toscano, and H. Zenil, "Algorithmic complexity for psychology: a user-friendly implementation of the coding theorem method," *Behavior Research Methods*, vol. 48, no. 1, pp. 314–329, 2016.

[43] J.-P. Delahaye and H. Zenil, "Numerical evaluation of algorithmic complexity for short strings: A glance into the innermost structure of randomness," *Applied Mathematics and Computation*, vol. 219, no. 1, pp. 63–77, 2012.

Hindawi
Complexity
Volume 2017, Article ID 5049836, 14 pages
https://doi.org/10.1155/2017/5049836

Research Article

On the Shoulders of Giants: Incremental Influence Maximization in Evolving Social Networks

Xiaodong Liu, Xiangke Liao, Shanshan Li, Si Zheng, Bin Lin, Jingying Zhang, Lisong Shao, Chenlin Huang, and Liquan Xiao

School of Computer, National University of Defense Technology, Changsha 410073, China

Correspondence should be addressed to Xiaodong Liu; liuxiaodong@nudt.edu.cn

Received 13 March 2017; Revised 4 July 2017; Accepted 1 August 2017; Published 28 September 2017

Academic Editor: Piotr Brodka

Influence maximization problem aims to identify the most influential individuals so as to help in developing effective viral marketing strategies over social networks. Previous studies mainly focus on designing efficient algorithms or heuristics on a static social network. As a matter of fact, real-world social networks keep evolving over time and a recalculation upon the changed network inevitably leads to a long running time. In this paper, we propose an incremental approach, IncInf, which can efficiently locate the top-K influential individuals in evolving social networks based on previous information instead of calculation from scratch. In particular, IncInf quantitatively analyzes the influence spread changes of nodes by localizing the impact of topology evolution to only local regions, and a pruning strategy is further proposed to narrow the search space into nodes experiencing major increases or with high degrees. To evaluate the efficiency and effectiveness, we carried out extensive experiments on real-world dynamic social networks: Facebook, NetHEPT, and Flickr. Experimental results demonstrate that, compared with the state-of-the-art static algorithm, IncInf achieves remarkable speedup in execution time while maintaining matching performance in terms of influence spread.

1. Introduction

The increasing popularity of online social network has promoted the diffusion of information, opinions, adoption of new products, and so forth and provided great opportunities for intelligent viral marketing. To benefit best from the word-of-mouth effect, influence maximization (IM) is one fundamental and important problem that aims to identify a small set of influential individuals so as to develop effective viral marketing strategies to maximize the influence over a given social network [1]. As a matter of fact, real-world social networks keep evolving over time. For example, in Facebook, new people might join, while old ones might withdraw, and people might make new friends with each other. Moreover, real-world social networks are evolving in a rather surprising speed; it is reported that as much as 1 million new accounts are created in Twitter every day [2]. Such massive evolution of network topology, on the contrary, may lead to a significant transformation of the network structure, thus raising a natural need of efficient reidentification.

Existing researches and solutions on influence maximization focus mainly on developing effective and efficient algorithms on a given static social network. Although one could possibly run any of the static influence maximization methods, such as [3–6], to find the new top-K influential individuals when the network is updated, this approach has some inherent drawbacks that cannot be neglected: (1) the running time of a specific static method can be extremely long and unacceptable, especially on large-scale social networks, and (2) whenever the network topology is changed, we need to recalculate the influence spreads for all the nodes, which leads to very high costs. Can we quickly and efficiently identify the influential nodes in evolving social networks? Can we incrementally update the influential nodes based on previously known information instead of frequently recalculating from scratch?

Unfortunately, the rapidly and unpredictably changing topology of a dynamic social network poses several challenges in the reidentification of influential users, which we list as follows. On one hand, the interconnections between edges in real-world social graphs are rather complicated; as a result, even one small change in topology may affect the influence spreads of a large number of nodes, not to mention the massive changes in large-scale social networks. It is very difficult to efficiently compute the changes of influence spreads for all the nodes after the evolution. On the other hand, since there are a great number of nodes in large-scale social networks, how to effectively limit the range of potential influential nodes and reduce the amount of calculation to the maximum is a very challenging problem.

To well address these challenges, we investigate the dynamic characteristics exhibited during the evolution of real-world social networks. Through tests on three real-world dataset traces, Facebook, NetHEPT, and Flickr, we observe that, first, the growth of social network is mainly based on the preferential attachment principle [7]; that is, the new-coming edges prefer to attach to nodes with higher degree, which naturally leads to the "rich-get-richer" phenomena; and, second, the top-K influential nodes are mainly selected from those high-degree nodes. Inspired by such observations, we know that the influence changes of some nodes will have no impact on the top-K selection and thus can be pruned to reduce the amount of calculation. Motivated by this, we propose IncInf, an incremental method to identify the top-K influential nodes in evolving social networks instead of recalculating from scratch, thus significantly improving the efficiency and scalability to handle extraordinarily large-scale networks. To sum up, the main contributions of IncInf are as follows.

First, we design an efficient approach to quantitatively analyze the influence spread changes from network topology evolution by adopting the idea of localization. A tunable parameter is provided for tradeoff between efficiency and effectiveness.

Second, we propose a pruning strategy that could effectively narrow the search space into nodes only experiencing major increases or with high degrees based on the changes of influence spread and the previous top-K information.

Third, we conduct extensive experiments on three dynamic real-world social networks. Compared with the state-of-the-art static algorithm, IncInf achieves remarkable speedup in execution time while providing matching influence spread. Moreover, IncInf provides better scalability to scale up to extraordinarily large-scale networks.

A preliminary version of this paper appears in [8], where we presented the basic idea of IncInf algorithm. In this paper, we make the following additional contributions. First, we add corresponding experiments to compare IncInf with IMM [9] in terms of influence spread and running time. Second, we test the effect of our pruning strategy to demonstrate its effectiveness. Third, we add a new experiment to evaluate the sensitivity of the localization parameter θ and pruning threshold η in terms of influence spread and running time.

The remainder of this paper is organized as follows. In Section 2, we show the related work. Section 3 presents related preliminaries and problem definition. Section 4 shows the structural evolution characteristics of dynamic social networks that we observe from three datasets: Facebook, NetHEPT, and Flickr. Section 5 details the design of our incremental algorithm IncInf. The performance of IncInf is evaluated by comprehensive experiments in Section 6. We conclude the paper in Section 7.

2. Related Work

Influence maximization on static networks has attracted a great deal of attention. The hill-climbing greedy algorithm proposed by Chen et al. suffers from low efficiency, and many efficient algorithms have been proposed recently to address this problem. Leskovec et al. [5] exploit the submodularity of influence spread function and develop an optimized greedy algorithm, CELF, which is much faster than basic greedy algorithm. Chen et al. [3] propose MixGreedy, which computes the influence spread for each seed set in one single simulation and incorporates the CELF optimization. MIA [4] uses local arborescence structures of each node to approximate the influence spread, thereby gaining efficiency by restricting computations and updates only to the local regions. However, MIA only considers static networks, while in this paper we specifically design an incremental algorithm for evolving social networks. Recently, Wang et al. [10] propose a Community Greedy Algorithm (CGA) that took community property into account. Goyal et al. propose CELF++ [11], which further exploits the property of submodularity of the spread function to avoid unnecessary recomputations of marginal gains and considerably improves the efficiency of CELF algorithm. IRIE [12] is also a heuristic proposed by Jung et al., which incorporates influence ranking algorithm with influence estimation method to achieve scalability. Liu et al. [13] design a new framework to accelerate the influence maximization by leveraging the parallel processing capability of GPU. Chen et al. [14] develop a community-based framework to tackle the influence maximization problem with an emphasis on the efficiency issue. Tang et al. [9] design a martingale approach that tries to find the top-K nodes in near-linear time. And, in [15], Wang proposes a method to obtain each node's marginal contribution by Owen value and deploys it in online terrorist network analysis. Lu et al. study the complexity of the influence maximization problem in deterministic linear threshold model in [16]. In [17], Lu et al. show how to efficiently estimate the influence spread for influence maximization under the linear threshold model. In [18], Nguyen and Zheng focus on the budgeted influence maximization (BIM) problem that aims to select seed nodes at a total cost no more than the fixed budget. Han et al. [19] study the influence maximization in timeliness networks and design a novel algorithm that incorporates time delay for timeliness and opportunistic selection for acceptance ratio. Liu et al. [20] propose the time-constrained influence maximization problem and develop a set of parallel algorithms for achieving more time savings. Pei et al. [21] take

advantage of the concept of subcritical path and propose CI-TM, a collective influence algorithm of optimal percolation for second-order transitions.

The influence maximization problem on dynamic social networks still remains largely unexplored to date. Habiba et al. [22] and Michalski et al. [23] propose a dynamic social network model that is different from ours. In their proposal, the network keeps evolving during the process of influence propagation, and their goal is to find the top-K influential nodes over such a dynamic network. When compared to [22, 23], our work is based on snapshot graph model and our goal is to incrementally identify top-K influential nodes based on the topology changes of two adjacent snapshots. Chen et al. [24] extend the IC model to incorporate the time delay aspect of influence diffusion among individuals in social networks and consider time-critical influence maximization, in which one wants to maximize influence spread within a given deadline. Meanwhile, in [25], the authors consider a continuous time formulation of the influence maximization problem in which information or influence can spread at different rates across different edges. Aggarwal et al. [26] try to discover influential nodes in dynamic social networks and they design a stochastic approach to determine the information flow authorities with the use of a globally forward approach and a locally backward approach. Their influence model and target are different from ours. Zhuang et al. [27] argue that the evolution of online social network could not be fully observed and design a probing strategy so that the actual influence diffusion process can be best uncovered with the probing nodes. Tong et al. [28] mainly focus on the fact that the diffusion processes in real-world dynamic social networks have many aspects of uncertainness and propose a method that selects seed users in an adaptive manner.

3. Preliminaries and Problem Statement

In this section, we illustrate the definition of social network and the influence diffusion model that we will use throughout the paper and then give the problem definition of influence maximization in evolving networks.

3.1. Preliminaries on Influence Maximization

Social Network. A social network is formally defined as a directed graph $G = (V, E, P)$, where node set $V = \{v_1, v_2, \ldots, v_n\}$ denotes entities in the social network. Each node can be either active or inactive and will switch from being inactive to being active if it is influenced by other nodes. Edge set $E \subset V \times V$ is a set of directed edges representing the relationship between different users. Take Twitter as an example. A directed edge (v_i, v_j) will be established from node v_i to v_j if v_i is followed by v_j, which indicates that v_j may be influenced by v_i. P denotes the influence probability of edges; each edge $(v_i, v_j) \in E$ is associated with an influence probability $p(v_i, v_j)$ defined by function $p : E \to [0, 1]$. If $(v_i, v_j) \notin E$, then $p(v_i, v_j) = 0$.

(1) Initialize $S = \emptyset$
(2) **for** $i = 1$ to K **do**
(3) Select $v = \arg\max_{v_i \in (V \setminus S)} (\sigma(S \cup v_i) - \sigma(S))$
(4) $S = S \cup \{v\}$
(5) **end for**

Algorithm 1: Basic greedy algorithm.

Independent Cascade (IC) Model. IC model is a popular diffusion model that has been well studied in [3, 6, 10, 29]. Given an initial set S, the diffusion process of IC model works as follows. At step 0, only nodes in S are active, while other nodes stay in the inactive state. At step t, for each node v_i that has just switched from being inactive to being active, it has a single chance to activate each currently inactive neighbor v_j and succeeds with a probability $p(v_i, v_j)$. If v_i succeeds, v_j will become active at step $t + 1$. If v_j has multiple newly activated neighbors, their attempts in activating v_j are sequenced in an arbitrary order. Such a process runs until no more activations are possible [29]. We use $\sigma(S)$ to denote the influence spread of the initial set S, which is defined as the expected number of active nodes at the end of influence propagation.

Basic Greedy Algorithm. Domingos and Richardson [1, 30] first introduced the influence maximization problem on static networks in 2001. In [29], Kempe et al. propose a basic hill-climbing greedy algorithm as shown in Algorithm 1. The proposed greedy algorithm works in K iterations, starting with an empty set S (line (1)). In each iteration, a node v_i that brings the maximum marginal influence spread $\sigma_S(v_i) = \sigma(S \cup v_i) - \sigma(S)$ is selected to be included in S (lines (3) and (4)). The process ends when the size of S reaches K (line (2)). However, this algorithm has a serious efficiency drawback due to the compute-intensive influence spread calculation. Several recent studies [3–6, 10, 12, 31–35] aimed at addressing this efficiency issue.

3.2. Formal Definition of IM Problem in Evolving Networks.

This paper distinguishes itself from previous works by considering the dynamic nature of online social networks. As a matter of fact, the real-world social networks are not wholly static but keep evolving gradually over time. The evolution of large social networks has raised new sets of questions; among them one interesting yet challenging problem is how to quickly identify the top-K influential users when the topology of the network is changed.

To solve such a problem, we define an evolving network $\zeta = (G^0, G^1, \ldots, G^t)$ as a sequence of network snapshots evolving over time, where $G^t = (V^t, E^t, P^t)$ is the network snapshot at time t. $\Delta G^t = (\Delta V^t, \Delta E^t, \Delta P^t)$ denotes the structural change of network graph G^t. Obviously, we have $G^{t+1} = G^t \cup \Delta G^t$. And the influence maximization problem is defined as follows:

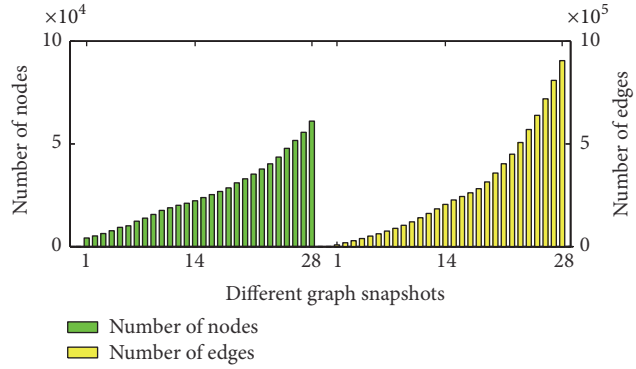


Figure 1: Number of nodes and edges per month of the Facebook dataset.

Given. One has the social network G^t at time t, the top-K influential nodes S^t in G^t, and the structural evolution ΔG^t of graph G^t.

Objective. The aim is to identify the influential nodes $S^{t+1} \subset V^{t+1}$ of size K in G^{t+1} at time $t+1$, such that the influence spread $\sigma(S^{t+1})$ is maximized at the end of influence diffusion.

4. Observations of Social Network Evolution

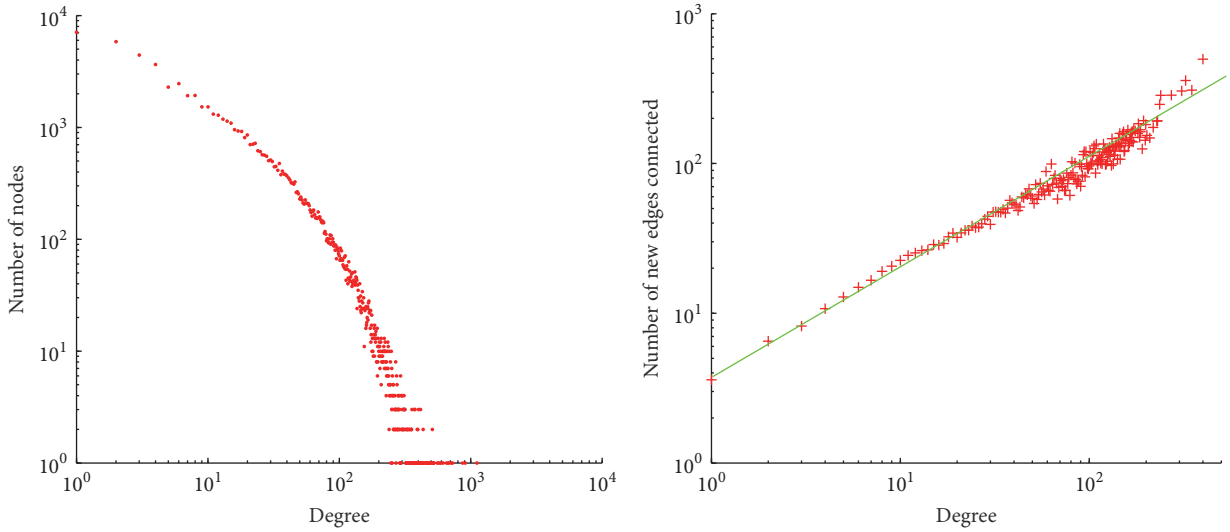
In this section, we study some patterns of social network evolution. The numbers of nodes and edges are firstly investigated in Section 4.1 to examine the growth of users and interconnections over time. Then, we look into the degree distribution of nodes and the preferential attachment rule for new edges in Section 4.2. We further examine the relation between the influence and the degree of node in Section 4.3. We study three network traces, namely, Facebook, NetHEPT, and Flickr, whose detailed description can be found in Section 6. Here we only show the results on Facebook, since the evolution trends on the other datasets are qualitatively similar and thus were omitted.

4.1. How Fast Does the Network Evolve? Nodes and edges are the basic elements of the social network topology. In this subsection, we use the numbers of nodes and edges to examine the growth of users and interconnections over time. Figure 1 illustrates the numbers of nodes and edges over the entire trace period on the Facebook dataset; we take a snapshot per month. From Figure 1, we observe a linear increase in the number of nodes, which indicates a steady number of new users who joined the network per month. Meanwhile, in terms of edges, the number goes up almost exponentially. The number of edges after 14 months is 25.6x of that in the initial graph while the number rises to 112.9x after 28 months. Such rapid growth of nodes and edges raises a natural need to efficiently find the most influential nodes after the topology evolution.

4.2. What Is the Pattern of Network Topology Evolution? Understanding the pattern of the network topology evolution is of primary importance to design efficient influence maximization algorithms for evolving social networks. In this subsection, we further investigate the degree distribution of nodes and the preferential attachment rule [7, 36, 37] for new coming edges. Figure 2(a) shows the degree distribution of the Facebook final graph in log-log scale. As expected, it mainly follows the well-known power-law distribution. A large percent of the users have only a small number of links with other users, while there exist some "hub" nodes with extremely large number of connections. This is consistent with the real-world networks.

We also study the preferential attachment rule or, in other words, the "rich-get-richer" rule [38], which postulates that when a new node joins the network, it creates a number of edges, where the destination node of each edge is chosen proportional to the destination's degree. This means that new edges are more likely to connect to nodes with high degree than ones with low degree. This is reasonable in reality; Lady Gaga gains 30,000 new followers on average every day [39], which can never image for any common individual. The results on the Facebook dataset are demonstrated in Figure 2(b), where x-axis is the degree of different nodes and y-axis is the average number of new edges attached to nodes of different degree. Note that both x-axis and y-axis are in log scale. From Figure 2(b), we can see that the degree of users in Facebook is linearly correlated with the number of new links created. This suggests that high-degree nodes get super preferential treatment. Consequently, the influence spread change should be considerably great for the influential nodes, while there may be only small or even no change for ordinary people.

4.3. What Is the Relation between Influence and Degree? Examining the relation between the influence and the degree of node can help us understand the effect of degree changing on the influence spread of nodes. For this reason, we run the static MixGreedy algorithm [3] on the final graph and identify the top-50 influential nodes. The results on the Facebook dataset are illustrated in Figure 3, where x-axis is the rank of degrees of different nodes (we only show the top 150). Obviously, all the selected influential nodes have a large degree. In particular, among the 50 nodes, 48 nodes rank in top 100 of the whole 61,096 nodes in terms of degree, and the



(a) Degree distribution (b) Preferential attachment

Figure 2: Degree distribution and preferential attachment on Facebook.

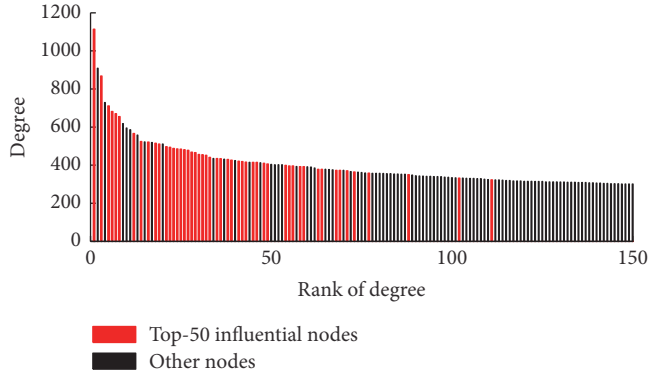


Figure 3: The relation between the influence spread and the degree in Facebook.

other two nodes rank 102 and 111, respectively. Meanwhile, on the NetHEPT and Flickr datasets, the top-50 influential nodes are selected from the top 1.79% and 0.84% nodes in degree, respectively. This demonstrates that the top-K influential nodes are mainly selected from those with large degrees. However, it is worth noting that the top-K influential nodes in influence maximization are usually not the top-K nodes ranking in degree, since the influence spreads of different nodes may overlap with each other.

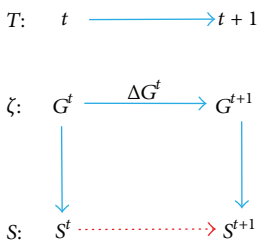


Figure 4: IncInf design.

5. IncInf Design

In this section, we present the detailed design of IncInf, an incremental approach to solve the influence maximization problem on dynamic social networks. The main idea of IncInf is to take full use of the valuable information that is inherent in the network structural evolution and previous influential nodes so as to substantially narrow the search space of influential nodes. In this way, IncInf can significantly reduce the computation complexity and improve the efficiency. Figure 4 briefly illustrates the general idea of IncInf in dynamic social networks. The top-K influential nodes S^{t+1} of G^{t+1} at time $t + 1$ are incrementally identified based on the previous influential nodes S^t at time t and the structural change ΔG^t from G^t to G^{t+1}. In particular, we design an efficient method to quantitatively analyze the impact of different structural changes on the influence spread of nodes by adopting the idea of localization (Section 5.2) and propose a pruning strategy to reduce the number of potential influential nodes

(Section 5.3). We first describe six types of basic operation of topology evolution in dynamic networks in Section 5.1.

5.1. Basic Operations of Topology Evolution. The evolution of social network, when reflected into its underlying graph, can be summarized into six categories, which are inserting or removing a node, introducing or deleting an edge, and increasing or decreasing the influence probability of an edge. We denote the six types of topology change as *addNode*, *removeNode*, *addEdge*, *removeEdge*, *addWeight*, and *decWeight*. The detailed descriptions and their effects on influence spread are shown in Table 1.

It should be noted that only after the *addNode* operation can node u establish links *(addEdge)* or sever links *(removeEdge)* with other nodes, and node u can only be removed when all its associated edges are deleted. Moreover, the weight operation can be equivalently decomposed into two edge operations. For example, $addWeight\,(u, v, \Delta w)$ can be divided into $removeEdge\,(u, v)$ and $addEdge\,(u, v, w + \Delta w)$, supposing that the previous weight of edge (u, v) is w.

5.2. Influence Spread Changes. As discussed above, whenever an edge (u, v) is introduced into or removed from the social network, the influence spread of all the nodes that can reach node u may be changed. However, as a matter of fact, the real-world social networks exhibit small-world network characteristics and the connections between nodes are highly complicated. As a result, even one small change in topology, such as an edge addition or removal, may affect the influence spread of a large number of nodes, thus introducing massive recalculations. In order to reduce the amount of computation, we design an approach to efficiently calculate the changes on the influence spread of nodes, which adopts the localization idea [4] and tries to restrict the influence spread to the local regions of nodes.

The main idea of localization is to use the local region of each node to approximate its overall influence spread. In particular, we use the maximum influence path to approximate the influence spread from node u to v. Here the maximum influence path $\mathrm{MIP}\,(u, v, G)$ from node u to v in graph G is defined as the path with the maximum influence probability among all the paths from node u to v and can be formally described as follows:

$$\mathrm{MIP}\,(u, v, G) = \arg \max_{p \in P(u,v,G)} \{\mathrm{prob}\,(p)\}, \tag{1}$$

where $\mathrm{prob}\,(p)$ denotes the propagation probability of path p and $P(u, v, G)$ denotes all the paths from node u to v in graph G. For a given path $p = \{u_1, u_2, \ldots, u_m\}$, the propagation probability of path p is defined as follows:

$$\mathrm{prob}\,(p) = \prod_{i=1}^{m-1} p\,(u_i, u_{i+1}). \tag{2}$$

Moreover, an influence threshold θ is set to tradeoff between accuracy and efficiency. During the propagation process, we only consider paths whose influence probability is larger than θ while ignoring those with probability smaller than θ. By doing this, the influence is effectively restricted to the local region of each node.

Similarly, in our proposal, we localize the impact of topology changes on influence spread into local regions and thus reduce the amount of computation. Among six types of topology change, *addNode* (or *removeNode*) is the most straightforward, since it simply sets the influence spread of the node to 1 (or 0); *addWeight* and *decWeight* as well as *removeEdge* are methodologically similar to *addEdge*. Consequently, in the following, we take *addEdge* as an example to show which nodes' influence spreads need to be updated and how to determine those changes when a new edge is added to the graph.

Consider the case when a new edge $e = (u, v, w)$ is introduced between two existing nodes u and v. We denote the graphs before and after such a topology change as G^t and $G^{t'}$, and the current seed set is S. The detailed algorithm is described in Algorithm 2. According to the principle of localization [4], if the propagation probability w is smaller than the specified threshold θ or not bigger than the probability of $\mathrm{MIP}\,(u, v, G^t)$, edge e can be simply neglected and there is no need to update any node's influence spread (lines (1)–(3)). Otherwise, the newly added edge e would become $\mathrm{MIP}\,(u, v, G^{t'})$. As a result, each node i whose maximum influence path to u has an influence probability larger than θ is likely to experience a rise in terms of influence spread (line (4)) because node i may influence more nodes through the new edge e. So, we then check the probability of the maximum influence path from i to v and its successors in G^t and $G^{t'}$. Based on the two probabilities, we divide the problem into two small cases.

The first case is when the probability of maximum influence path from i to j in G^t is smaller than θ, while that in $G^{t'}$ is larger than θ (lines (5)-(6)). Here j denotes the node whose probability of $\mathrm{MIP}\,(v, j, G^t)$ is larger than θ. In such a case, node i builds a new path to j through the new edge e, which increases the influence spread of i by $\mathrm{prob}\,(\mathrm{MIP}\,(i, j, G^{t'})) \times (1 - \mathrm{prob}\,(j, S))$ (line (7)). Here $\mathrm{prob}\,(j, S)$ is the probability of the fact that node j is influenced by the current seed set S, which is defined as follows:

$$\mathrm{prob}\,(j, S) = \begin{cases} 1, & \text{if } j \in S \\ 1 - \prod_{w \in n(j)} 1 - \mathrm{prob}\,(w, S) \cdot p\,(w, j), & \text{if } j \notin S. \end{cases} \tag{3}$$

Here $n(j)$ denotes the in-neighbour set of j.

The second case is when the probability of maximum influence path from i to j is larger than θ in both G^t and $G^{t'}$ (lines (9)–(11)). In this case, the influence increase of node i is $(\mathrm{prob}\,(\mathrm{MIP}\,(i, j, G^{t'})) - \mathrm{prob}\,(\mathrm{MIP}\,(i, j, G^t)))\times(1 - \mathrm{prob}\,(j, S))$.

We treat the network dynamics from G^t to G^{t+1} as a finite change stream $c_1, c_2, \ldots, c_i, \ldots$, where each change c_i is one of the six topology changes we described previously. When all the changes in the change stream are processed, we can obtain the influence spread change for all the nodes. Next, we will

Table 1: Details of six types of basic operation.

Operation	Description	Impact on influence spread
$addNode(u)$	Add a new node u to the current network	The influence spread of u is set to 1
$removeNode(u)$	Delete an existing node u from the network	The influence spread of u is set to 0
$addEdge(u, v, w)$	Introduce a new edge (u, v) with $p(u, v) = w$	The influence spread of all the nodes that can reach u may be increased
$removeEdge(u, v)$	Remove an existing edge (u, v) from the network	The influence spread of all the nodes that can reach u may be decreased
$addWeight(u, v, \Delta w)$	Increase $p(u, v)$ by Δw	The influence spread of all the nodes that can reach u may be increased
$decWeight(u, v, \Delta w)$	Reduce $p(u, v)$ by Δw	The influence spread of all the nodes that can reach u may be decreased

Input: a new edge $e = (u, v, w)$, graph G^t.
Output: The influence spread changes of nodes in $G^{t'}$.
(1) **if** $w < \theta$ or $w \leq$ prob (MIP (u, v, G^t)) **then**
(2) return;
(3) **end if**
(4) **for** each node i with prob (MIP $(i, u, G^t)) > \theta$ **do**
(5) **for** each node j with prob (MIP $(v, j, G^t)) > \theta$ **do**
(6) **if** prob (MIP $(i, j, G^t)) < \theta$ and prob (MIP $(i, j, G^{t'})) > \theta$ **then**
(7) $deltaInf[i] += \text{prob}\left(\text{MIP}\left(i, j, G^{t'}\right)\right) \times (1 - \text{prob}(j, S))$
(8) **end if**
(9) **if** prob (MIP $(i, j, G^t)) > \theta$ and prob (MIP $(i, j, G^{t'})) > \theta$ **then**
(10) $deltaInf[i] += \left(\text{prob}\left(\text{MIP}\left(i, j, G^{t'}\right)\right) - \text{prob}\left(\text{MIP}\left(i, j, G^t\right)\right)\right) \times (1 - \text{prob}(j, S))$
(11) **end if**
(12) **end for**
(13) **end for**

Algorithm 2: Edge addition.

show how to effectively find the top-K influential nodes in the new graph G^{t+1} based on these influence spread changes and the previous influential nodes information.

5.3. Potential Top-K Influential Users Identification. Inspired by the observations of Section 4, we design a pruning strategy to reduce the search space of potential influential nodes in this subsection. It is assumed that we only know which are the top-K influential nodes in graph G^t, but their detailed influence spreads are beyond our knowledge. The reasons are mainly twofold. First, several influence maximization algorithms, such as DegreeDiscount [3], do not calculate the influence spread information to identify influential users so that such information is unavailable. Second, even though this information is ready, storing it will cost as much as $O(nK)$ memory space, where n is the number of nodes in G^t. Since real-world social networks are typically of large scale, this will introduce serious storage overhead and directly affect the scalability.

From the preferential attachment rule, we know that the influence spread changes of those high-degree nodes should be much greater than the ordinary nodes. Moreover, according to the power-law distribution, such high-degree nodes only account for a small part of the whole nodes. Consequently, we can pick out nodes only experiencing major increases or with high degrees because these nodes are of great potential to become the top-K influential nodes in G^{t+1}. Then we only calculate the actual influence spreads for these selected nodes while ignoring the others. In this way, a large percent of nodes are pruned and the search space is largely narrow. It should be noted that a smart pruning strategy is of key importance, since a poor selection might either affect the efficiency or reduce the accuracy in terms of influence spread. We describe the details of our pruning strategy as follows:

(1) In the ith iteration, if the influence spread of the previous influential node S_i^t increases in G^{t+1}, the chosen nodes are those with a larger influence spread change than $deltaInf[S_i^t]$.

In most cases, the influential nodes will attract a great number of new nodes and establish new links. Thus, their influence spreads will increase drastically. In such a case, it is impossible for the nodes

Input: G^t, S^t, and G^{t+1}.
Output: the top-K influential nodes S^{t+1} in G^{t+1}.
(1) Initialize $S^{t+1} = \emptyset$;
(2) **for** $i = 1$ to K **do**
(3) **for** each topology change c_j from G^t to G^{t+1} **do**
(4) calculate the influence spread change $deltaInf[\cdot]$;
(5) **end for**
(6) select a set of potential nodes as pn according to pruning strategy;
(7) **for** each node $v_l \in pn$ **do**
(8) calculate the marginal influence spread $\sigma_{S^{t+1}}(v_j)$;
(9) **end for**
(10) select $v_{\max} = \arg\max_{v_j \in pn}(\sigma_{S^{t+1}}(v_j))$;
(11) $S = S \cup v_{\max}$;
(12) **end for**

Algorithm 3: IncInf.

whose influence spread changes are smaller than the influential nodes to become the most influential nodes in G^{t+1}. Therefore, when the influence spread of the previous influential nodes increases, we only select those whose influence spread changes are larger than the influential nodes in G^t. According to the preferential attachment rule, such a pruning method can greatly narrow the search space and reduce the amount of computation.

(2) In the ith iteration, if the influence spread of the previous influential node S_i^t decreases in G^{t+1}, in addition to qualification 1, the nodes are further selected to hold a sufficiently large degree or experience a sufficiently great increase. In order to formally define "large degree" and "great increase," here we set a threshold η for tradeoff between running time and influence spread. Here the nodes with sufficiently large degrees (or great increase) are defined as the set of nodes v_j whose degree (or degree increase ratio) is among the top η percent of all nodes in G^{t+1}. The degree increase ratio of v_j is defined as $\text{degree}_j^{t+1}/\text{degree}_j^t$, where degree_j^t denotes the degree of node v_j in graph G^t. Experimental results in Section 6 will demonstrate that 5% may stand as a good tradeoff between running time and influence spread.

It should be noted that although the case where the influence spread of a previous influential node decreases during the evolution rarely happens, we consider it here for completeness. In this case, except for qualification 1, we further select nodes because the number of nodes satisfying qualification 1 is relatively large, which leads to mass computation. Meanwhile, in reality, a node with small degree has only very low probability to become an influential node. In order to select only the most potential nodes, we refine the requirement and additionally select the nodes with large degree and large increase. Consequently, the search space is strictly circumscribed and the computational complexity is greatly reduced.

After the potential nodes are selected, we calculate the actual influence spread of these nodes in G^{t+1} and select the one with the maximum influence spread in each iteration. Algorithm 3 outlines the design of our proposed algorithm IncInf. IncInf iterates for K round (line (2)) and in each round selects one node, providing the maximum marginal influence spread. Lines (3)–(5) calculate the influence spread change of each node caused by the topology evolution. Nodes with great potential to become top-K influential are selected (line (6)) and their influence spreads are computed in G^{t+1} (lines (7)–(9)). And then the node providing the maximal marginal gain will be selected and added to the set S^{t+1} (lines (10)-(11)).

6. Experiments

In this section, we present the experimental results of our algorithm on identifying top-K influential nodes in dynamic social networks. We examine two metrics, running time and influence spread, for evaluating the effectiveness as well as the execution efficiency of different algorithms. The experimental results are detailed in Sections 6.2, 6.3, and 6.4.

6.1. Experimental Setup. We choose three real-world social networks: Facebook social network, NetHEPT citation network, and Flickr social network (Table 2 summarizes the statistical information of the datasets):

(i) Facebook: this dataset is the friendship relationship network among New Orleans regional network on Facebook, spanning from September 2006 to January 2009 [40]. There are more than 60 K users connected together by as much as 1.5 M links in the social network. 41.4% of these edges contain no time information and are thus discarded. In our experiments, the nodes and links from September 2006 to April 2007 are used as the first snapshot and then network snapshots are recorded every 3 months

Table 2: Summary information of the real-world social networks.

Datasets	Nodes			Edges		
	Initial number	Final number	Growth	Initial number	Final number	Growth
Facebook	12,364	61,096	394%	73,912	905,665	1125%
NetHEPT	5,802	29,555	409%	57,765	352,807	511%
Flickr	1,620,392	2,570,535	58.6%	17,034,807	33,140,018	94.5%

(ii) NetHEPT: this is an academic citation network [41] extracted from "High Energy Physics-Theory" section of the arXiv over the period from 1992 to 2003 and covers the citations within a dataset of 28 K papers with 352 K edges. In our experiments, the citation links of the first three years (i.e., from 1992 to 1994) are considered as the basic graph and the network snapshots are recorded once a year

(iii) Flickr: this dataset [42] contains the user-to-user links crawled from the Flickr social network daily over the period from 2 November 2006 to 3 December 2006 and again from 3 February 2007 to 18 May 2007, representing a total of 104 days of growth. There are totally 2.5 M Flickr users and 33 M links. During this period of observation, over 9.7 million new links are formed and over 950,000 new users joined the network. In our experiments, we use the network before 2 November 2006 as the basic graph and another five snapshots are recorded on 3 December, 3 February, 3 March, 3 April, and 18 May

We compare our algorithm with five static algorithms: *MixGreedy*, *ESMCE, MIA, IMM*, and *Random*. MixGreedy is an improved greedy algorithm on the IC model proposed by Chen et al. in [3]. ESMCE is a power-law exponent supervised estimation approach designed by Liu et al. in [6]. MIA is a heuristic that uses local arborescence structures of each node to approximate the influence propagation [4]. IMM is an algorithm designed by Tang et al. based on the martingales estimation techniques and is able to run in near-linear time [9]. Random is a basic heuristic that randomly selects K nodes from the whole datasets.

The propagation probability of the IC model is selected randomly from 0.1, 0.01, and 0.001 for each network snapshot. The parameters of the evaluated algorithms are set as suggested by their authors. For IMM, the parameters ε and ι are set to 0.5 and 1, respectively, in the experiments. For the setting of θ in MIA, as stated in Section 6.4, the knee point of the influence spread curve can serve as a good tuning point of θ for tradeoff between efficiency and effectiveness. For example, the knee points of influence spread curves of Facebook and NetHEPT are 1/180 and 1/200, respectively. The value of θ for a particular dataset in the experiment is selected according to the knee point to get the best tradeoff. For ESMCE, the confidence level is set to 95% and the maximal Monte Carlo error threshold is set to 5%. For IncInf, the value of θ is set similar to that of MIA, that is, the knee point of the influence spread curve, while the value of η is set to 5% as suggested in Section 6.4 for tradeoff between running time and influence spread.

6.2. Efficiency Study. In this subsection, the efficiency of our proposed algorithm is studied and compared with corresponding static algorithms, MixGreedy and MIA, through experiments on the Facebook, NetHEPT, and Flickr datasets. The experiments are conducted on a PC with Intel Core i7 920 CPU @2.67 GHz and 6 GB RAM. The running times of four algorithms are measured by selecting 50 seeds from the whole dataset.

The time costs of different algorithms are illustrated in Figure 5, where we record the total time cost for each snapshot of the three datasets. Since incremental and static algorithms have the same time cost in the initial snapshot, they are omitted in the figure. The experimental results show that the time costs of our algorithm on each snapshot are obviously less than those of static algorithms. Obviously, MixGreedy takes the longest time among four kinds of influence maximization algorithms. It takes MixGreedy more than as much as 6 hours to identify the top-50 influential nodes on the final NetHEPT dataset, while the time is even longer on the larger dataset Facebook. Moreover, MixGreedy is not feasible to run on the largest dataset Flickr due to the unbearably long running time. ESMCE, benefiting from its sampling estimation method, runs much faster than MixGreedy, but it still takes as much as 3511 seconds on average to run on the five snapshots of Flickr. Compared with two greedy algorithms, the heuristic MIA and the martingales approach IMM perform much better. It only takes MIA 23.8 seconds (IMM 8.2 seconds, resp.) to run on the final Facebook graph. When running on the Flickr dataset with as much as 2.5 M nodes and 33 M edges, however, the speedup of IMA is still far from being satisfactory, since it still needs more than 45 minutes to finish. Comparatively, IMM performs better than MIA and it takes IMM nearly 10 minutes to locate the influential nodes. Meanwhile our proposed algorithm, IncInf, outperforms all the static algorithms in terms of efficiency. In particular, IncInf is almost four orders of magnitude faster than the MixGreedy algorithm on the Facebook dataset. Compared with the MIA heuristic, the speedup of IncInf is 8.41x and 6.94x on the Facebook and NetHEPT datasets, respectively. What is more, when applied on the largest dataset Flickr, IncInf can achieve as much as 20.65x speedup on average. The efficiency of IncInf is only slightly better than IMM on our small dataset NetHEPT, but it performs considerably better than IMM on Flickr dataset: the running time of IncInf is only 40% on average of IMM on each snapshot. This is because IncInf only computes the incremental influence spread changes and adaptively identifies the influential nodes based on the previous influential nodes and the current influence spread changes. The experimental results clearly validate the efficiency advantage of our incremental algorithm IncInf.

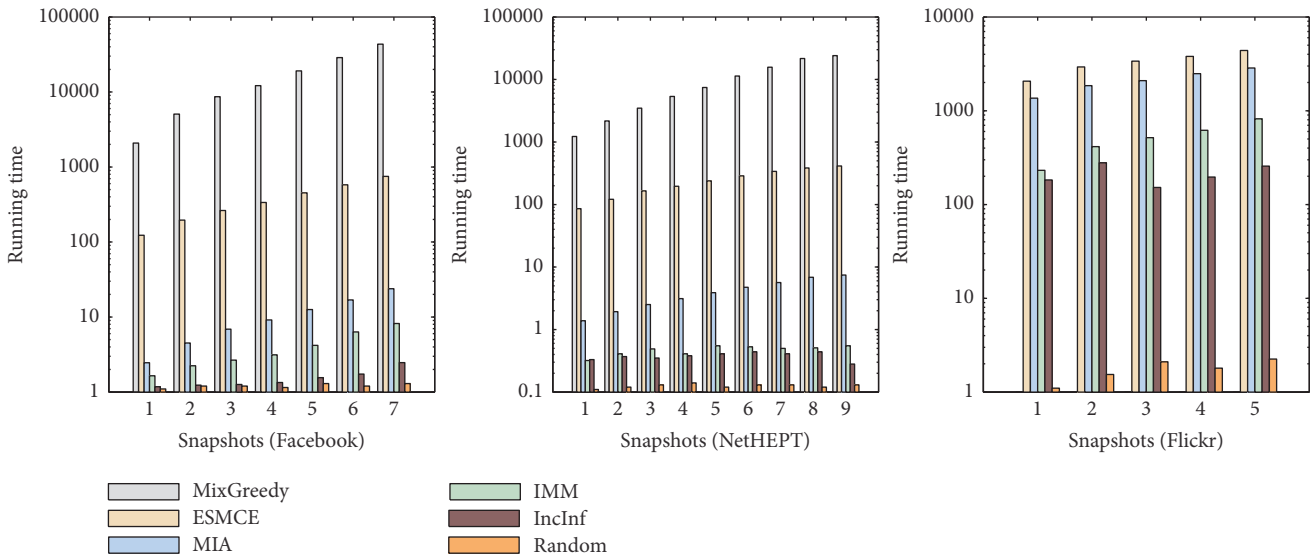


Figure 5: The time costs of different algorithms on three real-world datasets.

We can also observe that the running time of IncInf is not monotone like other algorithms as the time evolves. This is because the running time of IncInf is closely related to the topology change between two graph snapshots. An evident change in topology will usually lead to a relatively long running time and vice versa. Without doubt, Random runs the fast among all the algorithms. However, as we will show in Section 6.3, its accuracy is much worse and unacceptable when developing real-world viral marketing strategies.

We also test the effect of our pruning strategy. Here we take the Facebook dataset as an example; the results on other datasets are similar and thus are omitted. Different from other experiments, we recorded the Facebook graph from September 2006 to October 2007 (14 months) as snapshot A in this experiment. After that, we take snapshots every month as snapshot B. We use IncInf to find the top-K influential nodes in snapshot B based on ones in snapshot A. The result is shown in Figure 7. x-axis is the time interval between snapshots A and B, and y-axis is the ratio of the number of nodes after pruning to the total number of nodes in snapshot B. The minimum and maximum pruning ratios are 3.90% and 5.86%, respectively, with a mean ratio of 4.72% on all the 14 time intervals between snapshots A and B. This demonstrates that our pruning strategy can effectively limit the search space into a small percent of nodes. We can also see in Figure 7 that, with the increase of time interval, the ratio, although not monotone, generally becomes larger. This is mainly because a longer time interval means a larger amount of topology changes, and basically it will be possible for more nodes to become influential nodes.

6.3. Effectiveness Study. In this subsection, we study the influence spread of the top-K influential nodes selected by our algorithm as well as other static algorithms. The influence spreads of different algorithms are measured as the number of nodes that are influenced by the top-50 influential nodes selected. Obviously, the higher the influence spread, the better the effectiveness. We have not tested the performance of MixGreedy on the Flickr dataset as the running time is excessively long.

Figure 6 shows the experimental results. MixGreedy outperforms all the other algorithms in terms of influence spread. However, the efficiency issue limits its application to large-scale dataset such as Flickr. The performances of ESMCE, MIA, IMM, and IncInf almost match MixGreedy on the Facebook dataset, while on NetHEPT the gaps become larger but remain acceptable (only 3.4%, 4.7%, 4.5%, and 5.1% lower than MixGreedy on average). When applied to the Flickr dataset, ESMCE performs the best, since ESMCE strictly controls the error threshold by iterative sampling. The influence spread of MIA almost matches IMM on the three datasets and is slightly lower than ESMCE. Compared with MIA, IncInf shows very close performance and is only 2.87% lower on average of all five snapshots, which demonstrates the effectiveness of our proposal. Random, as the baseline heuristic, clearly performs the worst on all the graphs. Actually, the influence spread of Random is only 15.6%, 12.1%, and 10.9% of those of IncInf on Facebook, NetHEPT, and Flickr, respectively.

We shall note that the reason IncInf has slightly lower influence spread is mainly twofold. First, IncInf restricts the influence into local regions to speed up the computation of influence spread changes, which will affect the effectiveness. Second, a pruning strategy is designed to narrow down the search space based on the influence spread changes and previous top-K information. Despite slight loss in effectiveness, IncInf gains remarkable improvement in efficiency, as mentioned before.

6.4. Tuning of Parameters θ and η. First, we study how effectively the localization parameter θ of IncInf represents a

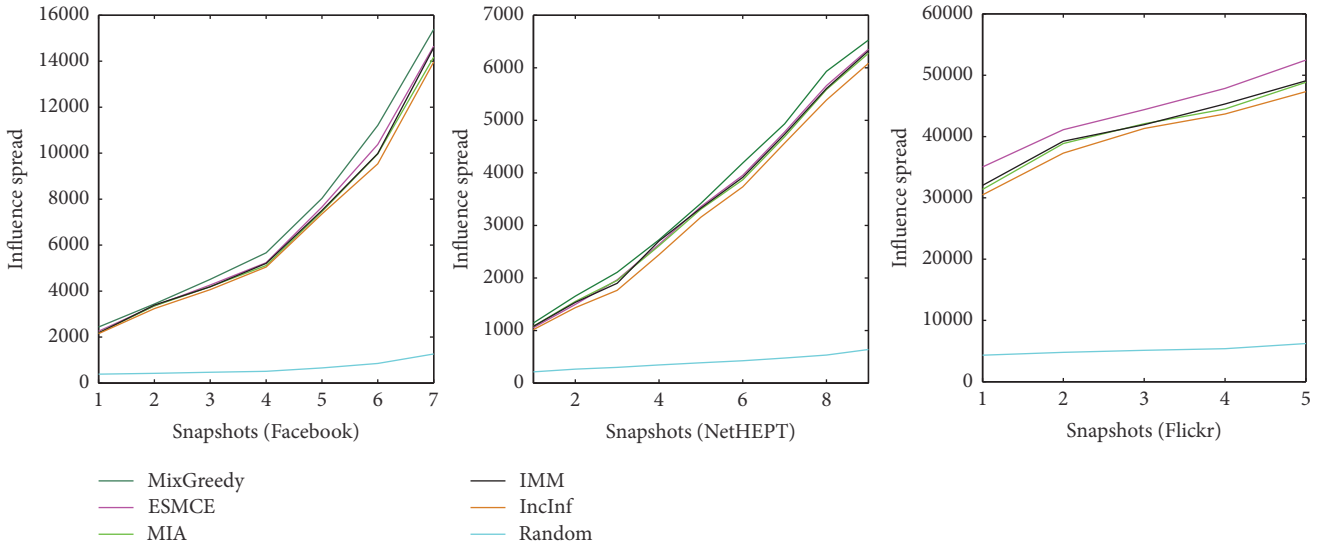


Figure 6: The influence spread of different algorithms on three datasets.

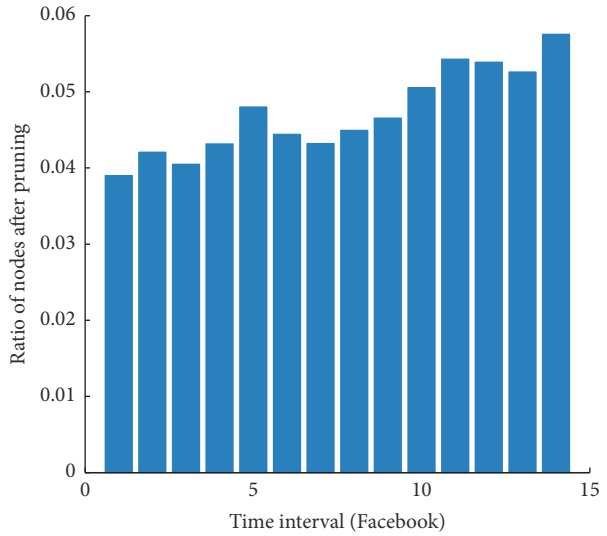


Figure 7: The effect of pruning strategy on the Facebook dataset.

tradeoff between efficiency and effectiveness. We run IncInf with different values of θ on the final Facebook and NetHEPT graphs. The running time and influence spread are measured based on seed size $K = 50$.

The experimental results are shown in Figure 8. Note that x-axis represents the reciprocal of θ. We observe that θ acts as a tradeoff between efficiency and effectiveness: with the decrease of θ, IncInf and MIA achieve better influence spread. However, this is gained at the cost of longer running time, that is, poor efficiency. For example, when we reduce θ from 1/200 to 1/500 on the Facebook dataset, the influence spread of IncInf increases by 15.4%, while the running time is 1.12x longer. Moreover, we can observe that the influence spread of IncInf almost matches that of MIA in all values of θ. For example, IncInf is only 1.87% lower than MIA in influence spread when θ is set to 1/200 in the NetHEPT dataset. But IncInf shows overwhelming advantages in terms of running time. When θ is set to 1/500 in Facebook, IncInf needs only 5 seconds to identify the top-50 influential nodes, while it takes MIA more than 150 seconds to finish the same work. More importantly, with the decrease of θ, the influence spread increases sharply at the beginning but the increase is no longer that significant after θ is lowered to a certain level. On the contrary, the running time is almost linear to $1/\theta$. This suggests that the knee point of the influence spread curve can serve as a good tuning point of θ, where we could obtain the best gain from both influence spread and running time.

Then, we will evaluate the sensitivity of pruning threshold η in terms of influence spread and running time. The results are illustrated in Figure 9. From Figure 9, we can see that, with the increase of η, the running time increases gently at the beginning and then turns into a sharp boost. For example,

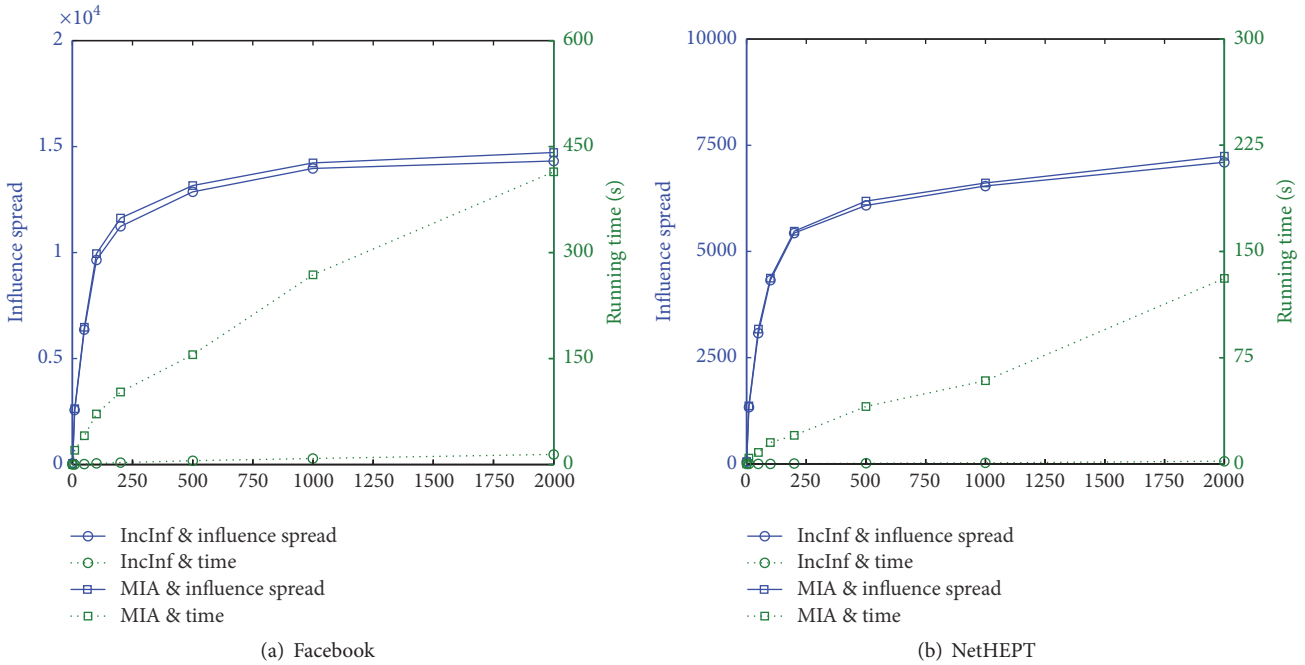


Figure 8: The effect of tuning of θ on running time and influence spread.

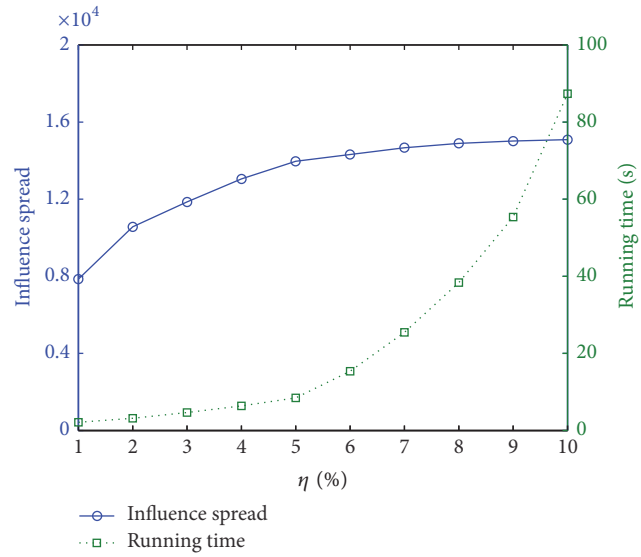


Figure 9: The effect of tuning of η on running time and influence spread.

when we increase η from 1% to 5%, the running time of IncInf on the Facebook dataset only increases from 2.13 s to 8.47 s, while it dramatically increases from 8.47 s to 87.35 when η is tuned from 5% to 10%. This phenomenon is closely related to the power-law distribution of degree in social network; when η is set to a large number, a relatively large number of potential nodes would be selected.

In terms of influence spread, with the increase of η, more nodes are selected as potential nodes, which will guarantee better influence spread. Different from the running time, the influence spread grows rather rapidly at the beginning and then gradually slows down. The influence spread on the Facebook dataset is 7854 when η is set to 1% and rapidly grows to 13967 when the maximum error threshold is 5%. After that, the growth trend slows down and the influence spread is about 15091 as η increases to 10%. The reason to explain such phenomenon is that the top-K influential nodes are mainly selected from high-degree nodes. Therefore, when η becomes larger, although more nodes would be selected, their contribution to influence spread is relatively small; thus the

growth trend slows down. Based on the above observation, here we suggest that 5% may stand as a good tradeoff between running time and influence spread.

6.5. Discussions. Experimental results demonstrate that our proposed IncInf algorithm significantly reduces the execution time of state-of-the-art static influence maximization algorithm while maintaining satisfying accuracy in terms of influence spread. Although IncInf performs better, it has a few limitations for further improvement.

First, IncInf directly depends on previous information of top-K influential nodes for effective pruning, while sometimes such information is incomplete or even unavailable. We plan to study this problem later. Second, IncInf is designed for the IC model, which may somehow limit its application. But we believe that our idea of incremental computation for influence maximization could be properly extended to other influence diffusion models.

7. Conclusion and Future Work

In this paper, we consider the influence maximization problem in evolving social networks and propose an incremental algorithm, IncInf, to efficiently identify top-K influential nodes in dynamic social networks. Taking advantage of the structural evolution of networks and previous information on individual nodes, IncInf substantially reduces the search space and adaptively selects influential nodes in an incremental way. Extensive experiments demonstrate that IncInf significantly reduces the execution time of state-of-the-art static influence maximization algorithm while maintaining satisfying accuracy in terms of influence spread.

There are several future directions for this research. First, IncInf has large potential to fit into modern parallel computing framework. This is because IncInf restricts the computation of influence spread changes into local regions, which could ease the partition of social graph for parallel computation. Moreover, the proposed pruning strategy could be effectively performed in parallel. Second, our current IncInf algorithm is derived from the basic IC model. We believe that the conception of incremental computation for influence maximization could be properly extended to other influence diffusion models, such as another classic LT model. Third, although there have been a few researches [43] about how to measure the propagation probability, this problem is not well addressed yet, especially for large-scale dynamic social networks.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was supported by NSFC under Grant no. 61402511.

References

[1] P. Domingos and M. Richardson, "Mining the network value of customers," in *Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '01)*, pp. 57–66, ACM Press, San Francisco, Calif, USA, August 2001.

[2] Twitter 2012 Facts and Figures, 2012, http://www.website-monitoring.com/blog/2012/11/07/.

[3] W. Chen, Y. Wang, and S. Yang, "Efficient influence maximization in social networks," in *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '09)*, pp. 199–208, ACM, July 2009.

[4] C. Wang, W. Chen, and Y. Wang, "Scalable influence maximization for independent cascade model in large-scale social networks," *Data Mining and Knowledge Discovery*, vol. 25, no. 3, pp. 545–576, 2012.

[5] J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. Vanbriesen, and N. Glance, "Cost-effective outbreak detection in networks," in *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '07)*, pp. 420–429, New York, NY, USA, August 2007.

[6] X. Liu, S. Li, X. Liao, L. Wang, and Q. Wu, "In-Time Estimation for Influence Maximization in Large-Scale Social Networks," in *Proceedings of the ACM EuroSys Workshop on Social Network Systems*, pp. 1–6, Bern, Switzerland, 2012.

[7] A.-L. Barabasi and R. Albert, "Emergence of scaling in random networks," *American Association for the Advancement of Science. Science*, vol. 286, no. 5439, pp. 509–512, 1999.

[8] X. Liu, X. Liao, S. Li, and B. Lin, "Towards efficient influence maximization for evolving social networks," in *Proceedings of the 18th Asia Pacific Web Conference*, pp. 232–244, 2016.

[9] Y. Tang, Y. Shi, and X. Xiao, "Influence maximization in near-linear time: a martingale approach," in *Proceedings of the ACM SIGMOD International Conference on Management of Data, (SIGMOD '15)*, pp. 1539–1554, Melbourne, Australia, June 2015.

[10] Y. Wang, G. Cong, G. Song, and K. Xie, "Community-based Greedy algorithm for mining top-K influential nodes in mobile social networks," in *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '10)*, pp. 1039–1048, ACM, July 2010.

[11] A. Goyal, W. Lu, and L. V. S. Lakshmanan, "CELF++: optimizing the greedy algorithm for influence maximization in social networks," in *Proceedings of the 20th International Conference Companion on World Wide Web, (WWW '11)*, pp. 47-48, Hyderabad, India, April 2011.

[12] K. Jung, W. Heo, and W. Chen, "IRIE: A Scalable Influence Maximization Algorithm for Independent Cascade Model and Its Extensions," https://arxiv.org/abs/1111.4795.

[13] X. Liu, M. Li, S. Li, S. Peng, X. Liao, and X. Lu, "IMGPU: GPU-accelerated influence maximization in large-scale social networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 1, pp. 136–145, 2014.

[14] Y.-C. Chen, W.-Y. Zhu, W.-C. Peng, W.-C. Lee, and S.-Y. Lee, "CIM: community-based influence maximization in social networks," *ACM Transactions on Intelligent Systems and Technology*, vol. 5, no. 2, article no. 25, 2014.

[15] X.-G. Wang, "An algorithm for critical nodes problem in social networks based on owen value," *The Scientific World Journal*, vol. 2014, Article ID 414717, 2014.

[16] Z. Lu, W. Zhang, W. Wu, J. Kim, and B. Fu, "The complexity of influence maximization problem in the deterministic linear

threshold model," *Journal of Combinatorial Optimization*, vol. 24, no. 3, pp. 374–378, 2012.

[17] Z. Lu, L. Fan, W. Wu, B. Thuraisingham, and K. Yang, "Efficient influence spread estimation for influence maximization under the linear threshold model," *Computational Social Networks*, vol. 1, no. 1, 2014.

[18] H. Nguyen and R. Zheng, "On budgeted influence maximization in social networks," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 6, pp. 1084–1094, 2013.

[19] M. Han, M. Yan, Z. Cai, and Y. Li, "An exploration of broader influence maximization in timeliness networks with opportunistic selection," *Journal of Network and Computer Applications*, vol. 63, pp. 39–49, 2016.

[20] B. Liu, G. Cong, Y. Zeng, D. Xu, and Y. M. Chee, "Influence spreading path and its application to the time constrained social influence maximization problem and beyond," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 8, pp. 1904–1917, 2014.

[21] S. Pei, X. Teng, J. Shaman, F. Morone, and H. A. Makse, "Efficient collective influence maximization in cascading processes with first-order transitions," *Scientific Reports*, vol. 7, Article ID 45240, 2017.

[22] Y. Y. Habiba, T. Y. Berger-Wolf, and J. Saia, "Finding Spread Blockers in Dynamic Networks," in *Advances in Social Network Mining and Analysis*, vol. 5498 of *Lecture Notes in Computer Science*, pp. 55–76, Springer, Berlin, Germany, 2010.

[23] R. Michalski, T. Kajdanowicz, P. Bródka, and P. Kazienko, "Seed selection for spread of influence in social networks: Temporal vs. static approach," *New Generation Computing*, vol. 32, no. 3-4, pp. 213–235, 2014.

[24] W. Chen, W. Lu, and N. Zhang, "Time-critical influence maximization in social networks with time-delayed diffusion process," in *Proceedings of the Proceedings of the 26th Conference on Artificial Intelligence*, Toronto, Canada, 2012.

[25] M. Gomez-Rodriguez and B. Scholkopf, "Influence Maximization in Continuous Time Diffusion Networks," in *Proceedings of the 29th International Conference on Machine Learning*, Scotland, UK, 2012.

[26] C. Aggarwal, S. Lin, and P. S. Yu, "On influential node discovery in dynamic social networks," in *Proceedings of the 12th SIAM International Conference on Data Mining, (SDM '12)*, pp. 636–647, SIAM, Calif, USA, April 2012.

[27] H. Zhuang, Y. Sun, J. Tang, J. Zhang, and X. Sun, "Influence maximization in dynamic social networks," in *Proceedings of the 13th IEEE International Conference on Data Mining, ICDM 2013*, pp. 1313–1318, Dallas, Tex, USA, December 2013.

[28] G. Tong, W. Wu, S. Tang, and D.-Z. Du, "Adaptive Influence Maximization in Dynamic Social Networks," *IEEE/ACM Transactions on Networking*, vol. 25, no. 1, pp. 112–125, 2017.

[29] D. Kempe, J. Kleinberg, and E. Tardos, "Maximizing the spread of influence through a social network," in *Proceedings of the Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 137–146, Washington, DC, USA, 2003.

[30] M. Richardson and P. Domingos, "Mining knowledge-sharing sites for viral marketing," in *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '02)*, pp. 61–70, ACM, Edmonton, Canada, July 2002.

[31] Y.-C. Chen, W.-C. Peng, and S.-Y. Lee, "Efficient algorithms for influence maximization in social networks," *Knowledge and Information Systems*, vol. 33, no. 3, pp. 577–601, 2012.

[32] M. Heidari, M. Asadpour, and H. Faili, "SMG: fast scalable greedy algorithm for influence maximization in social networks," *Physica A: Statistical Mechanics and its Applications*, vol. 420, pp. 124–133, 2015.

[33] F. Morone and H. A. Makse, "Influence maximization in complex networks through optimal percolation," *Nature*, vol. 524, no. 7563, pp. 65–68, 2015.

[34] G. Song, X. Zhou, Y. Wang, and K. Xie, "Influence maximization on large-scale mobile social network: a divide-and-conquer method," *IEEE Transactions on Parallel and Distributed Systems*, vol. 26, no. 5, pp. 1379–1392, 2015.

[35] C. Zhou, P. Zhang, W. Zang, and L. Guo, "On the upper bounds of spread for greedy algorithms in social network influence maximization," *IEEE Transactions on Knowledge and Data Engineering*, vol. 27, no. 10, pp. 2770–2783, 2015.

[36] A. Capocci, V. D. P. Servedio, F. Colaiori et al., "Preferential attachment in the growth of social networks: The internet encyclopedia Wikipedia," *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, vol. 74, no. 3, Article ID 036116, 2006.

[37] J. Leskovec, L. Backstrom, R. Kumar, and A. Tomkins, "Microscopic evolution of social networks," in *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2008*, pp. 462–470, Las Vegas, Nev, USA, August 2008.

[38] D. Easley and J. Kleinberg, *Networks, Crowds, and Markets: Reasoning about a Highly Connected World*, Cambridge University Press, Cambridge, Mass, USA, 2010.

[39] V. P. CORONA, "Memory, Monsters, and Lady Gaga," *The Journal of Popular Culture*, vol. 46, no. 4, pp. 725–744, 2013.

[40] B. Viswanath, A. Mislove, M. Cha, and K. P. Gummadi, "On the evolution of user interaction in Facebook," in *Proceedings of the 2nd ACM SIGCOMM Workshop on Online Social Networks*, pp. 37–42, Barcelona, Spain, August 2009.

[41] "ArXiv NetHEPT dataset," http://www.cs.cornell.edu/projects/kddcup/datasets.html.

[42] A. Mislove, H. S. Koppula, K. P. Gummadi, P. Druschel, and B. Bhattacharjee, "Growth of the flickr social network," in *Proceedings of the 1st ACM SIGCOMM Workshop on Social Networks (WOSN '08)*, pp. 25–30, ACM, August 2008.

[43] J. Tang, J. Sun, C. Wang, and Z. Yang, "Social influence analysis in large-scale networks," in *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '09*, pp. 807–815, Paris, France, July 2009.

Hindawi
Complexity
Volume 2017, Article ID 4543563, 12 pages
https://doi.org/10.1155/2017/4543563

Research Article

Influence of Personal Preferences on Link Dynamics in Social Networks

Ashwin Bahulkar,[1] Boleslaw K. Szymanski,[1] Nitesh Chawla,[2] Omar Lizardo,[2] and Kevin Chan[3]

[1]*Rensselaer Polytechnic Institute, 110 8th Street, Troy, NY 12180, USA*
[2]*University of Notre Dame, Notre Dame, IN 46556, USA*
[3]*US Army Research Laboratory, Adelphi, MD 20783, USA*

Correspondence should be addressed to Ashwin Bahulkar; ashwinbahulkar@gmail.com

Received 10 April 2017; Revised 30 June 2017; Accepted 24 July 2017; Published 20 September 2017

Academic Editor: Katarzyna Musial

We study a unique network dataset including periodic surveys and electronic logs of dyadic contacts via smartphones. The participants were a sample of freshmen entering university in the Fall 2011. Their opinions on a variety of political and social issues and lists of activities on campus were regularly recorded at the beginning and end of each semester for the first three years of study. We identify a behavioral network defined by call and text data, and a cognitive network based on friendship nominations in ego-network surveys. Both networks are limited to study participants. Since a wide range of attributes on each node were collected in self-reports, we refer to these networks as attribute-rich networks. We study whether student preferences for certain attributes of friends can predict formation and dissolution of edges in both networks. We introduce a method for computing student preferences for different attributes which we use to predict link formation and dissolution. We then rank these attributes according to their importance for making predictions. We find that personal preferences, in particular political views, and preferences for common activities help predict link formation and dissolution in both the behavioral and cognitive networks.

1. Introduction

A key observation in the sociological literature is that persons have a preference to connect to others with similar attributes as themselves [1]. This propensity is usually referred to as "homophily." Homophily can be based on a taste for similarity in values, beliefs, and attitudes. This is usually referred to as "value homophily." Homophily can also be based on a preference for similarity based on fixed or elective sociodemographic characteristics that define social groups (e.g., gender, age, race, and social class); this is usually referred to as "status homophily" [2]. Classic work in the social network analysis tradition sees homophily as a key tie formation mechanism [3]. Other things being equal, we should expect that new connections between previously disconnected persons should more likely to emerge among those who share common attributes [1].

While in some circumstances homophily operates as a direct tie formation mechanism, in some cases ties between similar alters may form not because people have a preference for people with similar attributes [4], but because people with similar demographics or opinions end up participating in a common "focus" of activity such as groups or associations (such as a sports league or a cultural club) or common leisure activities (playing games and exercising) [5, 6]. Thus, shared contexts or activities act as an additional tie formation mechanism, generating new connection among seemingly similar alters. Recent work examining the influence of shared contexts on tie formation processes confirms the impression that ties between similar alters are likely to be generated via this pathway [7–9].

Because either preferences for similar attributes or the sharing of common-activity foci create nonrandom dependencies between node characteristics and the likelihood of

tie formation and temporal persistence of social ties, both value and status homophily as well as shared contexts are key mechanisms implicated in explaining the temporal dynamics of social networks [3].

Most previous work in social network analysis focuses on the mechanisms that generate new ties between previously disconnected nodes [1]. More recent work starting with [10] has focused on the phenomenon of *tie decay*, which is the disappearance of an edge at a future point in time between two nodes that were previously connected [3]. Although empirical work on tie decay continues to be relatively scarce, recent work has focused primarily on how *structural* features of both the node (e.g., degree) and the edge (e.g., weight and triadic embeddedness) matter for decay [11–13]. This is primarily due to the fact that the usual data brought to bear to study decay processes in recent work (mostly based on networks constructed from interactions mediated via telecommunication technologies) is very thin on actual node attributes. Therefore very little is known as to how detailed node features such as attitudes, values, and leisure activities and shared contexts (but see [14, 15]) influence tie decay processes. Given the fact that some of these works hints at the fact that shared attributes serve to delay the tie decay process [10], it is likely that both status and value homophily as well as shared activities and foci may function as *decay delaying* mechanisms, protecting ties from dissolution over time [3, 14].

To advance the study of the role of personal preferences on the dynamics governing the temporal evolution of social ties, we leverage a unique social science dataset, NetSense [16], which is a dataset of about 200 students collected at the University of Notre Dame. The NetSense dataset is unique because, in contrast to other social network datasets, it is rich in attribute information: we know about every student's sociodemographic background, interests, opinions on social and political issues, and the activities in which every student participates at multiple time points. We refer to this information as attributes of the students. Calls and messages exchanged between students are also recorded. In addition to this, students declare periodically who their top twenty contacts are. NetSense data thus allows us to focus both on tie formation and tie decay processes defined over multiple (cognitive and behavioral) networks.

Using this information, we are able to identify two different social networks among the students: one is the behavioral network built from the call and message records in which students are the nodes and edges exist if a pair of students call or message each other. The other is the cognitive network built from the top twenty contacts reported by students in the periodic ego-network questionnaire. An edge between a pair of students exists if a student lists the other as a top contact in the surveys. Given the large amount of information that we have at our disposal about each student, we refer to both of these as attribute-rich networks. These networks are dynamic in nature, edges are created and dropped as persons add or subtract top contacts from the cognitive network and as communication volumes change over time in the behavioral network.

In previous work using the same dataset, we examined how both value and status homophily as well as preferences for common activities affect the formation of ties. Consistent with sociological theories of homophily, we found that, indeed, students with similar attributes are more likely to form ties with one another. In this previous work, we used an aggregated count of the number of common preferences as the main predictor. One question that remains unanswered, therefore, is whether there is heterogeneity across attributes in terms of their importance in producing the homophily effect on tie formation. For instance, it is possible that political views of the other person may not matter much when it comes to forming friendship, but the shared activities in which the other person takes part might matter more. A different hypothesis is that the political views of the other person are of paramount importance, but only in the case of behavioral ties based on communication volume.

Any ranking of which features matter more needs to be done while taking into account the fact that different personal preferences matter more or less for different people. That is, we need to take into account the fact that people may value different attributes more or less when making or breaking ties with other people, especially with regard to alternative values of the attribute. For instance, at each point in time, people have different values for an attribute; for example, the values a student can have for the attribute "political orientation" are conservative, moderate, and liberal. People may have varying degrees of preferences for or against each of those values. We look at the following scenarios as the main motivation for our study: a person who is liberal may have a strong preference for liberals, but he may not have any preference for or against moderates and conservatives. While on the other hand, another liberal may have a strong preference against conservatives and a moderate preference against moderates. Our previous approach to understanding the role of homophily [17] captures neither of these scenarios. The simplifying assumption was that all liberals have the same preference for or against conservatives and moderates. The preference for or against an attribute value can be guessed from the neighborhood of a person in the network. A person having a strong preference for an attribute value would have higher than average number of friends with this attribute value, while a person with a preference against would have lower than average number of friends with this particular attribute value.

Our proposed method, which we call the *Personal Preference Method*, takes these heterogeneous preferences into account and uses them to predict the formation and dissolution of edges. We look at the distribution of an attribute value in a person's neighborhood and compare it with the distribution of the attribute value in the entire network and use this difference to measure the preference the person has for or against the particular attribute value. We further use these preferences combined with a machine learning approach to predict formation or dissolution of edges.

Our paper advances over previous work by extending the *Personal Preference Method* to the task of predicting link dissolution. While prediction of edge formation has been studied well enough, prediction of edge dissolution has not

received as much attention [10]. Predicting dissolution of edges is harder than predicting formation of edges since formation of edges is a more structured process than decay [3]. We find significant differences in the number of attributes values over which nodes in edges which actually form agree, as compared to nodes in edges which do not ever form, with nodes forming edges agreeing across a wider range of attribute values a lot more than nodes which do not. However, differences between edges which dissolve and edges which persist are not very obvious [17]. With the help of our preference based method, we are able to improve the performance of prediction for link dissolution significantly from the performance values mentioned in [17].

2. Material and Methods

2.1. The NetSense Data. The NetSense data used in this study consists of university students' reports listing their personal traits, interests, views and opinions on various social and political issues, and background at the beginning of every school semester from the Fall 2011 to the Spring of 2013 [16]. At the beginning and end of each semester students are asked to fill out surveys where they list their friends. The data also consists of a record of the calls and texts exchanged between students participating in the study. We identify the evolving social networks among students out of this data.

Call and Text Messaging Data. We use the NetSense call and texts exchanged by students from August 2011 to May 2013. Each communication record consists of an entry for each call or text message, with the date, time, sender and receiver, and duration or length of the communication.

Friendship Surveys. Each student can list up to 20 friends at the beginning of the semester. The friends are either survey participants, students on the campus not in the survey, or family and friends outside the campus. We find that typically only two to three out of the 20 friends are survey participants. We form the network only out of the friendships which are between study participants.

Node Attributes. Students participating in the NetSense study filled out a survey at the beginning of each semester. Survey questions were about the students family background, major pursued in Notre Dame University, activities on campus, their views on different social issues, and their political inclinations. All attributes have multiple possible values out of which a student selects one. For example, students can select if their political views are conservative or moderate or liberal. For each student we selected the following attributes from the NetSense data:

Student background

(i) Concentration of study/major
(ii) Family income
(iii) Race/ethnic identification (e.g., Black/White)
(iv) Religious affiliation (e.g., Catholic/Protestant)

Social and political views

(i) General political orientation (e.g., liberal/conservative)
(ii) Opinion about legality of abortion
(iii) Opinion about marijuana legalization
(iv) Opinion about homosexuality and the legalization of gay marriage
(v) Views on racial equality and affirmative action

Habits and Lifestyle

(i) Drinking habits
(ii) Time spent weekly on activities like studying, partying, socializing, volunteering, campaigning for social causes, and exercising

Coevolving Networks in NetSense. From the NetSense data, we are able to identify and create two social networks among students. The first is the behavioral network, where an edge connects two students if calls are made or text messages are exchanged between them over the given semester. The second is the cognitive network, where an edge exists if one student lists the other as a top contact in the current survey. These two social networks evolve every semester, so we have four snapshots for both of the networks. The snapshots cover the following semesters: Fall 2011 semester ranging from August 2011 to December 2011, Spring 2012 semester lasting from January 2012 to May 2012, Fall 2012 semesters ranging from August 2012 to December 2012, and Spring 2013 semester lasting from January 2013 to May 2013. Since very few calls were made during the summer of 2012, we do not create a network for the summer semesters.

2.2. Related Work. Link prediction is a well-studied topic. The standard techniques for link prediction have been mentioned in [18, 19]. Most of the experiments are on collaboration networks between researchers. However, none of the networks in these papers are as rich in node attributes as NetSense. We study how homophily in terms of node attributes affects link prediction in [17]. While we were able to get reasonable results, the innovative *Personal Preference Method* proposed in this paper improves the quality of link prediction in NetSense.

Link dissolution is a less well-studied topic. Link dissolution in human mobility networks has been studied in [20] and link persistence prediction has been studied on phone calling data in [12]. However, the networks studied are not attribute-rich. Several methods for analyzing the effect of different network properties like reciprocity of links, assortativity on formation, and dissolution of links have been discussed in [21]. However, in contrast to the networks used in the study, the NetSense networks we study are much richer in node attributes. We also want to study the overall effect of all the node attributes that is why we use the machine learning methods described below. We experimented with a maximum likelihood approach to predict links as mentioned in [22]; however, we found that machine learning methods give much better performance. We also experimented with several

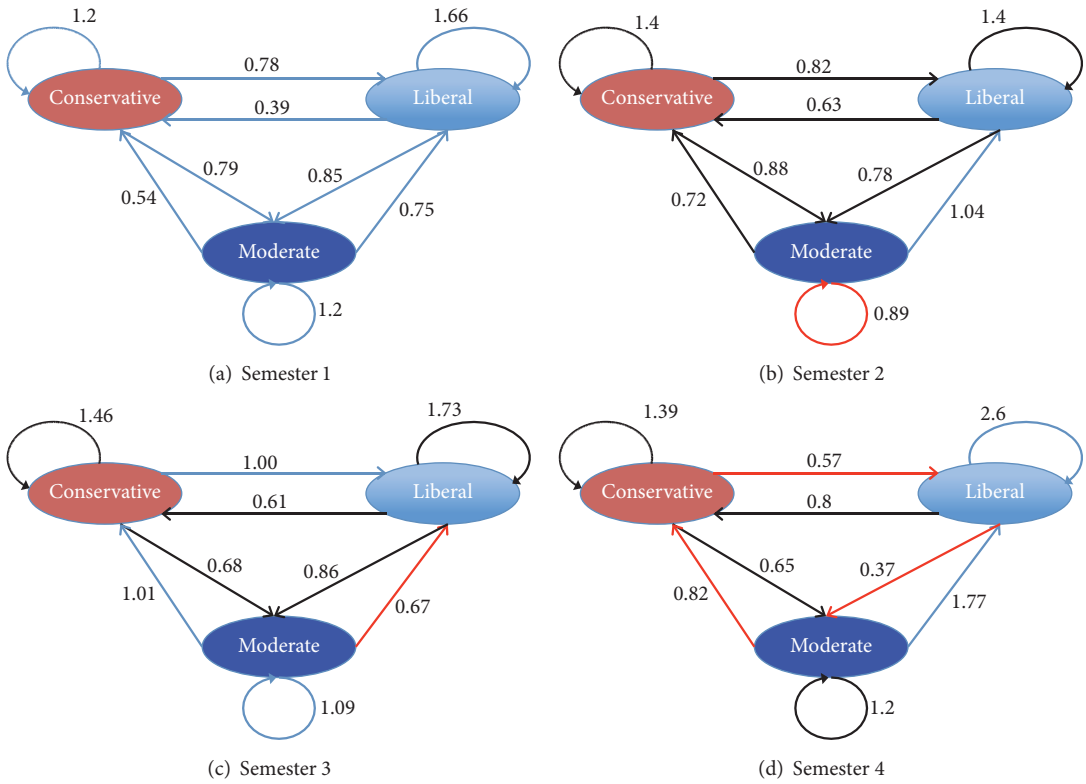


FIGURE 1: Views on politics, the average preference of nodes with a particular attribute value for all values of the attribute.

statistical methods and found that machine learning methods give us the best predictions. We studied link dissolution in the NetSense networks in [17]; however, the results were not very encouraging. With our new *Personal Preference Method*, we are able to make significant improvements over the previous effort. In addition, in [17], the experiments were performed exclusively on the behavioral network; in this paper, we extend the enhanced approaches to dynamics of decay in the cognitive network.

2.3. A Case for the Personal Preference Method. In our work in [17], we made the (reasonable) assumption, grounded in the sociological literature, that people prefer to form friendships with people who are exactly like them and we performed link prediction based on this assumption. In this section, we look at the limitations of this assumption. In the previous paper, our link prediction algorithm had assumed that a liberal would prefer liberals the most, followed first by moderates, and then by conservatives. Now, the distance between the preference values of moderates and liberals with each other and between moderates and conservatives was assumed to be equal. Also, we assumed that all liberals would have the same preferences. Preliminary analyses on the NetSense data reveal that this was probably too simplistic of an assumption. To illustrate dynamics of the links between people with different attribute values, in Figures 1–3, we visualize changes in the strength of connection between them. Attributes values are linked by the preferences between nodes possessing them. A value greater than 1 means a higher than average preference, a value less than 1 signifies a lower than average preference, and a value around 1 means an average preference. A blue line stands for a significant increase in the preference from the previous semester, a red line means a decrease in the preference, and a black line means no significant change in the preference. From these three figures, we observe the average preferences held by all the nodes possessing a particular attribute value, for all the values of this attribute. Although we find consistently that while people often have a strong preference for other people with the same attribute value as theirs, the preference for people with other attribute values does not necessarily follow a predetermined order. For example, in Figure 1, one would expect that liberals would have a higher preference for moderates than conservatives, but this is not always what we observe. Also, the changing preference values over the semesters make us reconsider our previous assumption that all the persons possessing a particular value for an attribute have the same preferences. This leads us to propose a method where we can account for every person's preferences and use them to make predictions as to which attributes are more important for social network evolution.

2.4. Methodology. We want to find out how well node preferences for different attributes predict the formation and dissolution of edges in both the cognitive and behavioral networks. We also want to know which attributes play the most important role in the formation and dissolution of edges. We first introduce a technique where we measure the

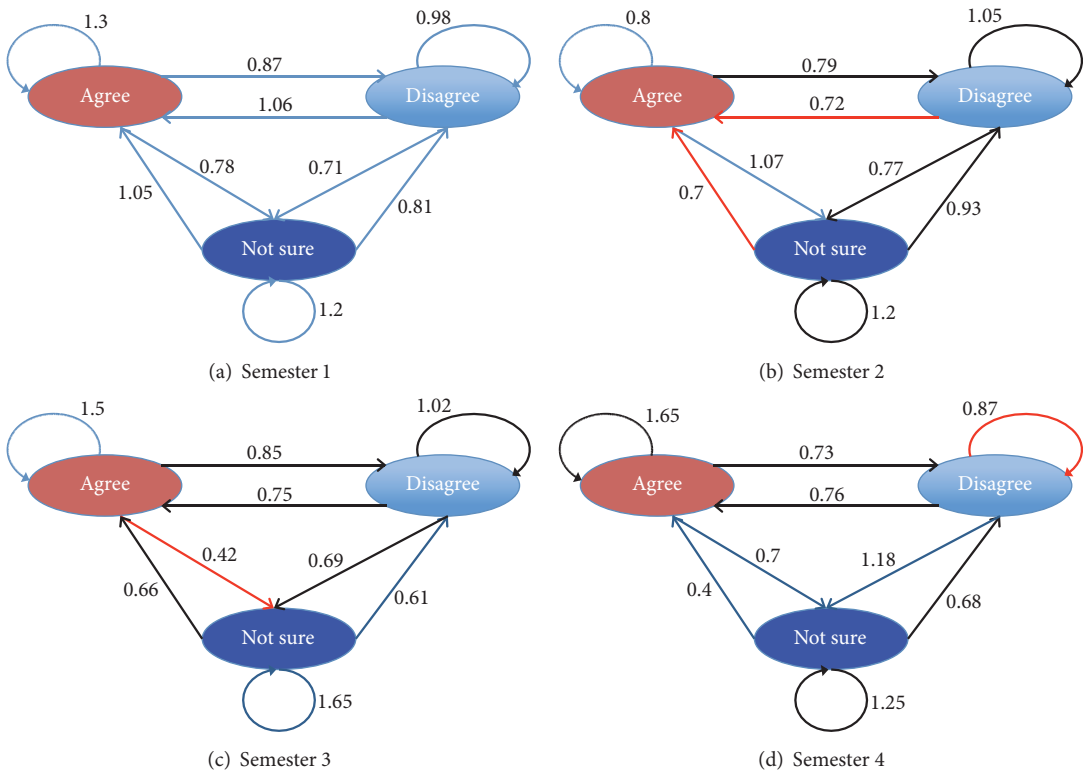


Figure 2: Views on gay marriage legalization, the average preference of nodes with a particular value for all values of the attribute.

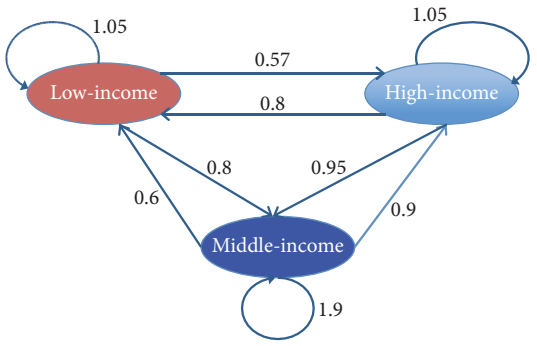


Figure 3: Parental income, the average preference of nodes with a particular value for all values of the attribute. Unlike others, parental income data was collected only once by the first survey.

preference of every node for every attribute value, based on the neighborhood of the node. We then propose a method to convert these preferences of a pair of nodes into features, which are used to predict the formation or dissolution of edges using machine learning. We then consider the different machine learning techniques used to make these predictions. Finally, we elaborate on how we obtain the relative importance of every attribute in the process of making predictions.

2.4.1. Link Formation Prediction. Link formation prediction can be seen as a classification problem where we predict whether an edge which connects a pair of nodes but still has not been formed will form in the future or not [23]. We use machine learning techniques to predict the formation of edges. Features for this prediction task are the preferences of the two nodes to be connected by the edge for each other's values of each of the attributes along with a network topology-based feature. We describe the *Personal Preference Method*, which creates these features, in Section 2.2. The network topology-based feature we use is the number of common neighbors between edge endpoints, or, in other words, the number of neighbors shared by those two nodes.

Classification Task and the Training and Test Datasets. New edges are predicted by machine learning models that learn from the way new edges were created in the past. To perform link formation prediction, we need three successive snapshots of the network, say, *net1*, *net2*, and *net3*. We predict which nodes will be joined by an edge in *net3*. So, yet to be formed edges in *net2* are the test set. Out of these edges, the ones which are formed in *net3* are true positive, and ones which are not formed in *net3* are true negative examples. The machine learning model has to be trained on the past, so we use the first two networks to train the model. The unformed edges in *net1* are the training set. Out of these, the ones which are created in *net2* are positive examples while the ones which do not form in the *net2* are negative examples.

Unbalanced Classification. The link formation prediction classification task is an unbalanced one. With n nodes and e

edges in the network, there are $((n^2 - 1)/2) - e$ possible edges which usually is many times more than the new edges formed. We observe that most nodes tend to link to other nodes when they are separated by no more than three hops. So, in our dataset, we consider only the edges whose nodes are separated by at the most three hops. However, still there are 50 times more negative examples than the positive ones. With machine learning algorithms, we have observed that we often need to compromise either on accuracy or on recall. Recall measures the percentage of new edges that were predicted to be as such, while accuracy measures the fraction of the predictions, regardless of whether they were positive or negative, that were accurate. The goal of our classification task is to select a prediction model that gives us the best balance between accuracy and recall. We choose a model which identifies as many new edges as possible, while, at the same time, not classifying too many negative edges as positive. We use the ROC curve with a weight assigned to recall being five times larger than that assigned to accuracy and choose the best model accordingly.

2.4.2. Link Dissolution Prediction. Link dissolution prediction too can be seen as a classification problem, where we predict whether an edge will dissolve or not. In [17], we found that predicting dissolution did not yield very good results. We know that a decrease in communication is a strong signal of declining friendship as shown in [24]. Yet, the cognitive edges are binary; they either exist or do not exist and the change comes once a semester. In contrast, the behavioral edges have weights, so, considering these facts, there are cognitive edges that may be under the process of dissolution, but this may not necessarily be reflected in the current friendship survey. To capture the process of dissolution, we define dissolving communication edges as those which reduce their communication in the succeeding semester to a third of their existing communication volume. Moreover, the cognitive edges whose corresponding communication edge is dissolving are also classified as dissolving. For clarity, an edge that gets dissolved in the current semester is also considered dissolving (this is by definition, for communication edges, but not necessarily for cognitive edges). With this definition, we redefine the classification task to predict dissolving edges and not only edges to be dissolved. To measure the weight of a communication edge, we aggregate the number of calls and text messages, assigning the weight 10 to each call and weight 1 to each message corresponding to the typical ratio of the number of messages to the number of calls in our data. We use machine learning techniques to predict the dissolving edges. The features used for prediction here are the same as those used for link formation prediction.

Test and Training Sets. We perform link dissolution prediction using machine learning models that learn from edges that have been dissolving in the past. The classification task is very similar to that of link formation prediction. For the classification task, we have three successive networks, *net1*, *net2*, and *net3*. We want to predict which the edges existing in *net2* will be dissolving in *net3*. From the nodes and network structure in *net2*, we predict which edges would be dissolving in *net3*. The machine learning model learns from the edges existing in *net1* and dissolving in *net2*. So, edges in *net1* form the training set. Edges which are dissolving in *net2* are true positive, while edges which are not dissolving in the *net2* are true negative examples. Similarly, for edges in *net2* that form the test set, edges which are dissolving in *net3* are positive examples, and edges which are not are negative examples.

2.4.3. Machine Learning Techniques Used. We use the standard Support Vector Machines (SVM), Linear Regression, and k-Nearest Neighbors (k-NN), Random Forests, and Naive Bayes classifiers as classification algorithms for all the classification tasks. They are the most commonly used classifiers in several link prediction works such as [18, 19, 23].

2.4.4. Validation Set. The validation set is used to fine-tune parameters of the machine learning algorithms. These parameters differ from algorithm to algorithm. In Linear Regression, we need to select the best threshold; with SVM and also with Linear Regression, we need to decide whether to use higher order features; with k-NN, we need to select the best value of "k"; and, with Random Forests, we need to select the best number of trees. The validation set contains randomly selected 20% of the training set.

2.4.5. Computing Node Preferences. We compute preference of a node for every possible value of every attribute a node can have. The preference a node has for a particular attribute value is computed based on how different the percentage of the node's friends having the particular attribute value is from the percentage of all the nodes having that particular attribute value in the entire network. This percentage of friends being lower than that percentage for the entire network indicates a negative node bias towards that particular attribute value, while opposite relation of these percentages indicates a preference for this attribute value. These percentages being equal indicates node's neutral attitude towards that particular attribute value. We use the statistical Z-Score [25] to measure how far from average is the number of friends with the said attribute for the given node. Z-Score is expressed in the standard deviation units. We normalized the Z-Score values into the range of $[0, 1]$ using the Z-Score tables [25].

The Personal Preference Method to Compute Node Preferences

Input. The input is a set, *Attrset,* of attributes, with each attribute, a, in the set having a set of possible values, $a.values$, that a node could have. Every node has a value for all the attributes. Each node n in the network has a set of neighbors.

Output. For every node n in the network, for each value v of each attribute, a preference value, $n.Preference(v)$, between 0 and 1 is returned, with 1 denoting a strong positive preference, 0 denoting a strong negative preference, and a value of 0.5 indicating no preference.

Step 1 (calculating the distribution of each attribute value in the network).

For each attribute a from *Attrset,*

for each value of attribute $av \in a.values$, find the percentage of nodes with the value v. We refer to this value as $a.v.percentage$.

Step 2 (calculating the preferences for each node).

For each node n in the network,

for each attribute a from *Attrset,*

for each value v of the attribute a from $a.values$,
(1) calculate the Z-Score:

Z-Score $= (x - \mu)/\sigma$, as defined in [25].
x is the actual number of friends of n with the particular attribute value.
μ is the expected number of friends with the particular attribute value in the network, which is $n.noNeighbors \times a.v.percentage$, where $a.v.percentage$ is obtained in Step 1 and $n.noNeighbors$ is the total number of neighbors n has.
σ is the standard deviation.

(2) Convert the Z-Score to a normalized range between 0 and 1 using the Z-Score table, and assign it to $n.Preference(v)$.

2.4.6. Building the Features for Classification. For the machine learning task, our feature set is computed from the preference scores of the nodes of the edge for every attribute. We define two methods here: one where preferences of both nodes matter, so we multiply the preference of one node by that of another for that particular attribute. We call this method the *Equal Preference Method*. Another method considers only the lower of the preference values of the two nodes. We want to find out if this simpler approach results in good predictions. We call this method the *Minimum Preference Method*.

Method for Converting Node Preferences to Edge Attributes

Input. Edge e with nodes $n1$ and $n2$. For a node n, the preference for a value val of each attribute a is denoted by $n.Preference(a.val)$.

Output. For each of the attributes, an agreement value between $n1$ and $n2$ is calculated. For the edge e, agreement on an attribute a is denoted by $e.Agreement(a)$.

Method.

For each attribute a,

$n1$'s value for a is denoted by $a.val1$; $n2$'s value for a is referred to as $a.val2$.

Equal Preference Method is as follows:

$e.Agreement(a) = n1.Preference(a.val2) \cdot n2.Preference(a.val1)$.

Minimum Preference Method is as follows:

$e.Agreement(a) = Min(n1.Preference(a.val2), n2.Preference(a.val1))$.

We use these feature values for classification of edges.

2.4.7. Estimation of Attribute Importance. We estimate the relative importance of every attribute that leads to formation and dissolution of edges in both the networks. We look at the coefficients of every attribute in the function returned by the Linear Regression classifier and use them to estimate the relative importance of each attribute. A higher coefficient is associated with higher importance in classification. The values of the coefficients cannot be interpreted literally, since several dependencies exist among the features. However, the coefficients are still a fair indicator of how important each attribute is, and the ranking of these weights still gives us a fair idea of the relative importance of each attribute.

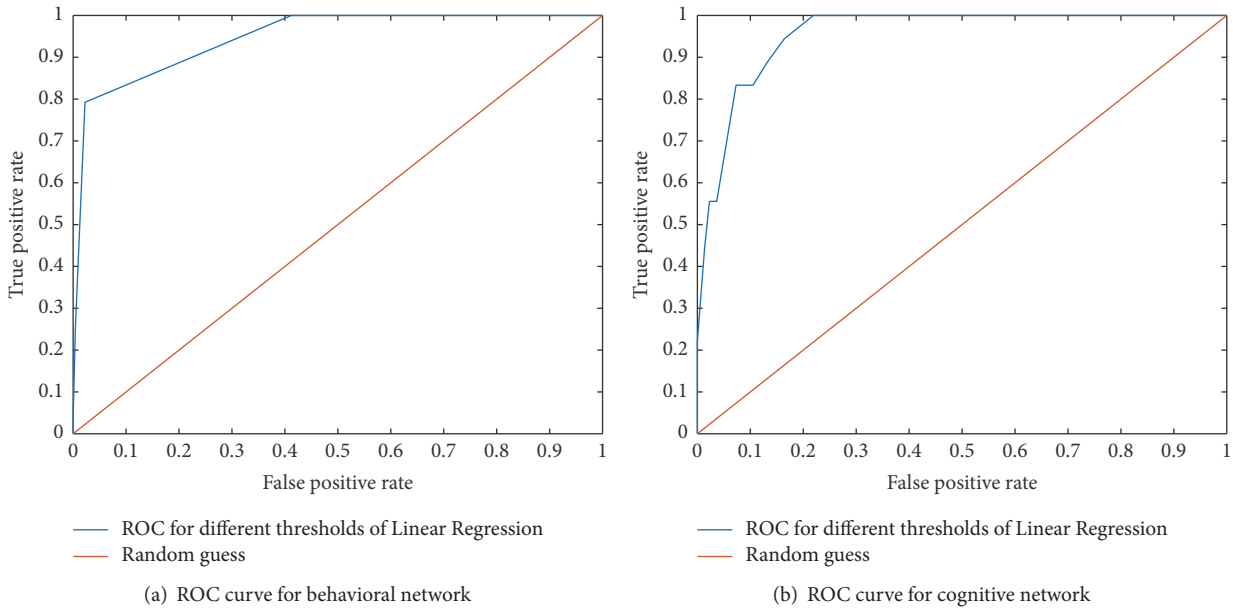
Table 1: Link formation prediction results for the behavioral network.

Semester		Semester 3	Semester 4
Linear Regression	Accuracy	97.5	97.8
	Recall	79.5	92.8
SVM	Accuracy	96.6	96.5
	Recall	89.9	92.8
k-NN	Accuracy	96.5	97.8
	Recall	92.8	88.1
Random Forests	Accuracy	98.5	98.0
	Recall	38.8	58.1

3. Results and Discussion

3.1. Link Formation Prediction Results. We measure link prediction performance using accuracy and recall. Recall measures the fraction of the created edges that were predicted as such while accuracy measures the fraction of predictions that were correct. We find that the accuracy and recall rates for link formation prediction have significantly improved over our earlier approach presented in [17]. This clearly demonstrates the benefit of using the node preferences for attribute values.

Linear Regression, SVM, and k-NN classifiers yield the best results, with high accuracy and high recall. Random Forests and Naive Bayes classifiers performed poorly. In [17], the best recall and accuracy achieved were 76% each, while here both were above 97%. Table 1 lists the results for the behavioral network, which is the same network we had used in [17]. We omitted results for Naive Bayes classifier due to its poor performance; we report results only for the *Equal Preference Method*, since *Minimum Preference Method* performed slightly lower. Table 2 shows the results for the cognitive network, which we did not analyze in [17]. We make predictions for formation of links in the third and the



(a) ROC curve for behavioral network

(b) ROC curve for cognitive network

Figure 4: The plot of the ROC curve for prediction of edge formation in both the behavioral and cognitive networks for different thresholds of edge weight in the second semester (the curves for other semesters are similar). The curve shows that the models fit the data well, but that the higher the recall, the higher the false positive rate. The prediction in behavioral network performs a little better than it does in the cognitive network.

Table 2: Link prediction results for the cognitive network.

Semester		Semester 3	Semester 4
Linear Regression	Accuracy	74.6	90
	Recall	100	100
SVM	Accuracy	92.8	77.8
	Recall	83.6	93.3
k-NN	Accuracy	94.5	90.0
	Recall	88.9	94.5
Random Forests	Accuracy	94.5	89.9
	Recall	58.1	50.4

fourth semester. The recall rates for the cognitive network are significantly higher than those for the behavioral network, possibly because of a smaller network size and a stronger tendency among nodes to adhere to their preferences while forming cognitive friends, as opposed to forming edges in the behavioral network. We also present the ROC curves in Figures 4(a) and 4(b).

In [17], we had used Singular Value Decomposition (SVD) for feature extraction. While this had enabled us to get an increase in the recall value then, using SVD with our *Personal Preference Method* here did not make any difference. We had also used additional features for classification like the "number of attributes on which the nodes of an edge agree" but using this feature did not make a difference in the results of our classification.

3.2. Link Dissolution Prediction. To assess the performance of link dissolution prediction, we measure the precision along with accuracy and recall, since this classification task is balanced, unlike link formation prediction. Precision is defined as the fraction of edges predicted to be dissolving by the classifier that are actually dissolving. Interestingly, the performance of the prediction of to be dissolved edges was very similar. We found that the accuracy, recall, and precision rates for prediction of edge to be dissolved have significantly improved over our earlier method presented in [17]. We are able to predict a significantly larger fraction of to be dissolved edges than in the past. This demonstrates the benefit of using the node preferences.

Table 3: Link dissolution prediction results for the behavioral and cognitive networks using Linear Regression.

Semester	Precision	Recall	Accuracy
Semester 3	80.6	92.5	83.1
Semester 4	82.4	90.1	84.2
Cognitive network			
Semesters 3 and 4	81.2	80.1	75.2

We also benefit from using edges which do not dissolve but whose nodes reduce the communication volume between them significantly. Linear Regression gives us the best results, with high accuracy, recall, and precision. SVM too yields good results, but not as good as Linear Regression. Random Forests and k-NN tend to yield pretty low recall results. We do not report the results for Naive Bayes, k-NN, and Random Forests classifier, since their accuracy rates were much lower than for the remaining methods. Table 3 lists results for both the behavioral and cognitive networks. We combine the results of two semesters for the cognitive network since there

Table 4: Weights of different attributes.

Semester	Link prediction		Link dissolution	
	Behavioral	Cognitive	Behavioral	Cognitive
Political views	1	0.6	0.78	0.22
Parental income	0.95	0.58	0.26	0.27
Drinking habits	0.22	0.05	0.13	0.08
Views on abortion	0.35	0.02	0.12	0.15
Views on gay marriage legalization	0.75	1	0.25	0.67
Views on homosexuality	0.2	0.55	1	0.28
Views on marijuana legalization	0.16	0.25	0.44	0.12
Major	0.38	0.18	0.21	0.26
Race	0.42	0.17	0.12	0.21
Religion	0.33	0.15	0.09	0.11
Number Common neighbors	0.84	0.35	0.13	0.07
Time spent on volunteering	0.8	0.35	0.14	0.48
Time spent on exercising	0.76	0.15	0.02	0.54
Time spent on studying	0.69	0.33	0.21	0.49
Time spent on partying	0.35	0.42	0.55	0.11
Time spent on university clubs	0.49	0.23	0.26	0.92
Time spent on socializing	0.71	0.12	0.57	1
Time spent on camping	0.69	0.43	0.01	0.35

Table 5: Ranks of different attributes.

Semester	Link prediction		Link dissolution	
	Behavioral	Cognitive	Behavioral	Cognitive
Political views	1	2	2	11
Parental income	2	3	6	8
Drinking habits	16	17	13	17
Views on abortion	14	18	15	13
Views on gay marriage legalization	6	1	8	3
Views on homosexuality	17	4	1	9
Views on marijuana legalization	18	10	5	14
Major	12	12	10	10
Race	11	13	14	12
Religion	15	14	15	16
Number Common neighbors	3	7	12	18
Time spent on volunteering	4	8	11	6
Time spent on exercising	5	16	17	4
Time spent on studying	9	9	9	5
Time spent on partying	13	6	4	15
Time spent on university clubs	10	11	7	2
Time spent on socializing	7	15	3	1
Time spent on camping	8	5	18	7

are very few edges which dissolve in the fourth semester. We report results from the *Equal Preference Method*.

3.3. Relative Importance of Attributes. We look at the coefficients of all attributes returned by Linear Regression. Table 4 lists the normalized relative weights of all the attributes, while Table 5 lists the rank of every attribute in the classification. A higher relative weight implies higher importance during classification. We present these rankings for the predictions for the fourth semester for both edge formation and dissolution prediction for both the networks. We observe that the coefficient weights are highly correlated for both semesters for the behavioral network, so the results shown here are just from the predictions made for semester 4.

We find that political views ranks high when it comes to formation of friendships in both networks and it ranks high in dissolution of edges in the behavioral network as well. Parental income, number of common neighbors, and

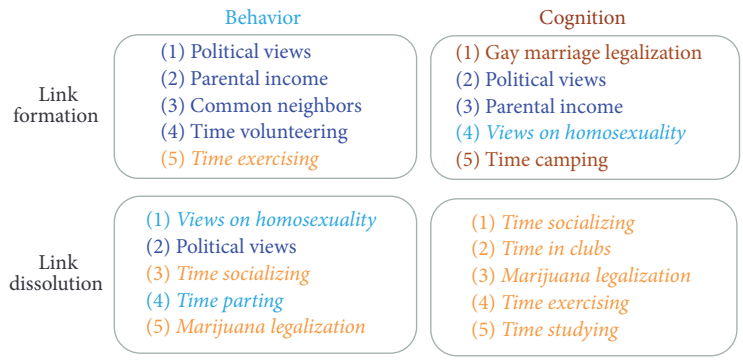


Figure 5: Comparison of top ranking attributes for link formation and dissolution of behavioral and cognitive edges. The four categories of edge dynamics, going from left to right and from top to bottom, are behavioral link formation, cognitive link formation, behavioral link dissolution, and cognitive link dissolution. The five highest ranking attributes for each category are shown in different colors. Interestingly, dissolution of cognitive edges has its attributes highest ranked among all categories. Only political views are shared by three categories, while parental income is common for link formation, and time socializing and marijuana legalization are common for link dissolution, while time exercising is shared diagonally and views on homosexuality are common along antidiagonal. Of the 13 distinct attributes listed, seven are unique for one category: common neighbors, time volunteering, gay marriage legalization, time camping, time parting, time in clubs, and time studying. More than half of listed attributes, seven, are some form of time spent together but, interestingly, different forms of spending time together have impact on different link categories of formation or dissolution.

time spent on common activities such as volunteering and exercising seem to rank higher in the formation of edges than in the dissolution of edges. Common activities such as partying and camping appear to matter more for dissolution than they do for link formation. Drinking habits, views on abortion, college major, race, and religion seem to rank low in all the networks, for both formation and dissolution. Views on the legality of gay marriage ranks higher in formation than in dissolution especially in the cognitive network for which it is the first and third most important feature, respectively. Views on moral propriety of homosexuality, time spent in clubs, and socializing seem to rank higher in dissolution than in formation of edges.

Comparing Figure 5 with Table 5, we can observe that six attributes: parental income, gay marriage legalization, political views, time volunteering, time in clubs, and time studying are among top 11 attributes for all four categories of edge dynamics. These attributes have highest influence on link formation and dissolution of both cognitive and behavioral edges. Among them, political views attribute is the most potent, being the first for behavioral link formation, the second for cognitive link formation and behavioral link dissolution, and 11 for cognitive edge dissolution. Collectively some form of spending time together is also very important; time socializing is ranked high for link dissolution; it is the first for cognitive and the third for behavioral link dissolution, while time volunteering is ranked fourth for behavioral link prediction with time camping and time parting ranked fifth and sixth for formation of cognitive links.

4. Long-Term Changes in the Network

We define strong edges as those whose nodes agree on more than t_s fraction of their attributes, and the remaining edges are called weak. Then, we check if generally the strong edges have a higher chance of survival than the weak edges do and how the difference depends on the threshold t_s used to define strong edges. Experimentally, we found that $t_s = 0.75$ is the best value for separating strong edges from the week ones. With this threshold, 80% of strong edges survive in semester 1, compared to 44% of weak ones. In semester 2, the survival rates are 63% for strong and 55% for weak edges, while, in the third semester, these rates are 75% for strong versus 78% for weak edges, a slight reverse in the trend. At the same time, the average fraction of strong versus weak edges changes slightly from 17% to 14% to 21% and finally to 20% over the four semesters. These semester-to-semester changes are not consistent because there are weak edges being created and dissolved in the network all the time. However, if only the edges which are at least one semester old are considered, then the clear trend is uncovered showing a steady increase of fraction of strong edges. This fraction grows from 17% in the second semester to 21% in the third and to 26% in the fourth semester. So over time, some tendency emerges to stabilize and increase homophily of the surviving edges.

5. Conclusions

Using the user's preferences for different attribute values we are able to make high quality predictions of formation and dissolution of edges. We have shown that this method is able to increase the performance of predictions in the NetSense networks in comparison to other strategies used in the past. We believe this method would be useful for making predictions in other attribute-rich networks and demonstrates how preferences of nodes can be harnessed to predict formation and dissolution of edges and thus contribute to our understanding of behavioral dynamics in social systems [3]. We also identified the relative importance of all the attributes in the formation and dissolution of edges. We found different attributes being top ranked for formation and dissolution of edges, suggesting that different factors might be responsible for formation and dissolution of ties

between people. We also found attributes occasionally having different rankings for predictions in behavioral and cognitive networks, suggesting that different factors play a role in the formation and dissolution of social ties based on subjective importance versus those based on behavioral frequency.

In all, the results are consistent with and provide important extensions of sociological approaches that see value homophily as a form of "cultural matching" and an important mechanism in both the formation of new social ties [15, 26] and dissolution of existing ones. Value homophily mechanism (e.g., views on homosexuality and gay marriage) seems particularly important when it comes to predicting the formation of new links in the cognitive network (based on subjective prominence) and when predicting behavioral dissolution of links.

This is particularly salient in the fact that political views emerged as the only factor that modulates both link formation and dissolution in the two networks (see Figure 5). This is in line with recent work [27] on the increasing salience of politics and the link between political views and lifestyles as an amplification mechanism (via homophily and social influence) driving patterns of social and geographical segregation in contemporary societies [28]. The fact that self-placement in the liberal-conservative continuum emerged as a preponderant predictor even when considering other attitudes and values associated with political orientation (e.g., views on abortion, marijuana legalization, and homosexuality) seems to indicate that persons are sorting into homogeneous group based on their self-identification as "conservative" or "liberal." This seems consistent with political views serving as marker of social identity [29] and not just as a factor impacting values and attitudes.

In addition, the preponderant role of common activities in generating link dissolution in the cognitive network (see Figure 5) lends support to Feld's theory of social foci [5] as an important complement to the value homophily mechanism in patterning tie decay in social networks [10]. Essentially, this means that once students stop having a set of common activities bringing them together for interaction, they decline in terms of subjective prominence as a "top contact." Note that in this respect, the mechanisms that produce new cognitive links are of a different nature than those that account for their decay (see upper right box of Figure 5) [3].

Finally, the relative lack of importance of group-level identifications ("status homophily") in modulating the temporal evolution of social ties in this network (e.g., common identities based on gender, race, and religion) is consistent with the view that most of the matching observed along these lines is modulated via either cultural matching or common-activity mechanisms [15, 30]. In all, the results reported here provide important sociological advances in our understanding of the role of cultural processes in generating patterns of connectivity and segregation in human social networks.

Disclosure

The views and conclusions contained in this document are those of the authors.

Conflicts of Interest

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

Acknowledgments

The authors would like to thank Panagiotis Karampourniotis for discussions. This work was supported in part by the Army Research Laboratory under Cooperative Agreement no. W911NF-09-2-0053 (the Network Science CTA) and by the Office of Naval Research (ONR) Grant no. N00014-15-1-2640.

References

[1] M. McPherson, L. Smith-Lovin, and J. M. Cook, "Birds of a feather: homophily in social networks," *Annual Review of Sociology*, vol. 27, pp. 415–444, 2001.

[2] P. Lazarsfeld and R. K. FMerton, "Friendship as a social process: A substantive and methodological analysis, Freedom and control in modern society," pp. 18–66, 1954.

[3] M. T. Rivera, S. B. Soderstrom, and B. Uzzi, "Dynamics of dyads in social networks: Assortative, relational, and proximity mechanisms," *Annual Review of Sociology*, vol. 36, pp. 91–115, 2010.

[4] P. M. Blau, *Structural contexts of opportunities*, University of Chicago Press, 1994.

[5] S. L. Feld, "The Focused Organization of Social Ties," *American Journal of Sociology*, vol. 86, no. 5, pp. 1015–1035, 1981.

[6] C. S. Fischer, *To dwell among friends: Personal networks in town and city*, University of Chicago Press, 1982.

[7] G. Mollenhorst, B. Völker, and H. Flap, "Social contexts and core discussion networks: Using a choice-constraint approach to study similarity in intimate relationships," *Social Forces*, vol. 86, no. 3, pp. 937–965, 2008.

[8] G. Mollenhorst, B. Völker, and H. Flap, "Social contexts and personal relationships: The effect of meeting opportunities on similarity for relationships of different strength," *Social Networks*, vol. 30, no. 1, pp. 60–68, 2008.

[9] G. Mollenhorst, B. Völker, and H. Flap, "Shared contexts and triadic closure in core discussion networks," *Social Networks*, vol. 33, no. 4, pp. 292–302, 2011.

[10] R. S. Burt, "Decay functions," *Social Networks*, vol. 22, no. 1, pp. 1–28, 2000.

[11] T. Raeder, O. Lizardo, D. Hachen, and N. V. Chawla, "Predictors of short-term decay of cell phone contacts in a large scale communication network," *Social Networks*, vol. 33, no. 4, pp. 245–257, 2011.

[12] C. A. Hidalgo and C. Rodriguez-Sickert, "The dynamics of a mobile phone network," *Physica A: Statistical Mechanics and its Applications*, vol. 387, no. 12, pp. 3017–3024, 2008.

[13] J. L. Martin and K.-T. Yeung, "Persistence of close personal ties over a 12-year period," *Social Networks*, vol. 28, no. 4, pp. 331–362, 2006.

[14] K. Lewis, J. Kaufman, M. Gonzalez, A. Wimmer, and N. Christakis, "Tastes, ties, and time: A new social network dataset using Facebook.com," *Social Networks*, vol. 30, no. 4, pp. 330–342, 2008.

[15] S. Vaisey and O. Lizardo, "Can cultural worldviews influence network composition?" *Social Forces*, vol. 88, no. 4, pp. 1595–1618, 2010.

[16] A. Striegel, S. Liu, L. Meng, C. Poellabauer, D. Hachen, and O. Lizardo, "Lessons learned from the NetSense smartphone study," in *Proceedings of the Annual Conference of the ACM Special Interest Group on Data Communication on the Applications, Technologies, Architectures, and Protocols for Computer Communication, ACM SIGCOMM 2013*, pp. 585–590, chn, August 2013.

[17] A. Bahulkar, B. K. Szymanski, O. Lizardo, Y. Dong, Y. Yang, and N. V. Chawla, "Analysis of link formation, persistence and dissolution in NetSense data," in *Proceedings of the 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, ASONAM 2016*, pp. 1197–1204, usa, August 2016.

[18] R. N. Lichtenwalter, J. T. Lussier, and N. V. Chawla, "New perspectives and methods in link prediction," in *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '10)*, pp. 243–252, July 2010.

[19] M. Al Hasan, V. Chaoji, S. Salem, and M. Zaki, "Link prediction using supervised learning," in *Proceedings of SDM 06 workshop on Link Analysis, Counterterrorism and Security*, 2006.

[20] Y. Yang, N. V. Chawla, P. Basu, B. Prabhala, and T. La Porta, "Link prediction in human mobility networks," in *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, ASONAM 2013*, pp. 380–387, can, August 2013.

[21] T. A. B. Snijders and C. Baerveldt, "A multilevel network study of the effects of delinquent behavior on friendship evolution," *Journal of Mathematical Sociology*, vol. 27, no. 2-3, pp. 123–151, 2003.

[22] T. A. Snijders, J. Koskinen, and M. Schweinberger, "Maximum likelihood estimation for social network dynamics," *The Annals of Applied Statistics*, vol. 4, no. 2, pp. 567–588, 2010.

[23] D. Liben-Nowell and J. Kleinberg, "The link prediction problem for social networks," in *Proceedings of the 12th ACM International Conference on Information and Knowledge Management (CIKM '03)*, pp. 556–559, ACM, November 2003.

[24] A. Bahulkar, B. K. Szymanski, K. Chan, and O. Lizardo, "Coevolution of two networks representing different social relations in NetSense," in *Complex Networks & Their Applications V*, vol. 693 of *Studies in Computational Intelligence*, pp. 423–434, Springer International Publishing, Cham, 2017.

[25] E. Kreyszig, *Advanced Engineering Mathematics*, Wiley, 4th edition, 1979.

[26] O. Lizardo, "How cultural tastes shape personal networks," *American Sociological Review*, vol. 71, no. 5, pp. 778–807, 2006.

[27] D. DellaPosta, Y. Shi, and M. Macy, "Why do liberals drink lattes?" *American Journal of Sociology*, vol. 120, no. 5, pp. 1473–1511, 2015.

[28] T. A. Diprete, A. Gelman, T. Mccormick, J. Teitler, and T. Zheng, "Segregation in social networks based on acquaintanceship and trust," *American Journal of Sociology*, vol. 116, no. 4, pp. 1234–1283, 2011.

[29] N. Ellemers and S. A. Haslam, "Social identity theory," *Handbook of Theories of Social Psychology*, pp. 379–398, 2012.

[30] A. Wimmer and K. Lewis, "Beyond and below racial homophily: ERG models of a friendship network documented on facebook," *American Journal of Sociology*, vol. 116, no. 2, pp. 583–642, 2010.

Hindawi
Complexity
Volume 2017, Article ID 4037049, 8 pages
https://doi.org/10.1155/2017/4037049

Research Article

Evolutionary Mechanism of Frangibility in Social Consensus System Based on Negative Emotions Spread

Yao-feng Zhang,[1,2] Hong-ye Duan,[2,3] and Zhi-lin Geng[2]

[1]*Collaborative Innovation Center of China Pilot Reform Exploration and Assessment-Hubei Sub-Center, Hubei University of Economics, Wuhan 430205, China*
[2]*School of Information Management and Statistics, Hubei University of Economics, Wuhan 430205, China*
[3]*Faculty of Mathematics and Statistics, Hubei University, Wuhan 430062, China*

Correspondence should be addressed to Zhi-lin Geng; zhilingeng@163.com

Received 20 March 2017; Accepted 22 May 2017; Published 28 June 2017

Academic Editor: Pasquale De Meo

To study the social consensus system under the spread of negative emotions, the nonlinear emergence model of frangibility of social consensus system is established based on Multiagent method, and effects of emotions spread frequency, opinion leaders, and shielding behavior of government on the frangibility of social consensus system are revealed. The simulation results show that the low-frequency negative emotions spread is better than the high-frequency one for reducing the frangibility of social consensus system. Low-frequency negative emotions spread will lead to the group polarization, while high frequency will lead to the collapse of system. The joining of opinion leaders who are with negative emotions can promote the frangibility of social consensus system, and collapse speed of social consensus system tends to increase with the influence of opinion leaders. Shielding behavior of government cannot effectively block the spread of negative emotions. On the contrary, it will enhance the frangibility of social consensus system.

1. Introduction

With the social transformation and the popularity of Internet, people's lives extend onto the Internet from reality gradually. So, the interpersonal relationship networks show complex characteristics of interaction between online and offline. With the fast spread of information on Internet, a complex social consensus system forms. This complex system is not only affected by the individual emotions within the group but also affected by the interference of external events. Through many times' diffusion of negative emotions, imbalance of group memory [1] tends to emerge. Then, collective behaviors on network frequently happen, which may result in many new collective behaviors in reality [2] and eventually lead to fragility of social system. The famous examples include Arab Spring [3] and Occupy Wall Street [4].

The original research of system frangibility began in the study of natural disaster system in late 1960s [5], and it refers to the changes of the system's structures and functions coursed by the sensitivity of the disturbance and lacking of resistance. Most early studies focused on the natural disasters [6], climate changes [7], ecosystem [8], groundwater system [9], and other natural sciences. With the rise of complexity science, studies about the frangibility of social system emerged gradually, like urban system, information system, economic system, and so on [10]. In particular, the frangibility of complex networks has attracted scholars' attention widely [11], examples include the frangibility of e-mail networks [12] and Internet [13] and transport networks and infrastructure networks [14].

Generally, for the studying of system frangibility, there are two kinds of methods. One is the evaluation index system [5, 15], and the other one is mathematical model which develops with the rise of the studies in complex network frangibility [16]. All kinds of comprehensive research methods based on these two methods are fully discussed and applied, such as AHP, entropy method, principal component analysis, Gray Cluster Analysis, fuzzy comprehensive evaluation method, and function method. Though the above research methods have pushed the studies of system frangibility to a new level,

most of the studies consider the system as a static system with little consideration of the complex structure of the system and seldom consider the coupling interaction between consensus diffusion and the frangibility of social system in the Internet age.

Because the social consensus system is not only a dynamic system with high frequency of update but also an interactive system between virtual world and real world, during the spread of consensus, we could not only take emotions spread as the cause (or effect) of frangibility but also take frangibility as the cause (or effect) of emotions spread. So, the above traditional methods could not reveal the inherent mechanisms of the frangibility evolution from the microlevel. In recent years, scientific computing (including modeling and simulation) has become the third paradigm of scientific research after scientific experiments and theoretical derivation [17]. The Multiagent System [18–21] (MAS) plays an indispensable and important role in complex system study because of its complex features such as dynamic high-dimensional, coupling feedback and the overall emergence [22]. Some recent studies show that MAS is becoming an important method in the research of social system frangibility [23, 24].

According to the above analysis, we proposed a Multi-agent simulation model for frangibility evolution in social consensus system. This model consists of two mechanisms which describe the processes of information spread: emotion diffusion mechanism and emotion influence mechanism. Emotion diffusion mechanism is used to describe the process of group information dissemination in the hot event, and the emotion influence mechanism presents the evolutional process of the internal viewpoints in the group after the information dissemination. We simulate many times occurrences of the negative events in social consensus system by repeating the above two processes serially and explore the influences on frangibility evolution from three factors: event frequency, intervention of opinion leaders, and government shielding behavior. Because many studies show that the interpersonal relationship networks have Free-Scale characteristics [25], we use BA Networks [25] as the topology structure of information spread network of social consensus system in this paper.

Compared with previous studies, this paper has the following differences: at first, most of the previous studies consider the system as a static system or only consider the impact of one node failure on the system frangibility [6–10, 12]. In this paper, we will study the frangibility of dynamic systems under repeated negative emotion diffusion. Secondly, most of the previous simulation studies focus on the cascade failure process of network nodes [23, 24], but the diffusion mechanism and the influence mechanism of node failures are not distinguished. In this paper, the two mechanisms are, respectively, corresponding to two kinds of process, emotion diffusion and emotion influence (postdiffusion evolution), which is much more fitting the actual characteristics of social consensus system. Thirdly, most of the simulations in the past discussed focus on the frangibility of social system caused by emergent events but paid little attention to the emergent events caused by frangibility of social system. In this paper, the interactive phase-disturbing process of the two factors is fully considered and analyzed with specific cases.

The paper is structured as follows. In the next section, we will present the networks model and the Multiagent model for frangibility evolution in social consensus system. Then, we will take "1.25 Incident in Egypt" [3] as an example to analyze the simulation results in Section 3 which we only consider the influences of spread frequency at first and, then, introduce the opinion leaders and shielding behavior of government into the system gradually. Finally, we will draw the conclusions with a brief discussion in the last section.

2. The Model

Studies have shown that either in real society or on Internet, interpersonal relationship networks exhibit the characteristic of Free-Scale [25]. Considering that the social consensus transmits through interpersonal relationship networks, we adopt BA Networks proposed by Barabási and Albert as the information spread network in social consensus system [25]. Assuming that the network has only m_0 nodes in the beginning, a new node with a connection degree m ($m < m_0$) is added at each time step, and the new node connects to m different nodes already existing in the network. The probability that new node connects with the existing node i is

$$P_i = \frac{k_j}{\sum_{j=1}^{N} k_j}, \tag{1}$$

where N is the total number of nodes and k_i is the degree of the node i. After t time steps, a scale-free network with $m_0 + t$ nodes and mt edges will be generated.

The above information spread network is also the emotion diffusion network of the consensus system, each node equipped with an Agent who represents the individual in networks. Because each person has a recognition or evaluation on the status of society and these recognitions or evaluations are usually shown as emotions, we regard the individual i's emotion to the society as $x_i(t)$ at time t. Usually, the value of $x_i(t)$ can be discrete [26] ($x_i(t) = \pm 1$) or continuous [27, 28] ($-1 \le x_i(t) \le 1$, $x_i(t) \in R$) in the study of consensus dynamics. Since continuous value is more suitable for reflecting the change process of individual emotions, we will take a continuous value for $x_i(t)$ in this paper.

In fact, in any society, all kinds of events are happening all the time. Some events attract more people's attention because of their specificity or sensitivity. And so the events change to hot events whose information spread very fast on the Internet. Because most of the hot events are full of "positive energy" or "negative energy," the process of information spread is also the process of emotion spread in the interpersonal relationship network. Usually, the information spread process of hot events contains two key subprocesses: the first subprocess is emotion diffusion. The core issue of this subprocess is whether the people who access the hot events are affected by it or will spread the information to other people. The second subprocess is emotion influence. That is, the process after the first subprocess (emotion diffusion)

tends to be stable. People who have accessed the hot events change their emotions by discussion in this subprocess. We will propose the model of two subprocesses which are corresponding to the emotion diffusion mechanism and emotion influence mechanism in the following.

2.1. Emotion Diffusion Mechanism. All social consensus is generated from the social hot events and exposed by few people who we called sponsors of emotion spread, and they get an initial emotion whose value is x_0 ($-1 \le x_0 \le 1$). The sponsor j will pass the event information and negative emotions to his neighbor i through the interpersonal relationship network. Now, the neighbor i should face two problems.

One is the accepting degree of emotions passed by the sponsor. The accepting degree is usually determined by i's conformity (acceptability) α_i and the sponsor's impact β_j [27]. The lager the individual conformity and the sponsor's impact are, the greater the individual accepting degree will be [2]. According to the literatures [2, 27], we assume that the emotion impact degree of the sponsor j to neighbor i is proportional to x_0, α_i, and β_j. Therefore, at time $t + 1$, the emotion of neighbor i can be described as

$$x_i(t+1) = x_i(t) + \alpha_i \beta_j x_0, \tag{2}$$

where $\alpha_i \sim U(0,1)$, $\beta_j \sim U(0,1)$.

Another problem is whether the individual wants to transfer the emotions to others who do not access the hot event yet. In fact, whether the individual will continue to transfer the information depends on the degree of event influence and importance [29]. According to the literature [29, 30], we assume that the probability that individual continues to spread the negative emotions is

$$P = \frac{1}{1 + e^{-\alpha_i \beta_j |x_0|}}, \tag{3}$$

where $|x_0|$ is the initial emotion generated by the hot event, which represents the importance of the event. $\alpha_i \beta_j$ represents the influence of the negative emotions on individual.

Emotion diffusion mechanism shows that the individual's emotion will be affected by events in varying degrees. And probability that individuals spread the negative emotion increases with the influence of events. The most important characteristic of emotion diffusion process is the emotion spread and dissemination on the network, which is a scale expansion process of the individuals who contact hot events.

2.2. Emotion Influence Mechanism. Events information and emotions quickly spread by the emotion diffusion mechanism in Section 2.1 after a social hot event occurs. In general, the duration of the process is short, and the longer process is the continual interactions and discussion among the individuals after the information diffusion. Individual's emotion will be influenced by neighbor groups and so evolve dynamically in the process of exchange and discussion and even tend to group polarization [27].

In fact, there are two main factors that determine an individual's emotion during the process of emotion influence. One is individual influence scope, and we use the proportion of individual degree of the total degree to measure this influence; for example, if individual j is one of individual i's neighbors, the influence of individual j on individual i is determined by the ratio of d_j/d_{it}, where d_j is j's degree and d_{it} is the total degrees of all i's neighbors. The other one is the differences of emotion. Because individuals always like to talk with individuals standing on the same camp but ignore the different views [28], if the emotions x_i and x_j differ by more than a fixed parameter ε, nothing happens because the two Agents think too differently to interact [31]. So the larger the differences in individual emotion, the smaller the mutual influence, and the converse is also true. Consistent with [31], we use $e^{-|x_i - x_j|}$ that represents the influence of emotion differences. Based on the above analysis, we define the influence of neighbor's emotion as

$$I_{Ni} = \frac{1}{n_i - 1} \sum_{i=1}^{n_i} \frac{d_j}{d_{it}} e^{-|x_i - x_j|} x_j, \tag{4}$$

where x_j is the emotion of individual j and n_i is the neighbor number of i. d_j/d_{it} is the ratio of individual degree of total degree and its value determines the influence of j on i.

According to the theory of social comparison [29], when $I_i > 0$, influence of neighbor's emotion tends to be positive. Then, under this situation, it is possible for individual to tend to change emotion to positive, and the bigger I_i, the bigger the emotion changing probability. On the contrary, when $I_i < 0$, the influence of neighbor's emotion tends to be negative. Now, individual maybe tends to change the emotion to negative, and the smaller I_i, the bigger the emotion changing probability. So, we present the following conversion rules of emotion as [27].

For individual i, when $I_i > 0$, add a small positive ε ($\varepsilon > 0$) on its emotion according to the probability $P_1 = 1/(1 + e^{-I})$:

$$x_i(t+1) = x_i(t) + \varepsilon. \tag{5}$$

On the contrary, when $I_i < 0$, subtract a small positive ε ($\varepsilon > 0$) on its emotion according to the probability $P_2 = 1/(1 + e^{I})$:

$$x_i(t+1) = x_i(t) - \varepsilon. \tag{6}$$

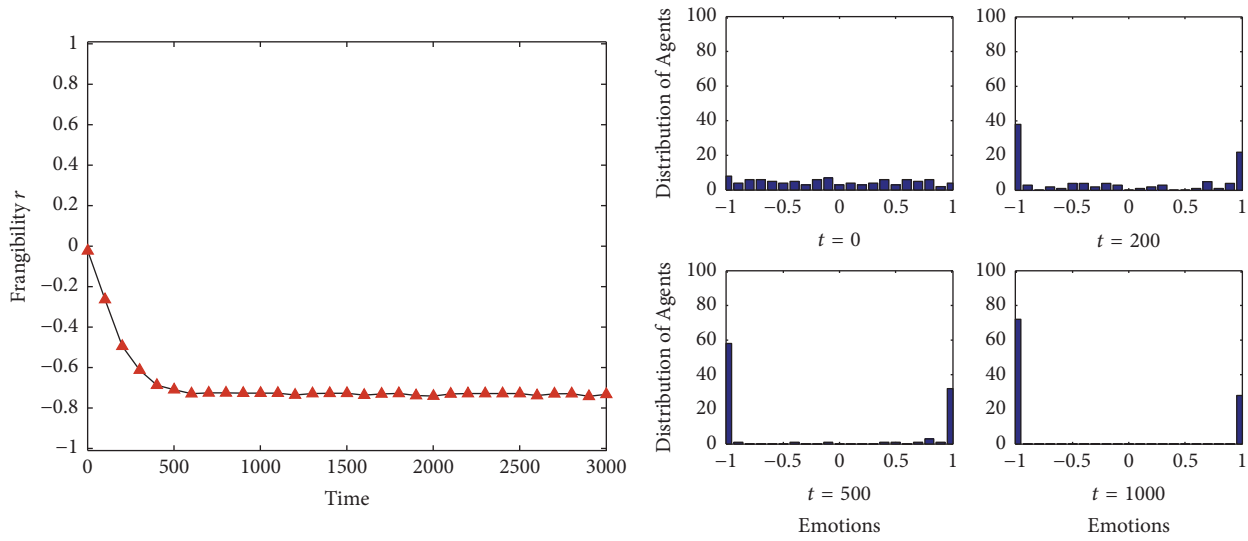
Over a period of time, hot events maybe arise several times, so the above process of emotion diffusion and influence will repeat continuously. Obviously, the average emotion of all individuals in system is one key factor which can represent the overall state of the social consensus system. Therefore, we define the parameter of system frangibility as

$$r = \frac{1}{N} \sum_{i=1}^{N} x_i(t). \tag{7}$$

Obviously, the social consensus system tends to be stable when $r \to 1$ and tends to be vulnerable when $r \to -1$.

The detailed simulation algorithm is as follows.

Step 1. Generate BA Network with N nodes as the information spread network, every node represents one Agent, and initialize the parameters α_i, β_j, and ε.



(a) Evolution of the frangibility in social consensus system

(b) Evolution of individual emotion distribution

FIGURE 1: Evolution of social consensus system with low negative event frequency.

Step 2. Select one node randomly as the negative emotion sponsor every time interval, and initialize x_0 between −1 and 0 as the initial value of the negative emotion.

Step 3. Update the value of emotion through emotion diffusion mechanism introduced in Section 2.1.

Step 4. Update the value of emotion through emotional influence mechanism introduced in Section 2.2.

Step 5. Repeat Steps 3 and 4 until the system tends to be stable.

The standard of system stability is that the emotions of individuals do not change anymore at the last 500 steps. When the system reaches stability, the simulation will terminate. It is important to note that the system frangibility is an important factor that affects the system evolution, and it will determine where the system terminates during the evolution sometimes [32, 33]. In this paper, the system frangibility is comprehensively determined by negative event frequency, opinion leaders, and government shielding. We will reveal the complex dynamics in the next section.

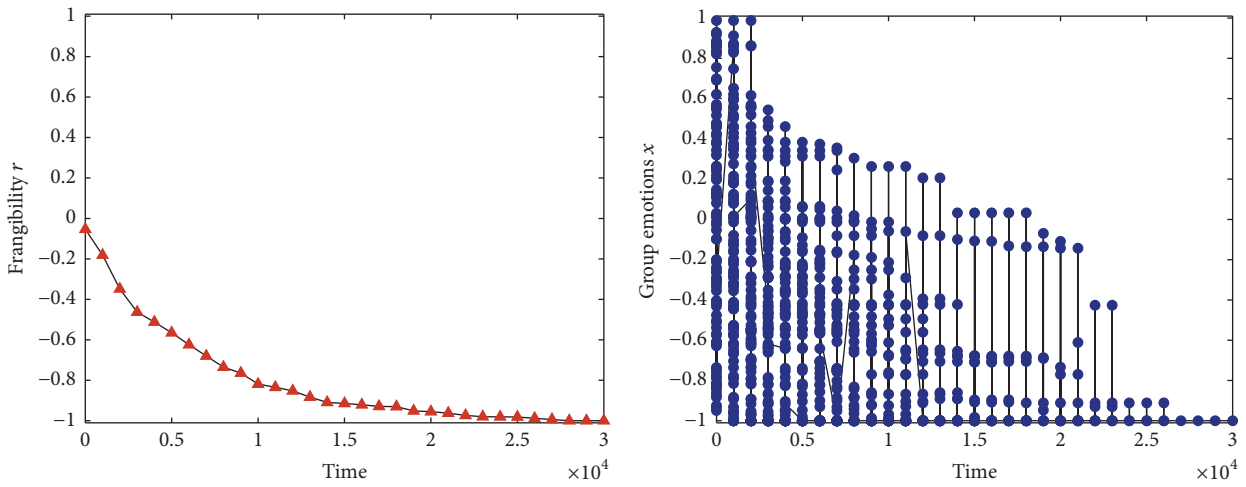
3. Simulation Results and Analysis

Based on the simulation model established in Section 2, we generate a scale-free network where $N = 100$. Each node is equipped with an Agent to represent an individual in the social consensus system. We take the initial state parameters of Agent are $\alpha_i \sim U(0,1)$, $\beta_j \sim U(0,1)$, $x_0 \sim U(-1,0)$, and $\varepsilon = 0.01$, and the initial emotion distribution of Agent follows a uniform distribution from −1 to 1. It is worth emphasizing that different values of ε may influence the evolutionary process of system. High values of ε significantly accelerate the convergence process, and small ε will take more time to achieve stability. The results of simulation are the average value of 30 experiments. We experiment for $N = 200$, 300, and 500, respectively, and get similar results except for the computation time.

3.1. Influence of Negative Event Frequency. Each negative event is associated with the spread of negative emotions. Firstly, we simulate the influence on social consensus system under the different frequencies of negative emotions spread (Figures 1 and 2). As we can see from Figures 1 and 2, individuals with positive emotions play an active role in the emotion influence process when the frequency of negative events is low (once every 50 simulation times). Although it has experienced the influence of negative emotions many times, frangibility in social consensus was still maintained at a certain level (Figure 1(a)). At this time, the group is divided into two extremes, the positive emotion group and the negative emotion group, and group polarization emerged (Figure 1(b)). With the increase of negative events frequency (once every 20 simulation times), system lost the support of the positive emotion group. Finally, social consensus system collapses completely (Figure 2). It is observed that the spread of high-frequency negative emotion can enhance the frangibility in social consensus than low-frequency one.

From 2010 to 2011, many countries in the Middle East have launched a series of large-scale collective behaviors called "Arab Spring" [3]. Among these collective behaviors, "1.25 Incident in Egypt" [3] is the most famous one for its large-scale, long-lasting, complete, and clear evolution process. So, we will take "1.25 Incident in Egypt" as an example to analyze the simulation results and reveal how the spread of negative emotions influences vulnerability of social consensus system as follows.

In fact, "Egypt 1.25 incident" is not an accident caused by extreme behavior but a breakout of the people's negative



(a) Evolution of the frangibility in social consensus system

(b) Evolution of group emotion

Figure 2: Evolution of social consensus system with low negative event frequency.

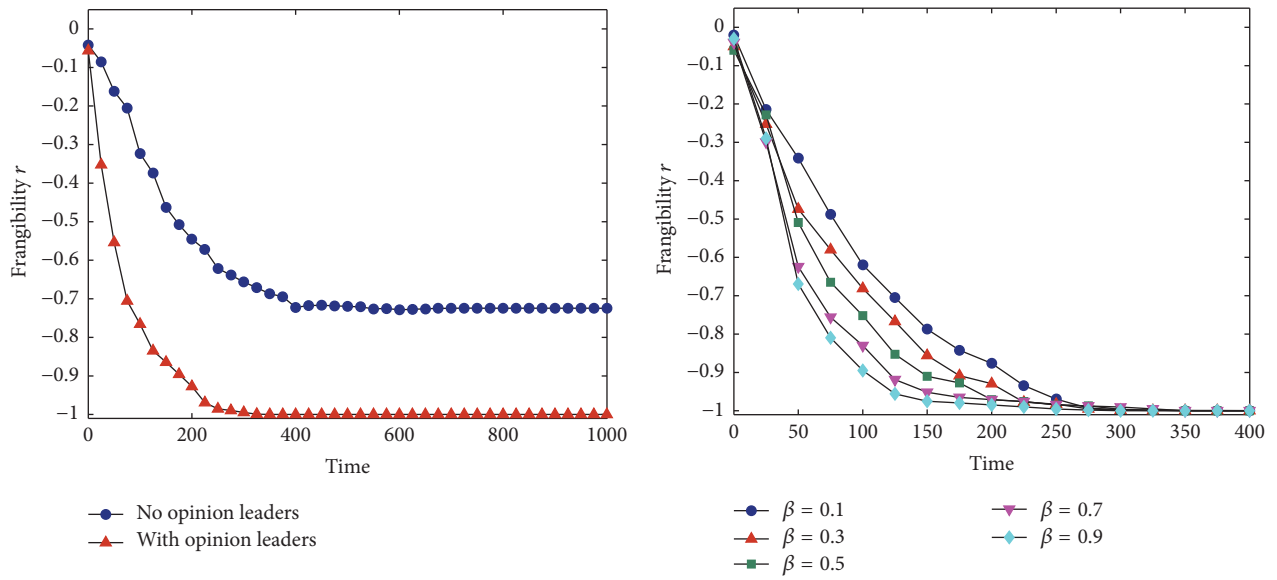
emotions under the accumulation of negative events. Before the "Twitter Revolution" in Tunisian and the "Saïd Incident" [3], the Egyptian government has widespread problems of autocracy and corruption, negative events break out every now and then, and persons who live in the bottom society have been depressed for a long time. But, at this point, the diffusion frequency of negative emotions is relatively low, it has not infected the emotions of middle-class people completely, and negative emotions are not enough to make this part of people give up the positive support to the Egyptian government. Therefore, the social consensus system was still in part stable state at this time (Figure 1). However, after a long time of dictatorial regime of Mubarak, unemployment rate increased, police violence and official corruption occurred frequently, and negative events often happened, situation of society was turbulent, and negative emotions even began to spread in the middle-class people. After the Tunisian "Twitter Revolution" and "Saïd Incident," large-scale negative emotions caused by the "fuse" incident spread at a higher frequency and the social consensus system collapsed completely (Figure 2). The negative social events in Egypt from low frequency to high frequency have been going on for a long time, with the increase of frequency of negative events, frequency of negative emotions spread speeded up, and people used Twitter, Facebook, and other online community tools to spread the negative emotions spontaneously, which increase the vulnerability in social consensus system gradually and lead to the collapse of the system in the end.

3.2. Influence of Opinion Leaders. The negative event frequency impact on evolution of the frangibility in social consensus system is important, while a special type of individuals in the group usually tends to be ignored, that is, opinion leaders who usually emerge sometimes during the evolution of social consensus system. Though few researchers begin to care about the issue in recent years, they do not consider the new situation and new features of the Internet [29, 34]. Here, according to the important role in evolution of Internet group emotion, we take the factor of opinion leaders into account. We will focus effort on how the opinion leaders impact on the frangibility in social consensus system as follows.

We randomly select one node as opinion leader of social consensus system in consensus information spread network at first. And, considering the stubbornness of opinion leader, we make its emotion always -1 [35]. By increasing the influence parameter of opinion leader β, we get the same conclusions as drawn in literature [29, 34–36]. That is to say, the appearance of opinion leaders with negative emotion promotes the emergence of consensus and strengthens the frangibility of social consensus system (Figure 3).

As we can see from Figure 3(a), though public consensus system is impacted by the negative emotions constantly, the system still can maintain a certain frangibility level before the opinion leaders appearance. With the emergence of opinion leader with negative emotions, the balance of public consensus system is broken. Under the influence of opinion leader, the consensus system evolves rapidly and tends to be more vulnerable, which eventually steps into a system crash. Figure 3(b) shows that, with the influence of opinion leader increasing, the speed of frangibility of social consensus system tends to speed up.

Then, we will analyze the effects of opinion leaders in "Egypt 1.25 Incident." As opinion leaders, social activist Mashahed who is a well-known journalist that publishes images, pictures, and other information constantly through the video site, which causes more and more participants to join in the protest team. These opinion leaders have played a key role in the development of "Egypt 1.25 Incident." Before the opinion leaders appeared, the social consensus system had been affected by the negative emotions of the "Twitter Revolution" and "Saïd Incident," but the middle class do not completely go backward to toiler group, and



(a) Evolution of the system frangibility before and after the emergence of opinion leaders

(b) Influence of opinion leaders on system frangibility under different β

FIGURE 3: Influence of opinion leaders on the frangibility in social consensus system.

the vulnerability of social consensus system still maintained a certain level (Figure 3(a)). With more and more opinion leaders such as the influential social activists and well-known journalists publicizing and disseminating information about the processions and rallies actively, the procession and protest team become bigger and bigger. Negative emotions spread to an unprecedented situation and finally the wavering middle class and even some upper-class also are pulled into the protest team. With the rapid development of the Internet and new media, popular websites and blogs have gradually developed into influential public opinion leaders. Their influence is determined by their authority and the number of their fans. The greater the influence is, the stronger the ability that the system evolves toward its own views will be (Figure 3(b)). The opinion leaders in "Egypt 1.25 Incident" are important influential members of the community, and their behavior leads Egyptian social consensus system to collapse.

3.3. Influence of Government's Shielding Behavior. In order to prevent the spread of negative emotions, the government might take technical measures like cutting Internet to shield the spread of information. So, we take the experiments on how the shielding behavior will influence the consensus system's frangibility as follows. We start the simulation system at first, and, then, when the simulation time reaches 1000, we randomly remove 50 nodes to simulate the government's behavior of shielding Internet. Figure 4 shows the results of several simulation experiments. We can see that though the government has taken the shielding behaviors (after simulation experiments reach 1000), they still undergo failure to prevent the social consensus system to tend to be vulnerable. On the contrary, they make frangibility of social consensus system increased dramatically, and government's behavior

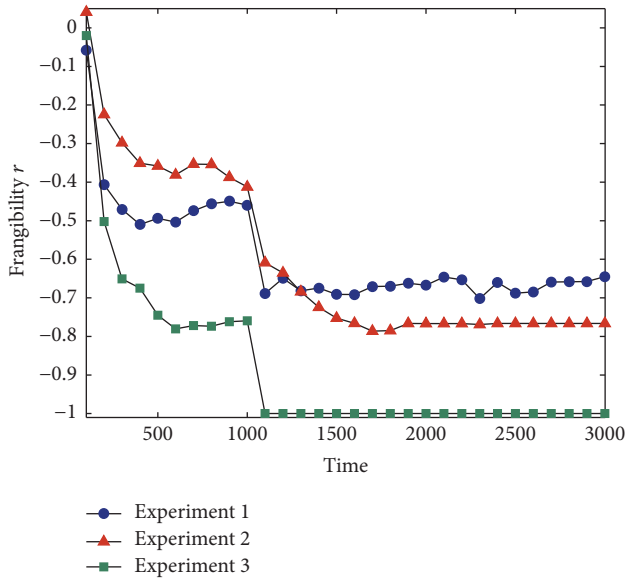


FIGURE 4: Evolution of the frangibility in social consensus system under government shielding behavior.

makes opposite effects. In fact, the government's information shielding behavior is a significant negative consensus event on itself, and at the same time it may lead to rapid spread of negative emotions among individuals, which will speed up the collapse of social consensus system in the end.

Finally, we analyze the effects of government's shielding behavior in the "Egypt 1.25 Incident." More than half of the people in Egypt are Internet users. Most of them have registered at Twitter, Facebook, and other online community

tools. At the early time of the "Egypt 1.25 incident," people spread the negative emotions of the government through these online community tools, telephones, or other communication tools which provided a communication platform for the breakout of "Egypt 1.25 incident." In order to curb this phenomenon and control the spread of negative emotions, the Egyptian government blocks and interrupts the Internet and even telephone communication for a long time. However, people have a deep-rooted negative impression to the Egyptian government, and these measures not only fail to control consensus but also exacerbate the people's confrontation emotions to government. People consider the government's blocking behavior itself as a serious negative event and continue to spread information through Internet, mouth to mouth, and so on. Combining with the guidance of opinion leaders, negative emotions spread among the people and increase the vulnerability of social consensus system (Figure 4).

4. Conclusions and Discussions

Social consensus system is a complex system which is formed by the diffusion and spread of consensus through complex network of interpersonal relationships. Like the most complex systems [6–14], social consensus system also has the characteristic of frangibility. When the average value of the individual emotions in the system is negative, the system becomes vulnerable. Different from precious studies which adopt the methods of the evaluation index system [5, 15] or mathematical model [16], this paper studies the evolutionary mechanism of frangibility in social consensus system based on Multiagent method. The simulation results show that high-frequency diffusion of negative emotion can enhance the frangibility of social consensus system compared to low-frequency one. The spread of high-frequency negative emotion even leads to system collapse and large-scale collective behaviors. In order to reduce the risk of social consensus effectively, government should avoid negative incidents by improving public service awareness and preventing public relations crisis, and improving the level of emergency management, and the positive propagandas and consensus guidance also should be strengthened at the same time. In addition, opinion leaders play an important role in the evolution of social consensus frangibility. The negative emotions propagated by opinion leaders may spread widely and fast, shielding behavior could not inhibit the spread of negative emotions; on the contrary, it will strengthen the frangibility of consensus system. So, the government should guide the opinion leaders correctly, for example, paying more attentions to deal with the negative events exposed by opinion leaders in time so as to prevent the rapid spread of negative emotions. Another suggestion is that the government can develop a government image management system based on the network data technology to realize the effective monitoring and early warning of the network consensus, so as to eliminate the negative impact of events and maintain social stability.

As the social consensus system is a typical self-organized emerging system which has the micro-macro effects, the characteristics of Multiagent System determine that it is not only a powerful tool to explore the micromechanism of complex systems but also an important method to study the micro-macro effects [37], which is very suitable to study the evolution of complex system [17]. Simulation results show that Multiagent simulation model proposed in this paper can reveal the evolution mechanism and provide a new perspective for the research of the frangibility in social system. Many further studies may be developed as the following aspects: (1) building a network model based on the actual data through the analysis of social network. Analyzing the impacts of network structure, individual characteristics, and negative events on the frangibility in social consensus system. (2) Studying the evolution of frangibility in social consensus system in the process of negative emotion interaction between online and offline; (3) using big data technology for analysis of negative emotion diffusion in social consensus system based on typical cases.

Conflicts of Interest

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

Acknowledgments

This research was supported by the National Statistical Science Foundation of China (Grant no. 2016LZ20), National Natural Science Foundation of China (Grant nos. 71671060, 41501183), Natural Science Foundation of Hubei (Grant no. 2014CFB374), and Science and Technology Innovative Team Foundation of Hubei (Grant no. T201516).

References

[1] J. Olick K, V. Vinitzky-Seroussi, and D. Levy, Eds., *The Collective Memory Reader*, Oxford University Press, New York, NY, USA, 2011.

[2] Y. Zhang and R. Xiao, "Modeling and simulation of synchronous threshold in vent collective behavior," *Discrete Dynamics in Nature and Society. An International Multidisciplinary Research and Review Journal*, no. atricle 170968, 12 pages, 2014.

[3] E. D. Mansfield and J. Snyder, "Democratization and the Arab Spring," *International Interactions: Empirical and Theoretical Research in International Relations*, vol. 38, no. 5, pp. 722–733, 2012.

[4] N. R. Kleinfield and C. Buckley, Wall Street Occupiers, Protesting till Whenever. New York times, 2011, October 1, A1.

[5] L. Rygel, D. O'Sullivan, and B. Yarnal, "A method for constructing a social vulnerability index: an application to hurricane storm surges in a developed country," *Mitigation and Adaptation Strategies for Global Change*, vol. 11, no. 3, pp. 741–764, 2006.

[6] A. Fekete, M. Damm, and J. Birkmann, "Scales as a challenge for vulnerability assessment," *Natural Hazards*, vol. 55, no. 3, pp. 729–747, 2010.

[7] H.-M. Füssel, "How inequitable is the global distribution of responsibility, capability, and vulnerability to climate change: a comprehensive indicator-based assessment," *Global Environmental Change*, vol. 20, no. 4, pp. 597–611, 2010.

[8] C. M. Beier, T. M. Patterson, and F. S. Chapin III, "Ecosystem services and emergent vulnerability in managed ecosystems: A geospatial decision-support tool," *Ecosystems*, vol. 11, no. 6, pp. 923–938, 2008.

[9] D. R. Pathak and A. Hiratsuka, "An integrated GIS based fuzzy pattern recognition model to compute groundwater vulnerability index for decision making," *Journal of Hydro-Environment Research*, vol. 5, no. 1, pp. 63–77, 2011.

[10] K. J. Farn and S. K. Lin, "A study on information security management system evaluation-assets, threat and frangibility," *Computer Standards and Interfaces*, vol. 26, no. 6, pp. 501–513, 2004.

[11] P. Holme, B. J. Kim, C. N. Yoon, and S. K. Han, "Attack vulnerability of complex networks," *Physical Review E, Statistical, Nonlinear, and Soft Matter Physics*, vol. 65, no. 5, article 056109, 2002.

[12] M. E. J. Newman, S. Forrest, and J. Balthrop, "Email networks and the spread of computer viruses," *Physical Review E*, vol. 66, no. 3, article 035101, 2002.

[13] D. Magoni, "Tearing down the Internet," *IEEE Journal on Selected Areas in Communications*, vol. 21, no. 6, pp. 949–960, 2003.

[14] M. Ouyang, L. Hong, Z.-J. Mao, M.-H. Yu, and F. Qi, "A methodological approach to analyze vulnerability of interdependent infrastructures," *Simulation Modelling Practice and Theory*, vol. 17, no. 5, pp. 817–828, 2009.

[15] W. N. Adger, N. Brooks, G. Bentham, M. Agnew, and S. Eriksen, "New indicators of frangibility and adaptive capacity [R]," Tech. Rep. 7, Tyndall Centre for Climate Change Research, 2004.

[16] B. Dixon, "Groundwater vulnerability mapping: a GIS and fuzzy rule based integrated tool," *Applied Geography*, vol. 25, no. 4, pp. 327–347, 2005.

[17] R. Xiao, Y. Zhang, and Z. Huang, "Emergent computation of complex systems: A comprehensive review," *International Journal of Bio-Inspired Computation*, vol. 7, no. 2, pp. 75–97, 2015.

[18] G. Lu and J. Lu, "Introduction to the investigating in neural trust and multi agent systems," in *Examining Information Retrieval and Image Processing Paradigms in Multidisciplinary Contexts*, J. Lu and Q. Xu, Eds., pp. 269–273, IGI Global, Hershey, Pa, USA, 2017.

[19] D. Rosaci, G. M. L. Sarné, and S. Garruzzo, "Integrating trust measures in multiagent systems," *International Journal of Intelligent Systems*, vol. 27, no. 1, pp. 1–15, 2012.

[20] T. D. Huynh, N. R. Jennings, and N. R. Shadbolt, "An integrated trust and reputation model for open multi-agent systems," *Autonomous Agents and Multi-Agent Systems*, vol. 13, no. 2, pp. 119–154, 2006.

[21] D. Rosaci, "Trust measures for competitive agents," *Knowledge-Based Systems*, vol. 28, pp. 38–46, 2012.

[22] M. Wooldridge, *An Introduction to Multi-Agent Systems*, John Wiley & Sons, New York, NY, USA, 2009.

[23] H. P. Thadakamalla, U. N. Raghavan, S. Kumara, and R. Albert, "Survivability of multiagent-based supply networks: a topological perspective," *IEEE Intelligent Systems*, vol. 19, no. 5, pp. 24–31, 2004.

[24] A. Nair and J. M. Vidal, "Supply network topology and robustness against disruptions - An investigation using multi-agent model," *International Journal of Production Research*, vol. 49, no. 5, pp. 1391–1404, 2011.

[25] A. L. Barabási and R. Albert, "Emergence of scaling in random networks," *American Association for the Advancement of Science. Science*, vol. 286, no. 5439, pp. 509–512, 1999.

[26] D. Stauffer, "Difficulty for consensus in simultaneous opinion formation of Sznajd model," *Journal of Mathematical Sociology*, vol. 28, no. 1, pp. 25–33, 2004.

[27] Y. Zhang and R. Xiao, "Modeling and simulation of polarization in internet group opinions based on cellular automata," *Discrete Dynamics in Nature and Society*, vol. 2015, Article ID 140984, 14 pages, 2015.

[28] G. Deffuant, D. Neau, F. Amblard, and G. Weisbuch, "Mixing beliefs among interacting agents," *Advances in Complex Systems*, vol. 3, pp. 87–98, 2000.

[29] J. A. Holyst, K. Kacperski, and F. Schweitzer, "Phase transitions in social impact models of opinion formation," *Physica A*, vol. 285, no. 1-2, pp. 199–210, 2000.

[30] G. Szabó and G. Fáth, "Evolutionary games on graphs," *Physics Reports. A Review Section of Physics Letters*, vol. 446, no. 4-6, pp. 97–216, 2007.

[31] A. Pluchino, V. Latora, and A. Rapisarda, "Changing opinions in a changing world: a new perspective in sociophysics," *International Journal of Modern Physics C*, vol. 16, no. 4, pp. 515–531, 2005.

[32] A. Szolnoki, Z. Wang, and M. Perc, "Wisdom of groups promotes cooperation in evolutionary social dilemmas," *Scientific Reports*, vol. 2, article 576, 2012.

[33] H. Rauhut and J. Lorenz, "The wisdom of crowds in one mind: how individuals can simulate the knowledge of diverse societies to reach better decisions," *Journal of Mathematical Psychology*, vol. 55, no. 2, pp. 191–197, 2011.

[34] M. Perc, A. Szolnoki, and G. Szabó, "Restricted connections among distinguished players support cooperation," *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics*, vol. 78, no. 6, Article ID 066101, 2008.

[35] M. Mobilia, "Does a single zealot affect an infinite group of voters?" *Physical Review Letters*, vol. 91, article 028701, no. 2, 2003.

[36] A. Szolnoki, M. Perc, and M. Mobilia, "Facilitators on networks reveal optimal interplay between information exchange and reciprocity," *Physical Review E*, vol. 89, no. 4, Article ID 042802, 2014.

[37] F. Zambonelli and A. Omicini, "Challenges and research directions in agent-oriented software engineering," *Autonomous Agents and Multi-Agent Systems*, vol. 9, no. 3, pp. 253–283, 2004.

Hindawi
Complexity
Volume 2017, Article ID 5094218, 9 pages
https://doi.org/10.1155/2017/5094218

Research Article

Research on Behavior Model of Rumor Maker Based on System Dynamics

Xiaoqian Zhu and Fengming Liu

School of Management Science and Engineering, Shandong Normal University, Jinan 250014, China

Correspondence should be addressed to Fengming Liu; liufm@sdnu.edu.cn

Received 13 February 2017; Revised 7 May 2017; Accepted 16 May 2017; Published 21 June 2017

Academic Editor: Pasquale De Meo

Laws of rumor makers' behaviors are the root of curbing rumor and effective way to block rumor occurrence. Therefore, based on system dynamics model, this paper proposed the rumors behavior evolution model of rumor makers, aimed at discovering the laws of rumor makers' behaviors to achieve rumors blocking. First, by refining the driving factors in disinformation behavior, we constructed causal diagram of disinformation behavior evolution; secondly, by means of causal diagram, we constructed stock-flow diagram for quantitative analysis; finally, simulation experiment was carried out by using Vensim Personal Learning Edition software (Vensim PLE). The results showed that negative attitude is a major factor in the occurrence of disinformation behavior; personal factors are more pronounced than the factors of social and government on the impact of disinformation propensity score.

1. Introduction

As a kind of information, rumors can not only disturb people's daily life but also destroy economic development and social stability, and it must be strictly guarded and controlled. In addition to the spreaders which add fuel to the fire of rumors, there are lots of rumor makers who promote the spreading of rumors. Therefore, the study on behavior rules of rumor makers, which curbs the rumors at the root causes, becomes a hot spot for blocking the rumors.

At present, most of the research on behavior rules of rumor makers was conducted by means of statistical tools. Through a large number of historical data, they obtain the statistical characteristics of behavior rules and make a macroprediction of human behavior. Miritello [1] combs literature on statistics of human behavior in information dissemination and finds that the research method based on statistics of historical data is one of the effective methods to study the law of human behavior. Ma and others [2] confirmed that the tails of RT distributions exhibit power law behavior. Therefore, through collecting and analyzing broadcast data sent by 140 Twitter users, Salathé [3] and others based on the theory of psychology conclude that the emotions from friends and social awareness are highly correlated with individual information production behavior; that is, the emotions from neighbors have an important impact on information manufacturing behavior. Based on hierarchical temporal memory, Li [4] and others construct a cognitive model of rumors makers. They simulate the cognitive process of rumors in heterogeneous groups with different knowledge and personal experience and conclude that the manufacturing of rumor is driven by social cognitive factors. Moreover, system dynamics are a powerful tool for studying causality and can be used to analyze driving factors. Based on a system dynamics approach and the net anthropogenic N input (NANI) concept, a NANI-SD model [5] was developed to simulate the relationship between NANI and its drivers. Then, the system dynamics model developed in this study identified key factors influencing regional anthropogenic N input. Therefore, in order to discover rumor maker's behavior law, we will find behavioral driving factors of rumor maker and construct dynamics mode of disinformation behavior. It is an important method in the field of net rumors. This paper combs the related literature in the field of network rumor and excavates the behavioral drivers of rumor maker. Based on system dynamics theory, this paper puts forward

Table 1: Rumor-driven factors.

Factor categories	Included factors
Personal factors (PF)	Event attention (EA), personal discernment (PD), negative mentality (NM)
Social factors (SF)	Group polarization (GP), social trust (ST), mass discussion frequency (DF)
Governmental factors (GF)	Governmental regulation (GR), political activity (PA), dissemination efforts (DE)

the disinformation behavior evolution model of the rumor maker. The rumor maker's behavior evolution is a dynamic process influenced by many factors. This paper analyzes these factors from the personal factors, the social factors, and the governmental factors. In this paper, we use rumor tendency (RT) as a quantitative index to reflect the impact on information contacts. The higher the RT is, the higher the probability that the information contacts become rumor makers will be. And, the lower the RT is, the less likely that the information contacts become rumor makers will be.

2. Related Work

In the field of rumors driver, many scholars have conducted research and achieved a series of results; most of the research focused on the field of human psychology [6]. As early as 1945, American personality psychologists Allport and Postman suggested that any human demand can provide rumor for power [7]. Later, Difonzo and Bordia [8] considered that rumor transmission is motivated by three broad psychological motivations: fact-finding, relationship enhancement, and self-enhancement. Rumor is closely entwined with a host of social and organizational phenomena, including social cognition, attitude formation and maintenance, prejudice and stereotyping, group dynamics, interpersonal and intergroup relations, social influence, and organizational trust and communication. Through experiments, Ajzen and Fisbbein [9] confirmed the conclusion that behavioral intentions correlated significantly with behavior. During emergency events, individuals are exposed to large quantities of information without being aware of their validity or risk of misinformation, but users are usually swift to correct them, thus making the social media "self-regulating" [10]. Insofar as some people's behavior is controlled by custom and convention, it is a product of society, of the individual's interpretation of his role, and so, indirectly, of collective action [11].

The impact of the social environment on RT is also crucial. Through analytic derivation and simulations, Shaw and others [12] found that gossip destroys clustering in weakly clustered networks and increases cliquishness in networks with already high clustering. Hu and others [13] suggested that lower interpersonal influence of weak ties increases the isolation of social groups; thus, collectivism is unfavorable to the spread of participation across whole network, and they also demonstrated the importance of national culture on collective action. On the basis of previous studies, this paper will be discuss rumors-driven factors divided into three categories, as shown in Table 1.

3. System Dynamics Simulation Model of Rumor Behavior Evolution

In the micronetworks (WeChat, microblogging, etc.), all individuals must become information producers but not necessarily rumor makers. Social, psychological, personal, and other internal and external environmental factors could lead to the transformation of the information maker to the rumor maker; the impact of different factors is different. Therefore, mining driving factors and studying the extent of its impact, we could grasp the evolution laws of rumors, to achieve rumors blocking.

System dynamics are a useful tool for studying the causal relationship between factors. By analyzing the causal relationship between factors in the dynamic process of behavior evolution, the causal relationship diagram is constructed and the system dynamics modeling is implemented. Causal relationship diagram is a one-way complex network diagram composed of the influencing factors and the causal relationship among these factors, where the factor represents the node of the network; if there is a causal relationship between any two factors, there is a one-way edge between the two points that points to the result node by the cause node. If the result node changes in the same direction as the reason node, there is a positive causal chain between the two nodes. Otherwise it is called negative causal chain. When the causal chain is the same at the beginning and end, it forms a causal loop. At the same time, the polarity of the causal loop is determined by the number of positive and negative causal relationships in the loop.

3.1. Causal Analysis of the Disinformation Behavior Drivers. In the micronetworks, the behavior-driven factors of rumor makers are mainly composed by the personal factors, the social factors, and the governmental factors.

3.1.1. Personal Factors. The personal factors are determined by the heterogeneity of the individual in the network [3], which refers to the specific attributes of the individual, including the event attention, the personal discernment, and the negative mentality.

Event attention depends on age, occupation, education, the region's network coverage, and other inherent attributes of the individual. The personal discernment refers to the information judgment ability, the information processing ability, and the ability to overcome position bias for individual, and these abilities are restricted to individual learning, cognition, and experience. Negative mentality refers to the influence of network rumor psychological causes. Therefore, personal discernment and negative mentality are the direct factors of

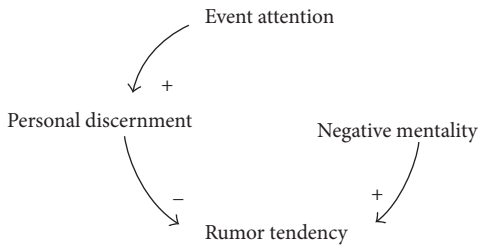


Figure 1: The personal factors causality diagram.

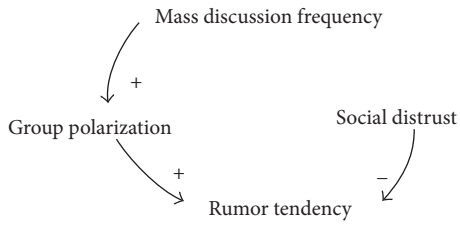


Figure 2: The social factors causal diagram.

RT, while event attention is indirectly affected by personal discernment.

The causality diagram of personal factors is shown in Figure 1.

3.1.2. Social Factors. Social factors refer to influencing factors from the surrounding groups, organizations, or media, including the group polarization, the social trust, and the mass discussion frequency.

Group polarization means that biased ideas of individual will produce more extreme negative effects when receiving the opinion of the group. Mass discussion frequency refers to the proportion of communication behavior between individual nodes, which will directly affect the influence scope and influence degree of group polarization. Social trust refers to the score of social trust, and low trust between members of the community easily leads to a crisis of confidence. In social factors, the social trust and the group polarization have a direct influence on RT.

Social factors causal diagram is shown in Figure 2.

3.1.3. Governmental Factors (GF). Governmental factors refer to the impact of government taking regulatory actions, including the governmental regulation, the political activity, and the dissemination efforts.

As a regulator of networks, the government is a balancing mechanism for disinformation behavior. Government regulation refers to the government's enforcement of existing laws and regulations, as well as the emergency mechanism for future events. Political activity refers to the degree of expression and participation in political affairs. Dissemination efforts refer to the objective expression of the opinions and the subjective acceptance of the audience. Political activity and dissemination efforts are the two-major government index. The high RT promotes the political activity,

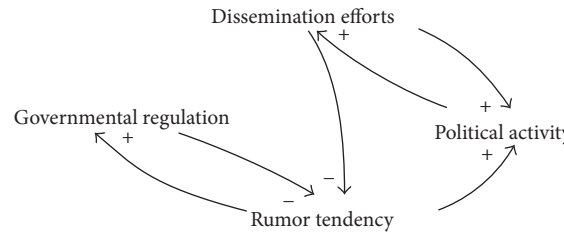


Figure 3: The governmental factors causal diagram.

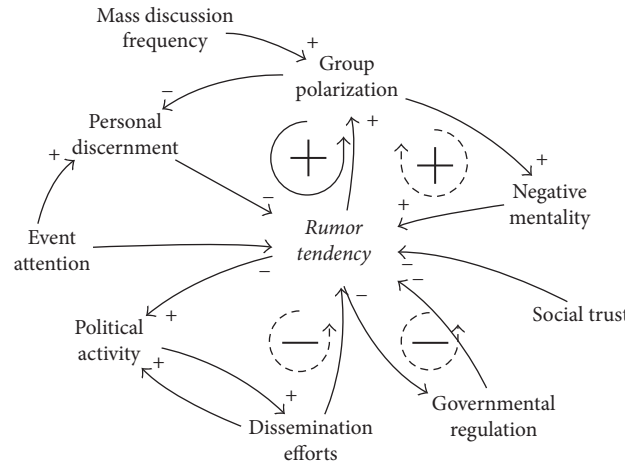


Figure 4: The disinformation behavior drivers causal-loop diagram.

strengthens the dissemination efforts, and thus reduces the RT. Finally, the balance of RT will be realized.

Figure 3 shows the governmental factor causality diagram.

3.1.4. Causal-Loop Diagram of the Disinformation Behavior Drivers. Figure 4 shows the causal-loop diagram of the disinformation behavior drivers. There are two positive-feedback loops and two negative-feedback loops.

(1) "RT" → "group polarization" → "personal discernment" → "RT" (positive-feedback loop 1)

(2) "RT" → "group polarization" → "negative mentality" → "RT" (positive-feedback loop 2)

Two positive-feedback loops indicate that RT is directly driven by two factors: the personal discernment and the negative mentality. Considering personal factors, the group polarization is indirect to the RT. Through the group polarization, RT also could weaken the personal discernment and contribute to negative mentality.

(3) "RT" → "political activity" → "dissemination efforts" → "RT" (negative-feedback loop 1)

(4) "RT" → "governmental regulation" → "RT" (negative-feedback loop 2)

As immature control techniques and unsound laws and regulations, the government often takes ex post measures of the management of network rumors. Only when rumors

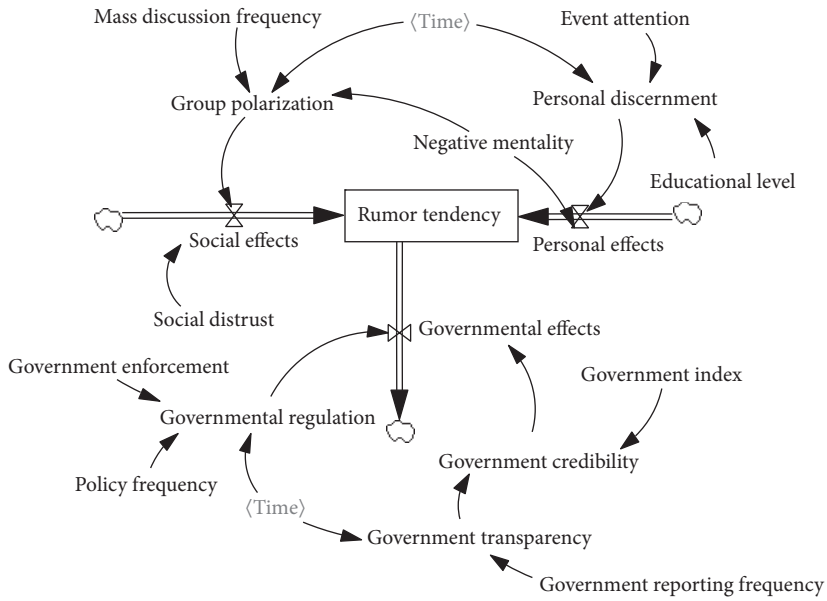


Figure 5: Flow-stock diagram of the disinformation behavior drivers.

endanger economic security and social stability will the government come forward to dispel rumors and strengthen supervision. The governmental regulation is not enough, and the RT will increase. In the society with high RT, strengthening the governmental regulation is helpful to safeguard government image and improve the credibility of the government, thus blocking the upward spiral of RT. The negative-feedback loop between the RT and the governmental regulation constitutes an important balance relationship. Therefore, government influence is a balance mechanism of disinformation behavior.

3.2. Flow-Stock Diagram of the Disinformation Behavior Drivers. The flow-stock diagram is another effective tool for system dynamics modeling. The causality diagram could qualitatively reflect the feedback structure of the system, that is, the causal relationship among the driving factors. Giving the variable dynamic accumulation, the flow-stock diagram is quantitative analysis of the system. In the stock-flow diagram, the stock in the feedback system represents the accumulation of the stock. The flow is represented by the change in stock over time, and the difference between the inflow and outflows produces the stock. In this paper, the RT is defined as the stock of disinformation behavior evolution model, and it is used to characterize the rumor state. Simultaneously, personal effects, social effects, and governmental effects are defined as the flow; these factors work together to influence the RT. The flow-stock diagram of the disinformation behavior drivers is shown in Figure 5.

3.2.1. Variable Formulas of the Disinformation Behavior Drivers

(1) Personal discernment = event attention (time) ∗ educational level

(2) Personal effects = **1**/{(**1** − negative mentality) ∗ personal discernment}

(3) Government transparency = government reporting frequency (time)

(4) Government credibility = a ∗ government index + b ∗ government transparency

(5) Governmental regulation = policy frequency (time) + government enforcement

(6) Governmental effects = c ∗ government credibility + d ∗ governmental regulation

(7) Group polarization = mass discussion frequency (time)/(1 − negative mentality)

(8) Social effects = social distrust ∗ group polarization

(9) Rumor tendency = integer (α ∗ social effects + β ∗ personal effects − θ ∗ governmental effects)

20 experts and sociologists which from the field of net rumors graded the importance of each factor, and we got the following conclusions. The weights of government index and government transparency are 0.3 and 0.7, respectively. And, government credibility and governmental regulation are obtained by the expert scoring method as, respectively, 0.7 and 0.3. The weight of social effects, personal effects, and governmental effects are 0.164, 0.583, and 0.253.

4. Simulation and Analysis of Model

4.1. Simulation of Disinformation Behavior Evolution Model. Based on the public opinion case statistics of China public sentiment network, this paper defines and initializes the model variables by means of table functions and expert scoring method. The event attention, the government reporting frequency, and the mass discussion frequency indicate,

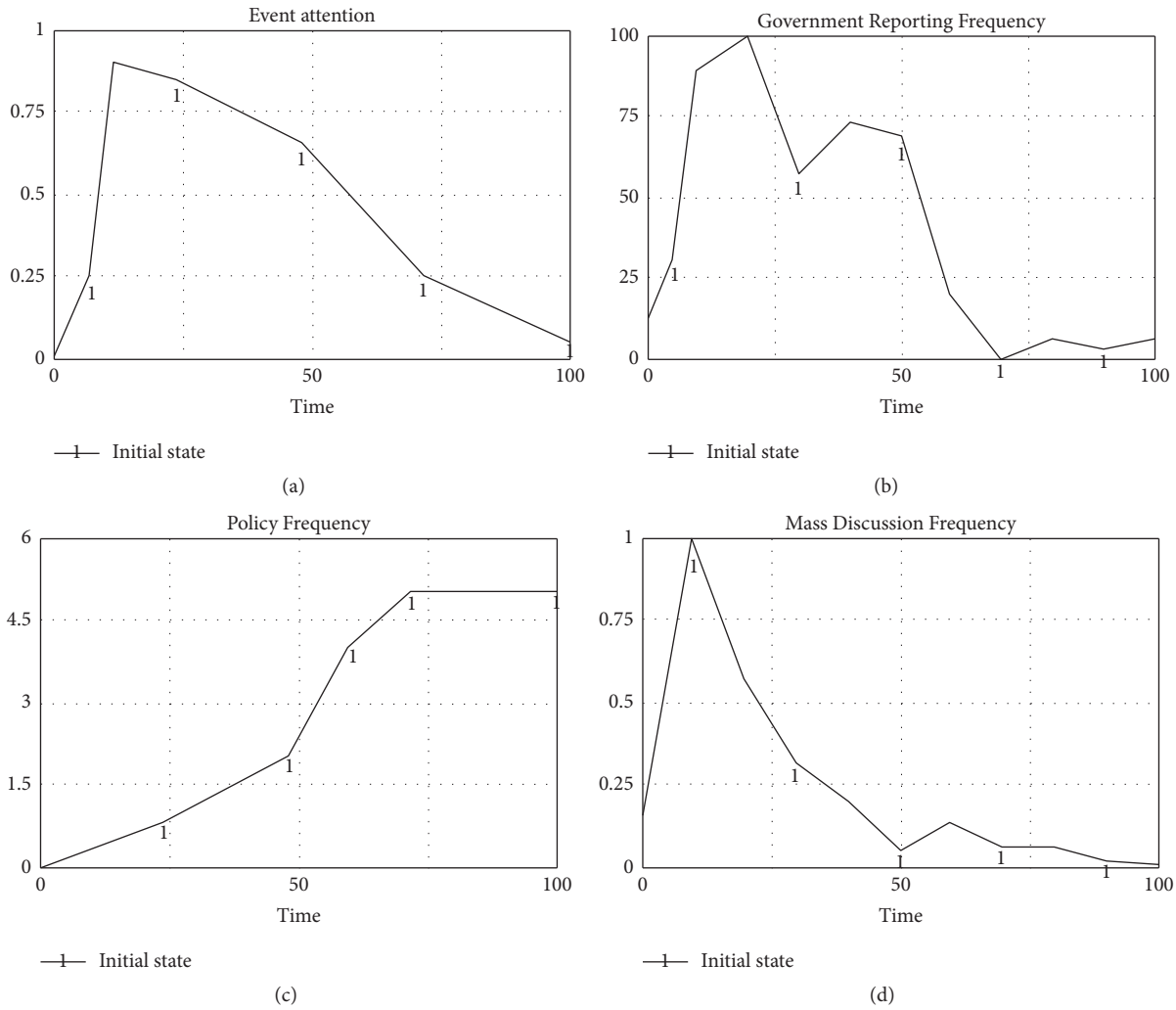


Figure 6: Table function representation of each variable (event attention shown in (a); government reporting frequency shown in (b); policy frequency shown in (c); mass discussion shown in (d)).

Table 2: Table function of event attention.

Time (hour)	0	7	12	24	48	72	100
Event attention frequency	0.01	0.25	0.9	0.85	0.65	0.25	0.05

Table 3: Table function of government reporting frequency.

Time (hour)	0	5	10	20	30	40	50	60	70	80	90	100
Government reporting frequency	12	30	89	100	57	73	69	20	0	6	3	6

respectively, the attention of the individual, the government, and the society to the hot issues (Figure 6). With the dynamic evolution, these variables often show irregular distribution. Therefore, this paper uses the table function method to express the above variables, as shown in Tables 2–5.

In the model, the government index, the government enforcement, the educational level, the negative mentality, and the social distrust are all constants. According to the expert scoring method and optimization of the simulation results, these constants' initial values are 80, 75, 45, 0.5, and 25, respectively, while the experimental results are more intuitive.

4.2. Analysis of Model Simulation Results

4.2.1. Rationality Analysis. According to the setting of the model parameters and the establishment of the variable formula, the simulation diagram of the RT is obtained. As can be seen from Figure 7, we know that the RT is to change dynamically over time. In the early stage of disinformation

Table 4: Table function of policy frequency.

Time (hour)	0	24	48	60	72	100
Policy frequency	12	30	89	100	57	73

Table 5: Table function of mass discussion frequency.

Time (hour)	0	10	20	30	40	50	60	70	80	90	100
Mass discussion frequency	0.157	0.997	0.57	0.312	0.194	0.045	0.134	0.054	0.06	0.017	0.01

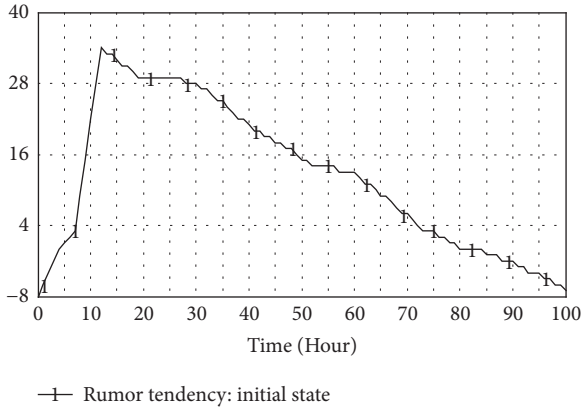


Figure 7: The rumor tendency graphs.

behavior, the RT will increase significantly. When reaching the peak, it begins to decrease slowly and finally returns to the initial state.

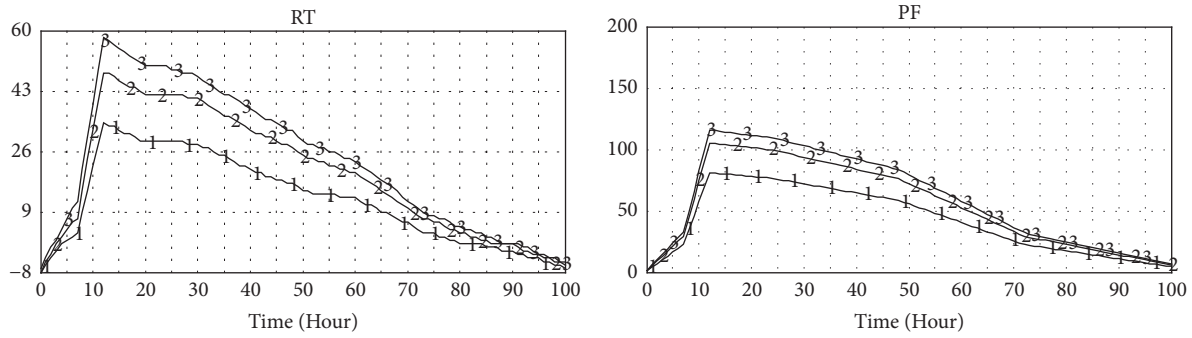
Because of the parametric hypothesis, the simulation results could not be completely consistent with the actual results. However, in the early stage of disinformation behavior, people will question or even contradict it. At this point, the RT is negative. With the increasing attention to the incident, RT is rising rapidly. When the rumors have finished disinformation, their attention to the incident will slowly reduce. And ultimately, they lose interest and unsubscribe it. It is concluded that the disinformation behavior evolution model is reasonable and could be used to simulate the disinformation behavior.

4.2.2. Sensitivity Analysis. Sensitivity analysis is that we could observe the changes of amount of dependent variable by changing the specific variables. And we could analyze the effect of variables on the dependent variable. By, respectively, changing the government index, the government enforcement, the educational level, the negative mentality, and the social distrust, we could observe the change of the RT and find out the main cause of disinformation behavior.

4.2.3. Personal Factors. In personal factors, the educational level (curve 2) and the negative mentality (curve 3) were increased by 30%. The change of the RT is shown in Figure 8(a). The influence of negative mentality on RT significantly is higher than the education level; that is, an information contact with negative mentality is more likely to become a rumor maker. And, the Indian psychologist Beside has also raised the fact that the unrest is one of the motivations of rumors [7]. At the same time, when raising the educational level, RT tends to increase. This indicates that the higher the educational level, the higher the possibility of rumors. Figure 8(b) shows the effect of personal factors on the personal effects. (Curve 1 shows the initial state dynamic changes of disinformation tendency and is used to compare the changes in each factor.)

4.2.4. Social Factors. In social factors, this paper observes the impact of social factors on the RT, by increasing the initial state of social distrust (curve 2) by 30%, as shown in Figure 9(a). The influence of social distrust on the RT is more significant; that is, social distrust will promote rumors. However, the social trust and the social distrust are reverse-changed. Therefore, improving social trust can effectively control the rumors. Figure 9(b) shows the impact of social distrust on the social effects. From curve 2, we can see that the effect of social distrust on the rumor has a significant impact in the initial period. At this point, misleading information from the social environment and surrounding friends can directly affect behavioral choices of information contactors. And the contactor is most likely to become a rumor maker in the early stages.

4.2.5. Governmental Factors. In government factor, this paper reduces government enforcement (curve 2) and government index (curve 3) by 30%, as shown in Figure 10(a). Government is a balancing act of rumors and should have commensurate influence and control. However, the enforcement and government index on the role of RT are not significant. At present, the real-name operations cannot be covered in the whole network, and human beings whose the real identity is not informed are prone to crime. That is to say, the anonymity of the networking and rumors of sudden and other characteristics lead to more complex network environment. The government often takes action after the rumor broke out, which causes government supervision and management on the network is difficult to achieve, especially for the disinformation behavior. Due to the "hysteresis" of the punishments, the deterrent effect of the government on the rumors is ignored. In Figure 10(b), with the same magnitude of decline, the enforcement for government impact is more significant. At the same time, unlike personal factors and social factors, the impact of government has effects throughout the whole rumor behavior evolution process.



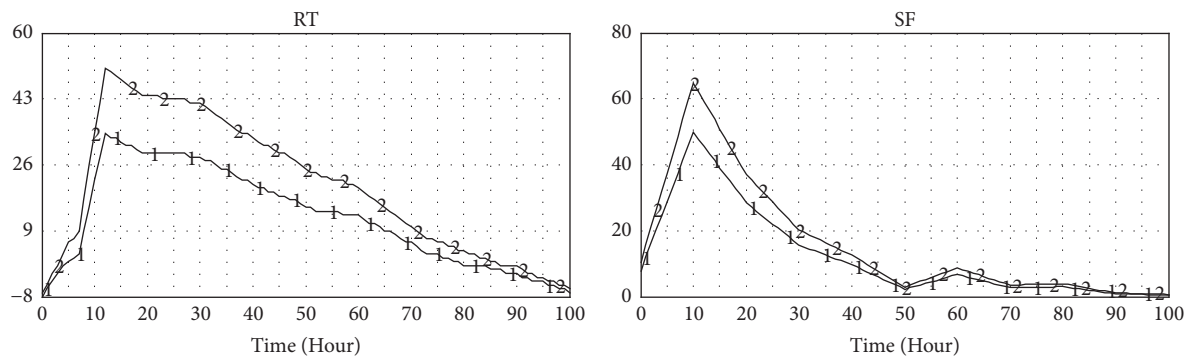
1 RT: initial state
2 RT: 30% increase in education level
3 RT: 30% increase in negative mentality

(a) The effect of personal factors on the rumor tendency

1 PF: initial state
2 PF: 30% increase in education level
3 PF: 30% increase in negative mentality

(b) The effect of personal factors on the personal effects

Figure 8



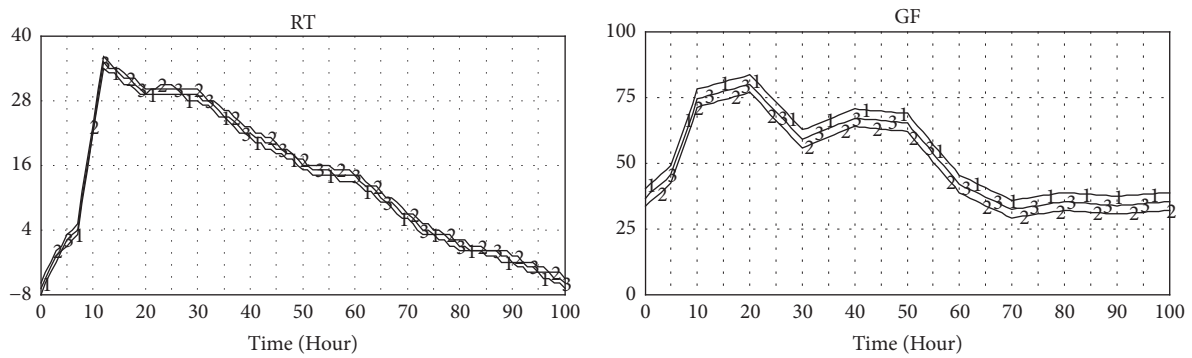
1 RT: initial state
2 RT: 30% increase in social distrust

(a) The effect of social factors on rumor tendency

1 SF: initial state
2 SF: 30% increase in social distrust

(b) The effect of social factors on the social effects

Figure 9



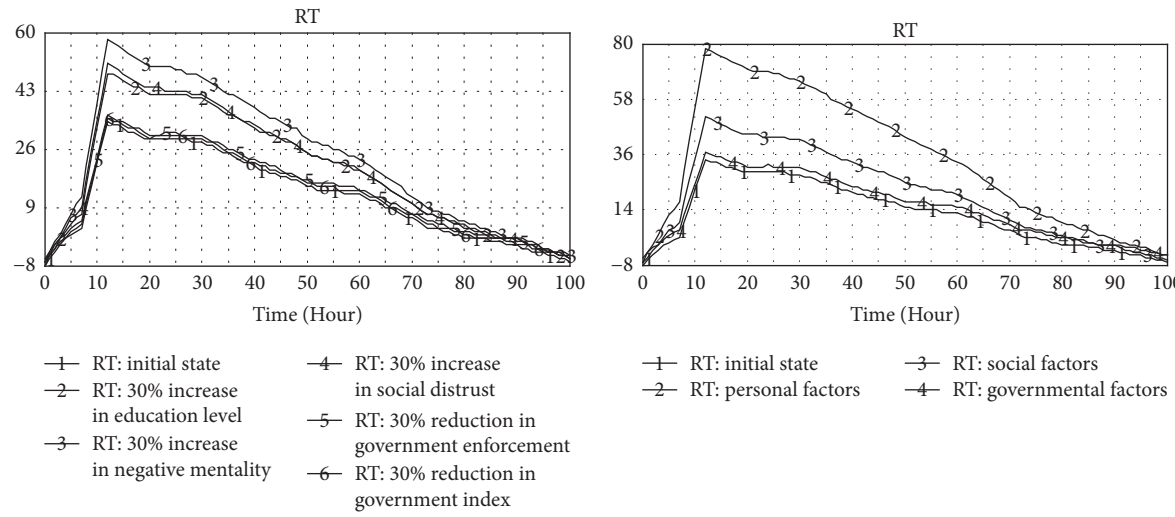
1 RT: initial state
2 RT: 30% reduction in government enforcement
3 RT: 30% reduction in government index

(a) The effect of governmental factors on rumor tendency

1 GF: initial state
2 GF: 30% reduction in government enforcement
3 GF: 30% reduction in government index

(b) The effect of governmental factors on governmental effects

Figure 10



(a) Analysis on the driving factors

(b) The influence of various subsystems on rumor tendency

Figure 11

4.2.6. Comprehensive Factor Analysis. In Figure 11(a), we combine the various factors of three models. Under the same amplitude changes, the influence of negative mentality on the RT is the most significant. Changes in negative mentality are most sensitive to rumors. And negative mentality is the main motivation in promoting the disinformation behavior. As shown in Figure 11(b), the effect of personal effects on RT is the most significant, followed by social effects. The government effects have only a subtle effect on the rumor tendency. Therefore, personal factors are the key factors of disinformation behavior.

5. Conclusions

Network rumors endanger national security and social stability. The traditional network rumor propagation model aims to achieve blocking and governance of rumors. Their object is existing and destructive network rumors. However, by the system dynamics, this paper puts forward disinformation behavior evolution model of the rumor maker. This mode solves network rumors from the origin and provides a basis for monitoring and early warning of network rumors. The model is simulated from three aspects, individual, society, and government, and draws the following conclusions.

The influence of personal factors on RT is the most significant. Negative mentality is the main cause of disinformation behavior. The more the negative mentality of information contacts, the greater the possibility of disinformation behavior. Therefore, improving people's better life index and reducing social instability are conducive to reducing the negative mentality of the masses and controlling rumors. For the government, the key to reducing the RT is the government's control efforts and measures introduced efficiency. Strengthen the rumor punishment mechanism; improve the emergency response to emergencies. Through the "micro" platform, E-government could achieve network guidance, mass interaction, and so forth and, then, could standardize the network environment.

Conflicts of Interest

The authors declare no competing financial interests.

Authors' Contributions

Zhu Xiao-Qian and Liu Feng-Ming wrote the main manuscript text and Zhu Xiao-Qian prepared the experiments. All authors reviewed the manuscript.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (no. 61170038, 61472231), the National Social Science Foundation of China (no. 14BTQ049), and a Project of International Cooperation in Training of Excellent Backbone Teachers for Advanced University in Shandong Province.

References

[1] G. Miritello, *Temporal Patterns of Communication in Social Networks*, Springer International Publishing, 2013.

[2] T. Ma, J. G. John, and R. A. Holden, "Distribution of human response times," *Complexity*, vol. 21, no. 6, pp. 61–69, 2016.

[3] M. Salathé, D. Q. Vu, S. Khandelwal, and D. R. Hunter, "The dynamics of health behavior sentiments on a large online social network," *EPJ Data Science*, vol. 2, no. 1, pp. 1–12, 2013.

[4] X. Li, X. Chen, and W. Wang, "Life system modeling and simulation: international conference on life system modeling and simulation, Lsms," in *Proceedings of the International Conference on Intelligent Computing for Sustainable Energy and Environment (Icsee '14)*, pp. 268–277, Springer Berlin Heidelberg, Shanghai , China, September 2014.

[5] W. Gao, B. Hong, D. P. Swaney, R. W. Howarth, and H. Guo, "A system dynamics model for managing regional N inputs from human activities," *Ecological Modelling*, vol. 322, pp. 82–91, 2016.

[6] P. E. Smaldino and J. C. Schank, "Human mate choice is a complex system," *Complexity*, vol. 17, no. 5, pp. 11–22, 2012.

[7] G. W. Allport and L. J. Postman, "Section of psychology: the basic psychology of rumor," *Transactions of the New York Academy of Sciences*, vol. 8, pp. 61–81, 1945.

[8] N. Difonzo and P. Bordia, *Rumor Psychology: Social and Organizational Approaches*, American Psychological Association, Washington, DC, USA, 2007.

[9] I. Ajzen and M. Fisbbein, "Factors Influencing Intentions and the Intention-Behavior Relation," *Human Relations*, vol. 27, no. 1, pp. 1–15, 1974.

[10] T. Simon, A. Goldberg, and B. Adini, "Socializing in emergencies-a review of the use of social media in emergency situations," *International Journal of Information Management*, vol. 35, no. 5, pp. 609–619, 2015.

[11] R. E. Park, "Human nature and collective behavior," *American Journal of Sociology*, vol. 32, no. 5, pp. 733–741, 1927.

[12] A. K. Shaw, M. Tsvetkova, and R. Daneshvar, "The effect of gossip on social networks," *Complexity*, vol. 16, no. 4, pp. 39–47, 2011.

[13] H.-H. Hu, J. Lin, and W. Cui, "Cultural differences and collective action: a social network perspective," *Complexity*, vol. 20, no. 4, pp. 68–77, 2015.

Hindawi
Complexity
Volume 2017, Article ID 4215805, 13 pages
https://doi.org/10.1155/2017/4215805

Research Article

Evolution of the Chinese Industry-University-Research Collaborative Innovation System

Jianyu Zhao[1,2] and Guangdong Wu[3]

[1]*School of Economics and Management, Harbin Engineering University, Heilongjiang, Harbin 150001, China*
[2]*School of Management, Harbin Institute of Technology, Heilongjiang, Harbin 150001, China*
[3]*School of Tourism and Urban Management, Jiangxi University of Finance and Economics, Nanchang 330013, China*

Correspondence should be addressed to Jianyu Zhao; jianyu64@sina.com

Received 28 November 2016; Revised 19 March 2017; Accepted 29 March 2017; Published 9 April 2017

Academic Editor: Katarzyna Musial

The goal of this study was to reveal the mechanism of the Chinese industry-university-research collaborative innovation (IURCI) and interactions between the elements in the system and find issues that exist in the collaborative innovation process. Based on the theoretical perspective of innovation and complexity science, we summarized the elements of the IURCI as innovation capability, research and development (R&D) configuration, and knowledge transfer and established a theoretical model to describe the evolution of the IURCI system. We used simulation technology to determine the interactions among variables and the evolution trend of the system. The results showed that the R&D configuration can promote the evolution of innovation capability and knowledge transfer and that innovation capacity is the current dominant factor in the evolution of the Chinese IURCI system and is highly positively correlated with R&D configuration. The evolutionary trend of knowledge transfer was gentler, and its contribution to the evolution of the Chinese IURCI system was less than that of R&D configuration. When innovation, R&D configuration, and knowledge transfer are relatively balanced, the collaborative innovation system can achieve high speed and stable evolution.

1. Introduction

Innovation is an important stimulator of economic development and is also a key element in the global competitiveness of the country. As the world's second largest economy, China has made great progress along the road of independent innovation, research, and development investment, and the number of academic achievements and patents has been ranked at the top in the world. But in spite of this progress, there is a disconnection between the economy and technology in the process of innovation in China. On the one hand, the *Global Competitiveness Report* developed by the World Economic Forum according to the Global Competitiveness Index (GCI) showed that China's technological readiness for innovation (ranked at #88) has seriously hampered the country's competitiveness ranking, indicating that the innovation capability of core technology in Chinese enterprises is still relatively backward, failing to form an innovation-driven development model, and many industry's core technologies with significantly shorter cycles of innovation are still heavily dependent on foreign countries. On the other hand, the conversion rate of technological achievements has been low for a long time in Chinese universities and research institutions; therefore, higher education training is lagging behind (ranked #62). Universities and research institutions cannot effectively meet the knowledge requirements for innovation, which is also an important factor that has led to delays in the Chinese innovation system. The *Organization for Economic Co-operation and Development's China Innovation Policy Research Report* pointed out that coordination and integration in China's innovation system is not perfect, and the synergy between constituent subjects in the system is low. Compared with developed countries, Chinese enterprises not only lack the R&D capabilities of core technology, but also the effects of knowledge accumulation are relatively poor. Enterprises tend to be more cost-oriented and lack the motivation to carry out and use public research achievements. Meanwhile, enterprises, universities, and research

institutes rarely share innovation resources, most types of technology transfer are carried out under the guidance of government, and universities and research institutions do not take the initiative to understand the technology needs of industry. These problems have seriously hampered knowledge spillover in the Chinese innovation system and have become an obstacle that China must overcome to build an innovation-driven country through independent innovation.

An effective measure to solve the above problem is to establish a practical and effective Chinese cooperative research innovation system, thus contributing to the rapid transformation of public scientific and technological achievements of Chinese enterprises and universities as well as research institutions, and to promote scientific research in Chinese universities and institutes that feeds the demand for industrial innovation, thus allowing technological development and industrial development to move forward together. Research on the evolution of the Chinese industry-university-research (IUR) collaborative innovation (IURCI) system can help identify the interactions between the elements of Chinese IURCI systems and, through identifying problems in the process of collaborative innovation, can help Chinese enterprises and universities as well as research institutions emerge from the knowledge dilemma of collaborative innovation. This study therefore has theoretical and practical value for enhancing collaborative innovation efficiency and promoting IURCI development in China.

In contrast to the past static perspective, we established an evolutionary logistics/dynamics equation to describe the research collaborative innovation system in China, using relevant methods from game theory to solve it. On this basis, we collected indices and data that have influenced Chinese IURCI from 2005 to 2014, simulated the evolution morphology of related variables by MATLAB software, analyzed the interactions between elements and dynamic evolution, and demonstrated in detail the evolution mechanism of the research innovation system in China. In this study, while revealing the essence of Chinese IURCI, we have tried to establish a new research framework and explain 2 issues: first, there are different degrees of interaction among variables in the Chinese IURCI system, with different levels of contribution to the evolution of a collaborative innovation system and, second, when the default initial value of evolution changes, an evolutionary trend develops among the variables.

In Section 2 of this paper, we describe our analysis of the constituent elements of the IURCI system and the synergy principle. On this basis, in Section 3, we describe how we established an IURCI system evolution model with a focus on innovation capacity, R&D configuration, and knowledge transfer, and conducted derivation and analysis. In Section 4, we conducted a simulation using actual data of relevant parameters and analyzed the results. Finally, in the conclusion, we discuss the innovation and contribution of research achievements and propose the main direction for future research.

2. Theory

IURCI refers to enterprises, universities, and research institutes, 3 main users of innovation to expand their resources and capabilities, which jointly develop technology innovation activities under the support of government, science and technology intermediary service agencies, financial institutions, and other relevant organizations [1, 2]. In the process of IURCI, enterprises, universities, and research institutes convert science and technology into practical, productive forces for the purpose of innovation, based on a clear division of functions, through complex nonlinear interactions. This realizes mutual benefits between enterprises and universities as well as research institutes and produces a synergistic innovation impact that each factor cannot achieve alone [3]. Canhoto et al. [4] believe that the essence of collaboration is the complementary use of resources and capabilities between competitive enterprises and universities as well as research institutions. The advantages of enterprises include the rapid commercialization of technology, relatively adequate innovation funding, suitable production and test equipment and sites, and market information and marketing experience, and their needs for innovation are in basic principles of knowledge and in scientific as well as technical human resources [5–7]. The advantages of universities and research institutions are theoretical research, professionals, scientific equipment, knowledge and technical information, and research methods and experience, and their needs for innovation are resource support and practical information [8–10]. The needs of enterprises for innovation in knowledge resources and the needs of universities and research institutions for spreading of scientific knowledge and practice demand constitute the IURCI system based on a retrieval mechanism and allocation rule for noncompetitive interests [11–13].

Improving collaborative innovation system performance is a strategic goal, and, from the 1980s up to the present, there have been theories such as the innovation systems theory [14–16], triple helix [17–20], and open innovation theory [21–24] which have discussed the elements and principles of an IURCI system at different levels [25, 26]. The innovation system theory is that, in the process of collaborative innovation and research, knowledge reformation is the key to enhance collaborative innovation performance [27–29]. This reformation mainly relies on the resource investment level and the specificity of the R&D configuration and on knowledge accessibility as well as access of universities and research institutions to enterprises. In fact, since technological innovation has become the key to business competition, more and more enterprises have started to pursue the development of industrial common technology or cutting-edge technologies. This development, on the one hand, stems from the enhanced ability of an R&D configuration to promote an innovation system, and, on the other hand, it results from knowledge flowing from the innovation platform of enterprises as they cooperate with universities and research institutions, that is, knowledge transfer within the IURCI system [30, 31]. Using resource dependence theory and knowledge management theory, the microperspective analysis of the Triple Helix Model found that the nature of evolution of the IURCI system

involves nonlinear interactions among knowledge transfer, R&D configuration, and innovation capability [32–34]. Further, the open innovation theory [35, 36], which looked at the aspects of fitness and openness between subjects, confirmed that R&D resource configuration is the basic premise for IURCI system evolution, and knowledge transfer is, within the constraints of transaction cost law, in the common interests of companies and universities well as research institutions. We therefore believe that the evolution of the IURCI system depends on innovation, R&D configuration, and knowledge transfer, and the interactions among the 3 not only surpass the general innovation paradigm of evolution in previous evolutionary economics but also represent the core element to decide and change collaborative innovation system performance.

IURCI is a complex social system, and the interactions among those constituent elements are the premise to enhance collaborative innovation performance. Innovation, R&D configuration, and knowledge transfer become the key to determine the evolution of a collaborative innovation system and determine its performance. Therefore, to analyze the IURCI mechanism, we use the theory as well as method of Synthetics [37], which illustrates the interactions among elements in complex system and establish the logistic equation which analyzes and explains the interactions among innovation capacity, R&D configuration, and knowledge transfer. We need to demonstrate precisely and comprehensively the interactions among the 3 elements in the collaborative innovation process, innovation, R&D configuration, and knowledge transfer, thus revealing the nature of IURCI.

3. Model

3.1. Model Establishment

3.1.1. Logistic Evolution Equation of Innovation Capacity. Faced with the fact of tight resources, in order to improve innovation performance, universities and industry are bound to demand cooperation. Knowledge transfer and flow in the IURCI system eventually lead to improved innovation capabilities in the system. Meanwhile, the IURCI system, in order to achieve higher innovation performance, will continue to increase efforts to improve R&D configuration, which to some extent will promote innovation capacity, and thereby the evolution equation of innovation capacity is written as follows:

$$\frac{dn_1}{dt} = \alpha_1 n_1 + \beta_1 n_1 n_3 + \gamma_1 n_2. \tag{1}$$

In formula (1), α_1 represents the influence of coefficient of innovation n_1 itself and $\beta_1 n_3$ stands for the influence factor of R&D configuration on innovation capacity, namely, the impact of increasing allocation of R&D configuration on innovation capacity. Under normal circumstances $\beta_1 > 0$, γ_1 is the influence factor of knowledge transfer and represents the interaction between knowledge transfer and R&D configuration.

3.1.2. Logistic Evolution Equation of R&D Configuration. Improved innovation capacity of the IURCI system will appeal to the willingness of businesses, universities, and research institutes to improve R&D configuration; therefore, innovative ability n_1 is also an influencing factor of R&D configurations n_2. Meanwhile, in the process of knowledge transfer in the IURCI system, enterprises will continuously increase resources investment in R&D in order to continuously create and achieve high-value heterogeneity knowledge, and we can thereby establish the evolution equation of R&D configuration as follows:

$$\frac{dn_2}{dt} = -\alpha_2 n_2 + \beta_2 n_1 n_2 + \gamma_2 n_3. \tag{2}$$

In formula (2), $-\alpha_2$ represents the influence coefficient of the R&D configuration itself. $\beta_2 n_1$ represents the influence coefficient of innovation capability on R&D configuration. Because there is positive feedback between R&D configuration n_2 and innovation capability n_1, the coefficient β_2 is also positive. Finally, γ_2 represents the impact size of knowledge transfer on R&D configuration.

3.1.3. Logistic Evolution Equation of Knowledge Transfer. Knowledge transfer between enterprises and universities as well as research institutions can have an impact on the innovation capacity in the IURCI system. Meanwhile, as the R&D configuration increases, the needs of the IURCI system for innovation and new knowledge continuously increase, thereby facilitating improvement in the knowledge transfer capability of the IURCI system. This demonstrated that there is a positive correlation between the 2 variables, knowledge transfer and R&D configuration, and thereby we can establish the evolution equation of knowledge transfer; thus,

$$\frac{dn_3}{dt} = \alpha_3 n_3 + \beta_3 n_2. \tag{3}$$

In formula (3), α_3 represents the influence coefficient of knowledge transfer itself and β_3 represents the impact degree of R&D configuration n_2 on knowledge transfer capacity.

Using innovation capacity, R&D configuration, and knowledge transfer of IUR as the 3 variables and using a simultaneous logistics evolution equation, we obtained a dynamic evolution model as follows:

$$\begin{aligned} \frac{dn_1}{dt} &= \alpha_1 n_1 + \beta_1 n_1 n_3 + \gamma_1 n_2, \\ \frac{dn_2}{dt} &= -\alpha_2 n_2 + \beta_2 n_1 n_2 + \gamma_2 n_3, \\ \frac{dn_3}{dt} &= \alpha_3 n_3 + \beta_3 n_2. \end{aligned} \tag{4}$$

In formula (4), n_1, n_2, n_3 are constants, and the definitions of α, β, γ are as follows.

$\alpha = \sqrt[i]{\prod_{i=1}^{p} \alpha_i}$ $(i = 1, 2, 3, \ldots, p)$ is an adjustment parameter of the state variable n_1, corresponding to an innovation capability index, wherein α_i is the resulting parameter after

conversion of each index in the evaluation system, wherein α_i is the resulting parameter after conversion of each index in innovation capacity evaluation system.

$\beta = \sqrt[i]{\prod_{i=1}^{p} \beta_i}$ $(i = 1, 2, 3, \ldots, p)$ is an adjustment parameter of the state variable n_2, corresponding to the R&D configuration index, wherein β_i is the resulting parameter after conversion of each index in R&D configuration evaluation system.

$\gamma = \sqrt[i]{\prod_{i=1}^{p} \gamma_i}$ $(i = 1, 2, 3, \ldots, p)$ is an adjustment parameter of the state variable n_3, corresponding to the knowledge transfer index, wherein γ_i is the resulting parameter after conversion of each index in knowledge transfer evaluation system.

3.2. Model Analysis

3.2.1. Linearization. According to Krasovskii's method [38], we used a gradient vector matrix in the operation system to solve the stable solution of evolution model, wherein the systematic coefficient matrix was solved using the Taylor Formula [39]. After performing Taylor, omitting higher-order terms of a second and higher order, we then obtained the linearized coefficient matrix M (Jacobian matrix) as follows:

$$M = \nabla F = \begin{bmatrix} \dfrac{\partial f_1(X)}{\partial x_1} & \cdots & \cdots & \dfrac{\partial f_1(X)}{\partial x_n} \\ \dfrac{\partial f_2(X)}{\partial x_1} & \ddots & \cdots & \vdots \\ \vdots & \vdots & & \vdots \\ \dfrac{\partial f_n(X)}{\partial x_1} & \cdots & \cdots & \dfrac{\partial f_n(X)}{\partial x_n} \end{bmatrix}. \tag{5}$$

Thus, according to Krasovskii's method, we rewrite formula (4) in the form of a matrix multiplication $\dot{X} = M \cdot X$ and obtain the following:

$$\begin{bmatrix} \dot{x}_1 \\ \vdots \\ \dot{x}_n \end{bmatrix} = \begin{bmatrix} \dfrac{\partial f_1(X)}{\partial x_1} & \cdots & \dfrac{\partial f_1(X)}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \dfrac{\partial f_n(X)}{\partial x_1} & \cdots & \dfrac{\partial f_n(X)}{\partial x_n} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}. \tag{6}$$

Formula (6) represents the initial state of the IURCI system with 0 input. We can use linear system theory to obtain the time domain expression of state variable X, that is, the solution of system variable X after linearization of the evolution equation.

For evolution equations, we solved using the Laplace transformation [40], and formula (4) was converted to the following:

$$\begin{aligned} sX(s) - X_0 &= M \cdot X(s), \\ (sI - M) X(s) &= X_0. \end{aligned} \tag{7}$$

Laplace transformation of X is obtained as follows:

$$X(s) = (sI - M)^{-1} X_0. \tag{8}$$

Thus we obtained an expression of the time domain of X as follows:

$$X = L^{-1}\left((sI - M)^{-1} X_0\right). \tag{9}$$

In formula (9), s is a pull complex symbol, X_0 represents the initial state of the matrix, $X(s)$ is the pull transformation of $X(t)$, $(sI - M)^{-1}$ represents the inverse matrix of matrix $(sI - M)$, and L^{-1} is the anti-Laplace transformation.

3.2.2. Stable Solution. According to the linearization results, formula (4) is solved to obtain the Jacobian matrix as follows:

$$\begin{aligned} J &= \begin{bmatrix} \dfrac{\partial f_1}{\partial n_1} & \dfrac{\partial f_1}{\partial n_2} & \dfrac{\partial f_1}{\partial n_3} \\ \dfrac{\partial f_2}{\partial n_1} & \dfrac{\partial f_2}{\partial n_2} & \dfrac{\partial f_2}{\partial n_3} \\ \dfrac{\partial f_3}{\partial n_1} & \dfrac{\partial f_3}{\partial n_2} & \dfrac{\partial f_3}{\partial n_3} \end{bmatrix} \\ &= \begin{bmatrix} \alpha_1 + \beta_1 n_3 & \gamma_1 & \beta_1 n_1 \\ \beta_2 n_2 & -\alpha_2 + \beta_2 n_1 & \gamma_2 \\ 0 & \beta_3 & \alpha_3 \end{bmatrix} \end{aligned} \tag{10}$$

to obtain

$$|J| = \begin{vmatrix} \alpha_1 + \beta_1 n_3 & \gamma_1 & \beta_1 n_1 \\ \beta_2 n_2 & -\alpha_2 + \beta_2 n_1 & \gamma_2 \\ 0 & \beta_3 & \alpha_3 \end{vmatrix}. \tag{11}$$

Meanwhile, according to the result of formula (11) of Jacobian matrix, we set $dn_1/dt = 0$; $dn_2/dt = 0$; $dn_3/dt = 0$, and formula (12) was obtained as follows:

$$\begin{aligned} \alpha_1 n_1 + \beta_1 n_1 n_3 + \gamma_1 n_2 &= 0, \\ -\alpha_2 n_2 + \beta_2 n_1 n_2 + \gamma_2 n_3 &= 0, \\ \alpha_3 n_3 + \beta_3 n_2 &= 0. \end{aligned} \tag{12}$$

According to this equation, we use Routh-Hurwitz stability criterion [41] and the stable solution of complex system to solve formula (12), so that a partial equilibrium point can be obtained as follows:

$$\begin{aligned} &A(0, 0, 0) \\ &B\left(\frac{\alpha_2\alpha_3 + \beta_3\gamma_2}{\alpha_3\beta_2}, \frac{\alpha_1\alpha_2\alpha_3^2 + \alpha_1\alpha_3\beta_2\gamma_2}{\alpha_3^2\beta_2\gamma_1 - \alpha_2\alpha_3\beta_1\beta_3 - \beta_1\beta_3^2\gamma_2}, \right. \\ &\left. \frac{\alpha_1\alpha_2\alpha_3\beta_3 + \alpha_1\beta_2\beta_3\gamma_2}{\alpha_2\alpha_3\beta_1\beta_3 + \beta_1\beta_3^2\gamma_2 - \alpha_3^2\beta_2\gamma_1}\right). \end{aligned} \tag{13}$$

Bringing the equilibrium points A and B, respectively, into the Jacobian matrix, and judging equilibrium according

to the trace and matrix determinant symbols of the Jacobian matrix, we obtained the following:

$$\mathrm{tr}\,(J) = \alpha_3 + \beta_2 n_1 + \beta_1 n_1 - \alpha_1 - \alpha_2. \tag{14}$$

For equilibrium point A,

$$|J| = \begin{vmatrix} \alpha_1 & \gamma_1 & 0 \\ 0 & -\alpha_2 & \gamma_2 \\ 0 & \beta_3 & \alpha_3 \end{vmatrix}. \tag{15}$$

$$\mathrm{tr}\,(J) = \alpha_3 - \alpha_1 - \alpha_2,$$

For equilibrium point B,

$$\mathrm{tr}\,(J) = \alpha_3 + \alpha_1 + \frac{\beta_3\gamma_2}{\alpha_3} + \frac{\alpha_1\alpha_2\alpha_3\beta_1\beta_3 + \alpha_1\beta_1\beta_2\beta_3\gamma_2}{\alpha_2\alpha_3\beta_1\beta_3 + \beta_1\beta_3^2\gamma_2 - \alpha_3^2\beta_2\gamma_1}$$

$$|J| = \begin{vmatrix} \dfrac{2\alpha_1\alpha_2\alpha_3\beta_1\beta_3 + \alpha_1\beta_1\beta_2\beta_3\gamma_2 + \alpha_1\beta_1\beta_3^2\gamma_2 - \alpha_1\alpha_3^2\beta_2\gamma_1}{\alpha_2\alpha_3\beta_1\beta_3 + \beta_1\beta_3^2\gamma_2 - \alpha_3^2\beta_2\gamma_1} & \gamma_1 & \dfrac{\alpha_2\alpha_3\beta_1 + \beta_1\beta_3\gamma_2}{\alpha_3\beta_2} \\ \dfrac{\alpha_1\alpha_2\alpha_3^2\beta_2 + \alpha_1\alpha_3\beta_2^2\gamma_2}{\alpha_3^2\beta_2\gamma_1 - \alpha_2\alpha_3\beta_1\beta_3 - \beta_1\beta_3^2\gamma_2} & \dfrac{\beta_3\gamma_2}{\alpha_3} & \gamma_2 \\ 0 & \beta_3 & \alpha_3 \end{vmatrix}. \tag{16}$$

Each point value of point A is 0, representing the initial steady state; thus initial state n_1^0 is substituted into expressions for n_1. It can be understood that since the initial value is set to 0, the system has been in a steady state, and the value is always 0. Since there are nonlinear interactions among the 3 variables, innovation capability, R&D configuration, and knowledge transfer in IURCI systems, these caused a new stable state B in the system. A systematic state change from A to B represents the evolution of the IURCI system. Interactions among the variables and evolution law will determine the outcome of system evolution.

4. Simulation

4.1. Parameter Value. To find existing insufficiency in the Chinese IURCI system, we set specific values of parameters, calculated the coefficient and Jacobian matrix of the model, simulated it using MATLAB, and drew an evolution chart of the 3 variables in the IURCI system. In contrast to a previous capability evaluation in the IURCI system, we measured values of the original control parameters α, β, and γ of innovation capacity, R&D configuration, and knowledge transfer through establishment of an index system. Because innovation capacity represents the presence of IURCI system evolution, we considered relevant content such as innovation effects and innovation gains in the index selection. R&D configuration represents a resource configuration in the IURCI system, so index selection involved funding, personnel, institutional settings, and other aspects. Knowledge transfer represents knowledge flow between enterprises and universities as well as research institutions in IURCI systems, so indicator selection was based on the state of cooperation as the core. We learned from the studies of [18, 26, 31] and other scholars to improve our procedures and ultimately arrived at the indicators shown in Table 1.

In Table 1, each index value was obtained from the *China Statistical Yearbook*, the *China Statistical Yearbook On Science and Technology*, the *China S & T Paper Statistics and Analysis*, the *China Industry Economy Statistical Yearbook*, *Statistics Yearbook On Science And Technology Activities of Industrial Enterprise*, the *Annual Report of Regional Innovation Capability of China*, the *NERI INDEX of Marketization of China's Provinces Report*, and the *SME Technology Innovation Fund*. Related public data from 2005 to 2014 were taken from the statistical yearbooks (2015, 2016 information was not disclosed) to give the final parameter values through data standardization.

According to the solving method [37] of α, β, γ in formula (4), we bring in the data of 2005–2014 from Table 1 and get the parameter values in Table 2.

Table 2 shows the parameter values of the evolution model, and the values of α_1, β_1, γ_1 were calculated from the specific data in the index systems shown in Table 1. In order to obtain the final parameter values of the evolutionary model, we averaged values for 2005–2014 and obtained Table 3.

We substituted values for the 9 parameters into the Jacobian matrix, calculated the determinant and trace of the matrix corresponding to the Jacobian matrix, and analyzed steady-state values of A and B by judging the determinant and trace symbols.

4.2. Tests and Analysis. Using the Jacobian matrix as the coefficient matrix of the equation, the initial state matrix of the differential equation is $[n_1^0, n_2^0, n_3^0]$, wherein n_1^0, n_2^0, n_3^0 represent the initial values of the 3 variables, innovation, R&D configuration, and knowledge transfer, respectively. Point A represents the initial state, and the initial state of the system is 0, such that $N^0 = [0, 0, 0]$; therefore, substituting this into the expression, we obtained the solution for the expression of innovation capacity of 0, that is, $n_1^0, n_2^0, n_3^0 = [0, 0, 0]$, and the output response curve is 0, which indicates the initial state (Figure 1).

In all the figures (Figures 1–8), the x-axis represents the input of three elements, whereas the y-axis represents evolution performance. For different values of the initial

Table 1: Index system of Chinese industry-university-research collaborative innovation system evolution.

Variable	Indicators
Innovation capacity	Growth in the number of patent applications (%)
	Enterprise market share (%)
	Ratio of new products to total enterprise products (%)
	Success rate of innovation projects (%)
	New product sales margins (%)
R&D configuration	Amount of money enterprises invest in product development (ten thousand RMB)
	Enterprises' business loan growth (%)
	Growth of R&D funds of universities and research institutes from corporate (%)
	Total number of R&D institutions of enterprises (set)
	The proportion of investment growth in R&D personnel (%)
Knowledge transfer	Productivity of high quality achievements (%)
	Number of high level articles of partner universities (piece)
	Number of staff entering into enterprises from universities (person)
	Number of patents enterprises purchased every year (set)
	Number of universities in industry-university-research cooperation (set)

Table 2: Parameter value of Chinese industry-university-research collaborative innovation system evolution from 2005 to 2014.

Year	α_1	β_1	γ_1
2005	0.0256	0.7732	0.8422
2006	0.0385	0.7985	0.8974
2007	0.0418	0.8279	0.9361
2008	0.0477	0.8741	0.9713
2009	0.0512	0.9278	1.0284
2010	0.0536	0.9516	1.2947
2011	0.0744	1.0479	1.5820
2012	0.0892	1.2346	1.8046
2013	0.0986	1.8512	1.9239
2014	0.1032	2.2017	2.1158

Table 3: Parameter value of evolution model.

Variable	α	β	γ
n_1	0.0624	1.1489	1.3396
n_2	0.1475	1.5267	1.9464
n_3	0.4587	1.2982	1.3983

conditions, the resulting response curves are different, representing the new steady state. Substituting the relevant data values into the Jacobian matrix, we used MATLAB software to simulate the evolution curves of the 3 variables n_1, n_2, n_3. When the initial values of n_1, n_2, n_3 are different, the output of the system response curve is as shown in Figures 2–8:

Figure 2 illustrates that, at a high setting of innovation in the Chinese IURCI system, collaborative innovation system variables show a rapidly increasing trend in a short period. Among them, the simulation curve of innovation capacity appears as a slowly rising trend after the first drop; we believe the reason for this phenomenon is that, in the real situation of limited resources, the original basis for the innovation capacity of the Chinese IURCI system needs to quickly adjust

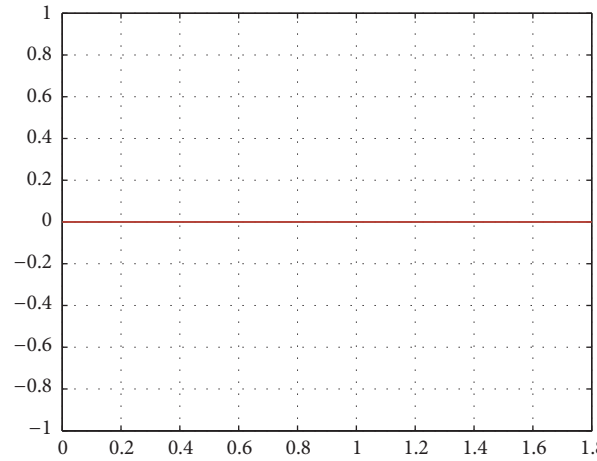


Figure 1: Steady state of industry-university-research collaborative innovation system (0 input).

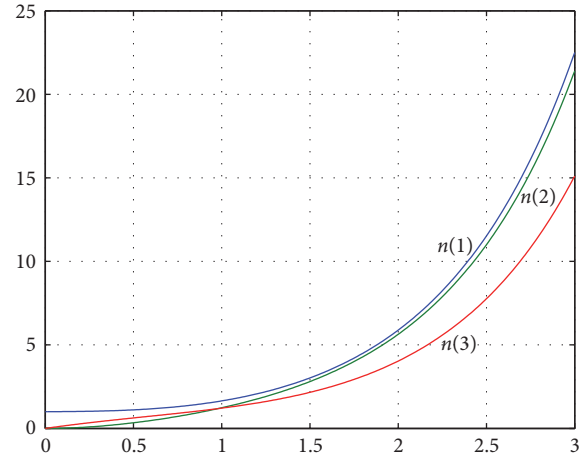


Figure 2: Evolution curve of variables when the initial state set as [1, 0, 0].

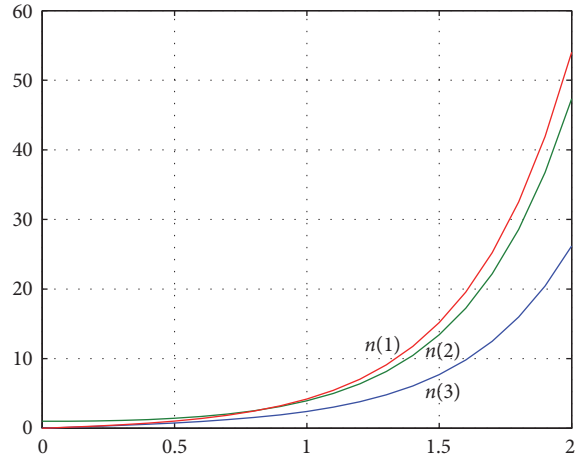


Figure 3: Evolution curve of variables when the initial state set as [0, 1, 0].

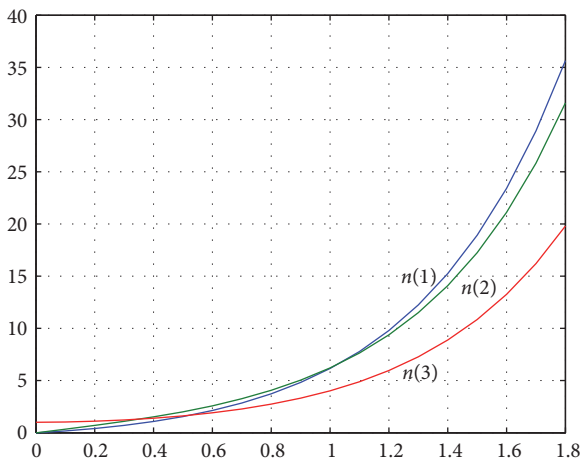


Figure 4: Evolution curve of variables when the initial state set as [0, 0, 1].

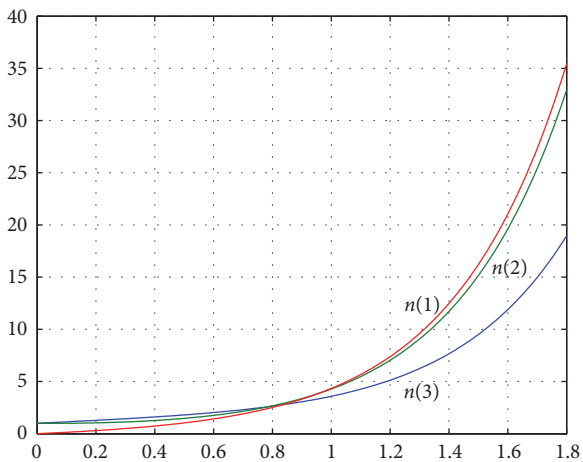


Figure 5: Evolution curve of variables when the initial state set as [0, 1, 1].

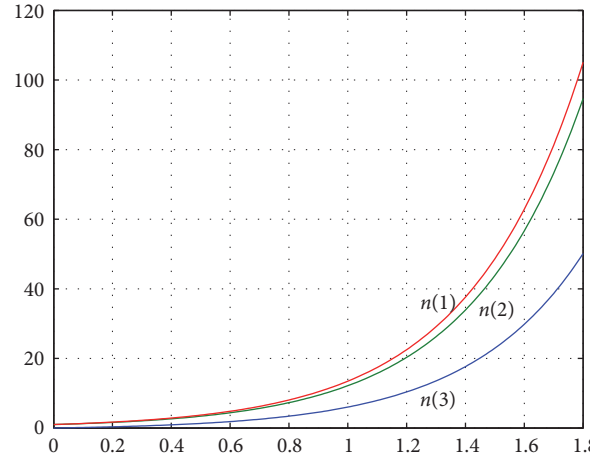


Figure 6: Evolution curve of variables when the initial state set as [1, 1, 0].

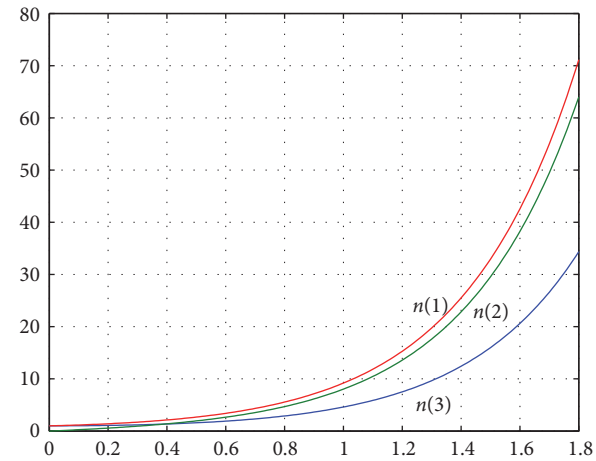


Figure 7: Evolution curve of variables when the initial state set as [1, 0, 1].

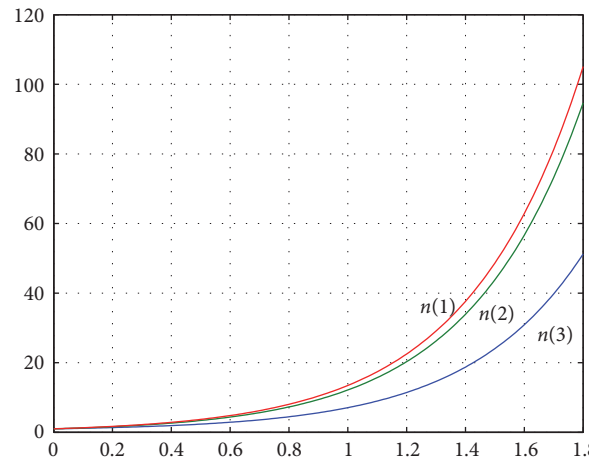


Figure 8: Evolution curve of variables when the initial state set as [1, 1, 1].

to market demand and devote effort towards knowledge transfer activities, resulting in the short-term declining trend of innovation capacity due to output of resources. In the long term, due to the fact that the knowledge base of the IURCI system has been stabilized, a complementary relationship between innovation capacity and R&D configuration is produced, resulting in a faster innovation capability upgrade. This indicates that since the Chinese IURCI system lags behind that of developed countries, the ability to innovate and the evolution of a collaborative innovation system will help to improve targeting of the R&D configuration for businesses, universities, and research institutes and can also enhance promotion of knowledge transfer capability. But, fundamentally, the Chinese IURCI system more likely depends on the innovation capacity for breakthrough innovations and on comprehensively promoting the industrialization of acquired knowledge. In order to foster and enhance the innovative capability, the Chinese IURCI system not only needs to more efficiently allocate limited resources, by means of knowledge transfer, but also should provide enterprises with the high-value heterogeneity knowledge created by universities and research institutions, form a sustained and stable directional flow of knowledge, establish a good feedback path, form complementary advantages, and establish risk sharing, resulting in mutual benefit and a win-win coinnovation situation. Meanwhile, long-term interactions between universities and research institutions and enterprises also contribute to the mutual fit of collaborative innovation between variables; thus knowledge of different attributes can interact to achieve an optimal combination of knowledge resources, achieve knowledge innovation and added value, and ultimately produce the desired collaborative innovation effect, which will enhance the overall competitive advantages of university and industry.

Figure 3 shows that when the systemic R&D configuration is set as 1, that is, if the Chinese IURCI has a more comprehensive and adequate R&D configuration, each variable in the same period has increased significantly, and the evolution speed of innovation capacity is closer to that of the R&D configuration; then the evolution speed of knowledge transfer is relatively slow, being increased only from 0 to about 5.2. Meanwhile, we compared Figures 2 and 3 and found that the evolution curve of knowledge transfer has always been lower than that of innovation capacity and R&D configuration; this phenomenon reveals the fact that, in the evolution process of the Chinese IURCI system, the contribution of knowledge transfer to the evolution of innovative capacity is low, which explains why the influence degree of knowledge transfer on Chinese IURCI system evolution is significantly low. Thus companies can understand that, in the current external environment and policy context, too much dependence on knowledge transfer from universities and research institutions may not achieve the best collaborative innovation strategy. Therefore, under the current circumstances, the Chinese IURCI system mainly considers knowledge transfer as an assistance mode for collaborative innovation and new knowledge access and focuses on enhancing their own innovation capacity and the reasonability of the R&D configuration. Further, as the Chinese IURCI system continues to evolve, universities and research institutions have clearer requirements for the distribution of benefits, and, coupled with a gradual improvement of the intellectual property system, this could lead to a longer cycle of knowledge transfer between enterprises, universities, and research institutions. At this time, collaborative innovation and research will be more inclined to be aided by innovation capability improvement, so that the focus on knowledge transfer is declining. It can be seen that, in the Chinese IURCI system, enhancing the innovation capacity is more important and knowledge transfer has not played its due role, which shows that the Chinese IURCI system has issues around weak knowledge transfer, inadequate trust and cooperation mechanisms between enterprises and universities as well as research institutions, and insufficient knowledge acquisition by enterprises from universities and research institutions.

Figure 4 illustrates, in the context of a lack of innovation capacity and incomplete R&D configuration, that if the Chinese IURCI system was to have more prominent knowledge transfer ability, then knowledge transfer can to some extent contribute to the evolution of collaborative innovation. From Figures 3 and 4, it can be seen that whether we set the R&D configuration as 1 and the knowledge transfer as 0, or the knowledge transfer as 1 and the R&D configuration as 0, as long as these 2 variables have an initial value, they will encourage the innovation capacity in the Chinese IURCI system to improve rapidly. It means that, in this context, innovation capacity is highly correlated with R&D configuration and knowledge transfer in the Chinese IURCI system. Enhancing knowledge transfer capability can help businesses more efficiently and in a more targeted way obtain knowledge that universities and research institutions create and promulgate, which can enhance the speed of knowledge commercialization and help IURCI to achieve high performance, thus strengthening the Chinese IUR union by maintaining internal synergies between knowledge supply and knowledge application. This has a positive impact on promoting collaborative innovation activities and enhancing the confidence of IUR cooperation. At the same time, compared to Figures 2 and 3, it can be seen that, although more prominent knowledge transfer can speed up innovation capacity, the enhancing effectiveness is still lower than that of R&D configuration. The reason may be that knowledge transfer from universities and research institutions is the main way to promote Chinese IURCI system evolution, since the basis of innovation and scientific and technical human capital on the part of enterprises are poor, knowledge input and support of the collaborative innovation system are totally dependent on knowledge transfer from the universities and research institutions side. However, due to the nature of knowledge transfer, there is a certain lag, so, in the absence of appropriate R&D configuration support, the promotion of innovation capability from universities and research institutions is not obvious, so the Chinese IURCI requires a longer cooperation period.

Figure 5 shows that when R&D configuration and knowledge transfer are more prominent in the Chinese IURCI system, the increase in innovation capacity is more obvious than the results of Figures 2, 3, and 4. This demonstrates that R&D configuration and knowledge transfer coaffect

the evolution of innovation capacity in the Chinese IURCI system; that is, there are significant positive feedback effects among the 3 variables, which mutually promote each other. It means that, at such time, as the Chinese IURCI system has some input resources and knowledge transfer capability, these factors will be beneficial to rapidly increasing the collaborative innovation knowledge level and innovation capacity, thus improving collaborative innovation performance. We believe that the reasons for this may be twofold: on the one hand, more effective knowledge transfer between corporations and universities as well as research institutions promotes knowledge industrialization, so the innovation capability of enterprises quickly upgrades in the process of repeated practice and constant learning, which to a certain extent promotes IURCI performance. On the other hand, rational and targeted R&D configuration allows universities and research institutions, under the premise of having a stable knowledge base, to reduce the time delay of knowledge transfer in the process of collaborative innovation and put part of their resources into the process of promotion of knowledge transfer capacity, thereby achieving a mutual fit with R&D configuration. It should be noted that, in the initial stage of simulation, namely, the early period of Chinese IURCI practice, favorable Chinese government policies and attention to IURCI could promote the speed of knowledge industrialization. But knowledge transfer from universities and research institutions is subject to adverse factors in the cooperation run-in period and may produce a situation of no improvement. With the in-depth evolution of the IURCI system, cooperation between businesses and universities and research institutions can mature, so that universities and research institutions have clearer ideas about industry needs and more accurately provide the necessary knowledge to enterprises for knowledge industrialization. Universities and research institutions will also benefit from collaborative innovation as they better understand industry technology needs and market demand, allowing them to optimize their knowledge structure, thus enhancing the IURCI performance.

Figure 6 illustrates that, in the evolution process of IURCI that relies on innovation capacity and R&D configuration, continuously increasing R&D configuration will make evolution of innovation capacity significantly faster. At the same time, compared to Figure 5, it can be found that, in the evolution of the Chinese IURCI system, if R&D configuration has an initial value, to enhance the IURCI performance, enterprises must operate on the basis of continuously developing their R&D configuration, obtaining as much as possible valuable heterogeneity knowledge from universities and research institutions, and universities and research institutions also need to enhance their abilities to transfer knowledge and fit the production needs of industry in order to obtain more significant innovation performance. In addition, this case also shows that although there may be differences in their abilities to innovate, for those enterprises that lack the ability to innovate, the input of external resources and the gain of valuable knowledge by way of knowledge transfer may change the current disadvantage and help them achieve rapid growth.

Figures 5 and 6 both show that the innovation capacity and R&D configuration of the Chinese IURCI system can in a short period of time produce significant synergies and promote each other, indicating that China's current IURCI development still requires a lot of resources investment as the primary means to enhance innovation capability. But when R&D configuration is high, the result of innovation capacity evolution [38] is lower than that when knowledge transfer is relatively high, which gives high values of innovation evolution [42]. On the one hand, this indicates that, in the current process of IURCI in China, the universities and research institutions side creates and sends high-value heterogeneity knowledge, which has a fundamental role in promoting innovation capacity. On the other hand, to a certain extent, this confirms that if Chinese universities and research centers could face knowledge-oriented industrial demand, enterprises could accurately identify the market value of accepted knowledge, change the discrepancy that exists between technology supply and technology demand, and thereby reduce its dependency on R&D configuration, while, at the same time, shortening the innovation cycle and avoiding the issues such as information asymmetry and transaction costs that may arise during collaborative innovation [42–44].

Figure 7 illustrates that when innovation capacity and knowledge transfer are more prominent and R&D configuration is lower, variables of the Chinese IURCI could promote each other by interaction, but the growth rate is relatively small, indicating that relying solely on knowledge transfer to promote innovation capacity is not effective. On the one hand, this confirmed that the type of knowledge produced by Chinese universities and research institutions must have applicability to be promoted to companies for industrialization. On the other hand, this maybe results from enterprises having an inadequate knowledge absorptive capacity. Meanwhile, from Figures 7, 5, and 6, it can be found that if the R&D configuration of the Chinese IURCI system is poor, then there will be a very significant impact on the ability to enhance knowledge transfer speed. The possible reasons are that when R&D configuration is not enough, the Chinese IURCI system is already in the process of tight resource allocation, and it is bound to devote more resources into promotion activities that create a high-value heterogeneity innovation capacity, and it therefore lacks the resources to facilitate knowledge transfer within the system. Correspondingly, if resources are relatively abundant, enterprises must allow for a bottleneck phenomenon in enhancing innovation capacity or accept greater risks in innovative activity and may assign the resources originally devoted to innovative ability to universities and research institutions, supporting the knowledge transfer willingness of universities and research institutions and thus enhancing knowledge transfer for collaborative innovation. In this case, the evolution curve of knowledge transfer will be more significant.

Figure 8 clearly shows that when the innovation capacity of the Chinese IURCI system is more prominent, R&D configuration is more targeted, knowledge transfer capability is better, and the 3 state variables of collaborative innovation

show rapid and steady evolution. Among them, the growth rate of innovation capability is the most significant, so that after a new stable state is established, knowledge transfer and R&D configuration both can have a more substantial growth effect on innovation capability. This means that, in order to achieve the objective of significantly improved collaborative innovation performance, the Chinese IURCI system must adhere to long-term independent innovation activities and in addition must maintain and increase the basic R&D configuration, constantly improve knowledge transfer capacity, and more effectively and pertinently send to enterprises the complementary knowledge that universities and research institutions own. With advances in IURCI, this approach may bring 2 advantages for enterprises, universities, and research institutions involved in Chinese IURCI, but, from a business perspective, at present, the awareness of Chinese enterprises to acquire complementary knowledge in an IUR-coordinated manner is poor, and so initiative is very low. If enterprises realize that, on the basis of improving R&D configuration, they can obtain complementary research achievement by making use of knowledge transfer form; then this will greatly increase the willingness of Chinese enterprises to participate in IURCI, thereby using new technology, developing new products, and approaching university R&D personnel, which is beneficial to building a trust-based cooperation mechanism between enterprises and universities as well as research institutions [45–48]. From the perspective of Chinese universities and research institutes in the process of knowledge creation, the biggest drawback is that new knowledge cannot be applied to production practice; that is, universities and research institutions pay too much attention to academic value. With periodic IURCI, universities and research institutions can not only receive financial support from the business, according to the needs of the business to carry out research activities but also fully explore new research areas, which will also play a positive role in promoting more academic achievements [49, 50].

From observing Figures 3–8 we found that, in the current Chinese IURCI process, R&D configuration always maintains a high degree of positive correlation with innovation capacity. The reasons for this may be that China's current IURCI system is not perfect, and enterprises and universities as well as research institutions are always looking for cooperation modes suitable for both sides. Meanwhile, the process of knowledge transfer itself is relatively slow, and, between Chinese enterprises and universities, there might exist negative factors affecting knowledge transfer speed (such as knowledge flow cost and familiarity running process between staffs), and these factors will result in reduced efficiency of knowledge transfer; therefore growth of knowledge transfer is always slow.

In short, in China's IURCI process, a consistent relationship indeed exists among innovation capability, knowledge transfer, and R&D configuration. In a higher knowledge transfer and comprehensive R&D configuration scenario, enterprises can propose more targeted knowledge needs and initially provide financial and material support for universities and research institutions involved in innovation. Universities and research institutions from the strategic level are also concerned about how to establish knowledge R&D to serve enterprises and, under the premise of integrated resources, establish a balanced benefits distribution, actively carry out scientific and technological achievement transfer, train technology and management personnel required for enterprises, exert collaborative innovation effects by complementary advantages, and thus bring new benefits for both sides.

5. Conclusion

By combining methods of game theory and complex science, we studied the evolution of the Chinese IURCI system and interactions among the variables. We used interactions among variables in a collaborative innovation system to "map" the perspective of collaborative innovation evolution and study innovation capacity, R&D configuration, and knowledge transfer as key elements of collaborative innovation, and, on the basis of establishing a dynamic system evolution model of collaborative innovation, we collected data, simulated, discussed interactions among the variables in the process of Chinese IURCI, and analyzed principles of collaborative innovation. We mainly obtained the following conclusions:

(1) In the evolution of the Chinese IURCI system, innovation capacity and knowledge transfer are dependent on R&D configuration, and innovation capability is highly positively correlated with R&D configuration. The higher the target and investment level of R&D configuration, the faster the evolution of innovation capacity and knowledge transfer. It is noted that currently Chinese IURCI development is using a lot of resource investment as its main factor. Innovation capacity has become the key variable to promote collaborative innovation system evolution.

(2) Strong knowledge transfer capability can accelerate the evolution speed of innovation capacity, but overall knowledge transfer remains at a relatively stable evolution state and does not change with evolution of collaborative innovation. This explains that, in the evolution process of China's current IURCI system, virtuous cooperation mechanisms based on trust between enterprises and universities as well as research institutions are lacking. Universities and research institutions also pay more attention to the academic value of knowledge, leading to insufficient contribution of knowledge transfer to knowledge industrialization by enterprises and reduced collaborative innovation system evolution.

(3) In a relatively balanced state of innovation capacity, R&D configuration, and knowledge transfer, the faster the evolution speed of Chinese IURCI systems, the higher the stability. This shows that the Chinese IURCI system must adhere to long-term independent innovation, and, on the basis of maintaining and improving R&D configuration, constantly improve internal knowledge transfer capability.

Compared with early research, this article has 3 main findings: first, we chose China as a developing country with rapid economic growth, used a dynamic perspective, analyzed the evolution mechanism of the Chinese IURCI system, and demonstrated the interactions among variables of the collaborative innovation system. Secondly, we learned from

the macroinnovation system theory, the meso-Triple Helix Model, and microopen organization innovation theory, separated the components of the IURCI system, and confirmed contribution degrees of different variables to collaborative innovation system evolution, with particular emphasis on different initial values. Our rationale of analyzing 3 variables with different evolutionary trends not only theoretically contributed to enrich the innovative theory system but also made developmental strategies for Chinese enterprises and universities as well as research institutions and provided an optimized direction to enhance collaborative innovation performance. Thirdly, in this study, we considered IURCI as a complex system with dynamic evolution and, by analysis of systemic evolution and cooperative status among variables, we effectively revealed some issues existing in the current Chinese IURCI.

This study made 2 contributions to studies of IURCI based on resources and knowledge management theory: first, this study overcame the disadvantage of only using transaction cost theory and organization management theory to analyze comprehensiveness and complexity of collaborative innovation. The research framework and methodology proposed in this study will help researchers deal with challenges so that, in a dynamic environment, the synergistic effect of different variables of the IURCI system results in complexity of collaborative innovative evolution. Secondly, the method chosen for this study is different from past collaborative innovation research processes that were narrower and divided by limited dimensions. We started from the variables that determine the IURCI system, established a nonlinear model based on the interactions among variables, confirmed the impact level of different state variables on collaborative innovation, and defined the mechanism of interactions between state variables and evolution trend. The use of this new method to obtain new conclusions is necessary for further research on collaborative innovation.

This study is also of great practical implication. Based on 10 years of data published by Chinese government, interactions and evolution trend among innovation capacity, R&D configuration and knowledge transfer in different contexts are given by means of simulation, thus specifying existing problems in China's IURCI system. Conclusions obtained are beneficial in helping enterprises managers take measures to promote knowledge absorption, in helping universities and research institutions clarify knowledge transfer dilemma in certain knowledge demand condition, and helping government form proper resource allocation mode. Therefore, the new conceptualization provides a useful guide for managers to make rational knowledge developing strategy, for universities and research institutes to diagnose barriers in knowledge transfer, and for government to take necessary policy-making.

Selecting data from different countries to conduct comparisons and discuss classification is the main research direction in the future. Because the framework proposed in this study can be used in many fields, this study plays a guiding role to some extent in practical sample collection and analysis when trying to select future strategic alliances, practice communities, and other subjects. Meanwhile, we hoped to bring the phenomenon of knowledge diffusion into research and, by improving existing models, achieve a complete interpretation of IURCI issues on aspects of theory and practice, thus making research more focused.

Conflicts of Interest

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

Acknowledgments

This study is supported by the National Natural Science Foundation of China (71602041, 71602042 and 71301065); National Social Science Key Project of China (ID: 14AGL004); China Postdoctoral Science Foundation (2015M570299 and 2016M590605); Fundamental Research Funds for the Central Universities of China (ID: HEUCF150901); Postdoctoral Science Foundation of Heilongjiang Province (LBH-15075); and Postdoctoral Science Foundation of Jiangxi Province (2016KY27).

References

[1] O. W. Maietta, "Determinants of university-firm R&D collaboration and its impact on innovation: a perspective from a low-tech industry," *Research Policy*, vol. 44, no. 7, pp. 1341–1359, 2015.

[2] M. D. Santoro and P. E. Bierly III, "Facilitators of knowledge transfer in university-industry collaborations: a knowledge-based perspective," *IEEE Transactions on Engineering Management*, vol. 53, no. 4, pp. 495–507, 2006.

[3] O. Al-Tabbaa and S. Ankrah, "Social capital to facilitate "engineered" university-industry collaboration for technology transfer: a dynamic perspective," *Technological Forecasting and Social Change*, vol. 104, pp. 1–15, 2016.

[4] A. I. Canhoto, S. Quinton, P. Jackson, and S. Dibb, "The co-production of value in digital, university-industry R&D collaborative projects," *Industrial Marketing Management*, vol. 56, pp. 86–96, 2016.

[5] S. Arvanitis, U. Kubli, and M. Woerter, "University-industry knowledge and technology transfer in Switzerland: what university scientists think about co-operation with private enterprises," *Research Policy*, vol. 37, no. 10, pp. 1865–1883, 2008.

[6] M. Perkmann and K. Walsh, "University-industry relationships and open innovation: towards a research agenda," *International Journal of Management Reviews*, vol. 9, no. 4, pp. 259–280, 2007.

[7] K. Koschatzky, "Networking and knowledge transfer between research and industry in transition countries: empirical evidence from the Slovenian innovation system," *The Journal of Technology Transfer*, vol. 27, no. 1, pp. 27–38, 2002.

[8] A. Inzelt, "The evolution of university-industry-government relationships during transition," *Research Policy*, vol. 33, no. 6-7, pp. 975–995, 2004.

[9] N. Yumusak, I. Ozcelik, M. Iskefiyeli, M. F. Adak, and T. Kırktepeli, "University industry linkage projects management system," *Procedia—Social and Behavioral Sciences*, vol. 174, pp. 3254–3259, 2015.

[10] J. Zhao, X. Xi, and S. Yi, "Resource allocation under a strategic alliance: how a cooperative network with knowledge flow spurs

co-evolution," *Knowledge-Based Systems*, vol. 89, pp. 497–508, 2015.

[11] J. Berbegal-Mirabent, J. L. Sánchez García, and D. E. Ribeiro-Soriano, "University-industry partnerships for the provision of R&D services," *Journal of Business Research*, vol. 68, no. 7, pp. 1407–1413, 2015.

[12] R. Fontana, A. Geuna, and M. Matt, "Factors affecting university-industry R and D projects: the importance of searching, screening and signalling," *Research Policy*, vol. 35, no. 2, pp. 309–323, 2006.

[13] S.-H. Chen, M.-H. Huang, and D.-Z. Chen, "Driving factors of external funding and funding effects on academic innovation performance in university-industry-government linkages," *Scientometrics*, vol. 94, no. 3, pp. 1077–1098, 2013.

[14] B. A. Lundvall, *National Innovation System: Toward a Theory of Innovation and Interactive Learning*, Pinter, London, UK, 1992.

[15] P. Cooke, M. G. Uranga, and G. Etxebarria, "Regional systems of innovation: an evolutionary perspective," *Environment and Planning A*, vol. 30, no. 9, pp. 1563–1584, 1998.

[16] A. K. W. Lau and W. Lo, "Regional innovation system, absorptive capacity and innovation performance: an empirical study," *Technological Forecasting and Social Change*, vol. 92, pp. 99–114, 2015.

[17] I. A. Ivanova and L. Leydesdorff, "Rotational symmetry and the transformation of innovation systems in a Triple Helix of university-industry-government relations," *Technological Forecasting and Social Change*, vol. 86, pp. 143–156, 2014.

[18] I. A. Ivanova and L. Leydesdorff, "Knowledge-generating efficiency in innovation systems: the acceleration of technological paradigm changes with increasing complexity," *Technological Forecasting and Social Change*, vol. 96, pp. 254–265, 2015.

[19] L. Leydesdorff and M. Meyer, "Triple Helix indicators of knowledge-based innovation systems: introduction to the special issue," *Research Policy*, vol. 35, no. 10, pp. 1441–1449, 2006.

[20] H. Etzkowitz and L. Leydesdorff, "The triple helix University-industry government relations: a laboratory for knowledge based economic development," *EASSR Review*, vol. 14, no. 1, pp. 11–19, 1995.

[21] H. W. Chesbrough, *Open Innovation: The New Imperative for Creating and Profiting from Technology*, Harvard Business Press, 2003.

[22] J. West and K. R. Lakhani, "Getting clear about communities in open innovation," *Industry and Innovation*, vol. 15, no. 2, pp. 223–231, 2008.

[23] A. Draghici, C. Baban, M. Gogan, and L. Ivascu, "A knowledge management approach for the university-industry collaboration in open innovation," *Procedia Economics and Finance*, vol. 23, pp. 23–32, 2015.

[24] L. Striukova and T. Rayna, "University-industry knowledge exchange: an exploratory study of Open Innovation in UK universities," *European Journal of Innovation Management*, vol. 18, no. 4, pp. 471–492, 2015.

[25] G. George, S. A. Zahra, and D. R. Wood Jr., "The effects of business-university alliances on innovative output and financial performance: a study of publicly traded biotechnology companies," *Journal of Business Venturing*, vol. 17, no. 6, pp. 577–609, 2002.

[26] K. Laursen and A. Salter, "Open for innovation: the role of openness in explaining innovation performance among U.K. manufacturing firms," *Strategic Management Journal*, vol. 27, no. 2, pp. 131–150, 2006.

[27] A. Winkelbach and A. Walter, "Complex technological knowledge and value creation in science-to-industry technology transfer projects: the moderating effect of absorptive capacity," *Industrial Marketing Management*, vol. 47, pp. 98–108, 2015.

[28] Y.-J. Chen, Y.-M. Chen, and M.-S. Wu, "An empirical knowledge management framework for professional virtual community in knowledge-intensive service industries," *Expert Systems with Applications*, vol. 39, no. 18, pp. 13135–13147, 2012.

[29] M. C. J. Caniëls and B. Verspagen, "Barriers to knowledge spillovers and regional convergence in an evolutionary model," *Journal of Evolutionary Economics*, vol. 11, no. 3, pp. 307–329, 2001.

[30] M.-C. Hu, "Knowledge flows and innovation capability: the patenting trajectory of Taiwan's thin film transistor-liquid crystal display industry," *Technological Forecasting and Social Change*, vol. 75, no. 9, pp. 1423–1438, 2008.

[31] J. Bercovitz and M. Feldmann, "Entpreprenerial universities and technology transfer: a conceptual framework for understanding knowledge-based economic development," *Journal of Technology Transfer*, vol. 31, no. 1, pp. 175–188, 2006.

[32] A. Bonaccorsi and A. Piccaluga, "A theoretical framework for the evaluation of university-industry relationships," *R&D Management*, vol. 24, no. 3, pp. 229–247, 1994.

[33] E. G. Carayannis, J. Alexander, and A. Ioannidis, "Leveraging knowledge, learning, and innovation in forming strategic government-university-industry (GUI) R&D partnerships in the US, Germany, and France," *Technovation*, vol. 20, no. 9, pp. 477–488, 2000.

[34] L. Anatan, "Conceptual issues in university to industry knowledge transfer studies: a literature review," *Procedia—Social and Behavioral Sciences*, vol. 211, pp. 711–717, 2015.

[35] H. W. Chesbrough, W. Vanhaverbeke, and J. West, *Open Innovation: Researching a New Paradigm*, Oxford University Press, Oxford, UK, 2008.

[36] Y. Baba, N. Shichijo, and S. R. Sedita, "How do collaborations with universities affect firms' innovative performance? The role of 'Pasteur scientists' in the advanced materials field," *Research Policy*, vol. 38, no. 5, pp. 756–764, 2009.

[37] H. Haken, *The Science of Structure: Synergetics*, Van Nostrand Reinhold, Grantham, UK, 1981.

[38] J. L. Kuang, P. A. Meehan, and A. Y. Leung, "Suppressing chaos via Lyapunov-Krasovskii's method," *Chaos, Solitons & Fractals*, vol. 27, no. 5, pp. 1408–1414, 2006.

[39] G. A. Anastassiou, "Distributional Taylor formula," *Nonlinear Analysis. Theory, Methods & Applications*, vol. 70, no. 9, pp. 3195–3202, 2009.

[40] Y. Khan, H. Vázquez-Leal, and N. Faraz, "An auxiliary parameter method using Adomian polynomials and Laplace transformation for nonlinear differential equations," *Applied Mathematical Modelling*, vol. 37, no. 5, pp. 2702–2708, 2013.

[41] S. F. AL-Azzawi, "Stability and bifurcation of pan chaotic system by using Routh-Hurwitz and Gardan methods," *Applied Mathematics and Computation*, vol. 219, no. 3, pp. 1144–1152, 2012.

[42] S. Owen and A. Yawson, "Information asymmetry and international strategic alliances," *Journal of Banking and Finance*, vol. 37, no. 10, pp. 3890–3903, 2013.

[43] R. A. Jensen, J. G. Thursby, and M. C. Thursby, "Disclosure and licensing of University inventions," *International Journal of Industrial Organization*, vol. 21, no. 9, pp. 1271–1300, 2003.

[44] X. Gao, X. Guo, and J. Guan, "An analysis of the patenting activities and collaboration among industry-university-research institutes in the Chinese ICT sector," *Scientometrics*, vol. 98, no. 1, pp. 247–263, 2014.

[45] Y. S. Lee, ""Technology transfer" and the research university: a search for the boundaries of university-industry collaboration," *Research Policy*, vol. 25, no. 6, pp. 843–863, 1996.

[46] M. Hemmert, L. Bstieler, and H. Okamuro, "Bridging the cultural divide: trust formation in university-industry research collaborations in the US, Japan, and South Korea," *Technovation*, vol. 34, no. 10, pp. 605–616, 2014.

[47] M. D. Santoro and P. A. Saparito, "The firm's trust in its university partner as a key mediator in advancing knowledge and new technologies," *IEEE Transactions on Engineering Management*, vol. 50, no. 3, pp. 362–373, 2003.

[48] K. Blomqvist, P. Hurmelinna, and R. Seppänen, "Playing the collaboration game right—balancing trust and contracting," *Technovation*, vol. 25, no. 5, pp. 497–504, 2005.

[49] A. Geuna and L. J. J. Nesta, "University patenting and its effects on academic research: the emerging European evidence," *Research Policy*, vol. 35, no. 6, pp. 790–807, 2006.

[50] Y. S. Lee, "The sustainability of university-industry research collaboration: an empirical assessment," *Journal of Technology Transfer*, vol. 25, no. 2, pp. 111–133, 2000.

Hindawi
Complexity
Volume 2017, Article ID 3203615, 11 pages
https://doi.org/10.1155/2017/3203615

Research Article

Optimization of the Critical Diameter and Average Path Length of Social Networks

Haifeng Du,[1] Xiaochen He,[1] Wei Du,[1] and Marcus W. Feldman[1,2]

[1]*Center for Administration and Complexity Science, Xi'an Jiaotong University, Xi'an, Shanxi Province 710049, China*
[2]*Morrison Institute for Population and Resource Studies, Stanford University, Stanford, CA 94305, USA*

Correspondence should be addressed to Marcus W. Feldman; mfeldman@stanford.edu

Received 29 November 2016; Accepted 13 February 2017; Published 28 March 2017

Academic Editor: Katarzyna Musial

Optimizing average path length (APL) by adding shortcut edges has been widely discussed in connection with social networks, but the relationship between network diameter and APL is generally ignored in the dynamic optimization of APL. In this paper, we analyze this relationship and transform the problem of optimizing APL into the problem of decreasing diameter to 2. We propose a mathematic model based on a memetic algorithm. Experimental results show that our algorithm can efficiently solve this problem as well as optimize APL.

1. Introduction

Following the introduction of models for small-world and scale-free networks, much research has been devoted to analyzing network characteristics [1–5]. In particular, there has been a focus on finding indices to quantify features of network structure such as structural entropy, robustness, or modularity [6–8]. These indices play an important role in measuring specific performance aspects of networks, and optimizing them can help to improve network performance.

Average path length (APL), the average shortest distance between all nodes in a network, is not only a measurement of static characteristics such as connectivity and robustness but also an important control variable in dynamic processes, such as the spread of diseases or target searching [9–11]. Optimizing APL has also attracted attention in the field of structural optimization. Decreasing APL by adjusting nodes or edges can effectively enhance the transfer efficiency and synchronization ability [12–17]. In addition, optimization of APL has also been widely used in urban planning and site selection [14, 18, 19]. Xuan et al. [20] proposed a simulated annealing model to optimize APL in order to speed up convergence. Keren [21] employed a spectral technique to reduce APL in binary decision diagrams.

In order to optimize APL, many scholars focus on adding a given number of edges to produce the largest decrease in APL. These added edges are called "shortcut" edges and the problem of finding the best set of shortcut edges is defined as the "shortcut-selection" problem [22]. A series of methods have been proposed to solve this problem. Meyerson and Tagiku proposed an approximation method, which involved finding a source node and then connecting k other nodes to this node to decrease APL [22]. Parotsidis et al. analyzed the exact effect of a single edge insertion on APL and proposed the EdgeEffect Algorithm to maximize the effect of edge insertion [23]. A greedy algorithm, which adds edges one by one and which makes the maximum reduction of APL for each added edge, has proved to be efficient [24, 25]. These methods have solved the shortcut-selection problem to some extent. However, a common phenomenon has been ignored in the process of adding edges. In experiments to optimize APL by adding edges, we find that no matter which method is used to add edges, there always exists a turning point at which APL begins to decrease linearly as more edges are added. This phenomenon can be related to the network diameter.

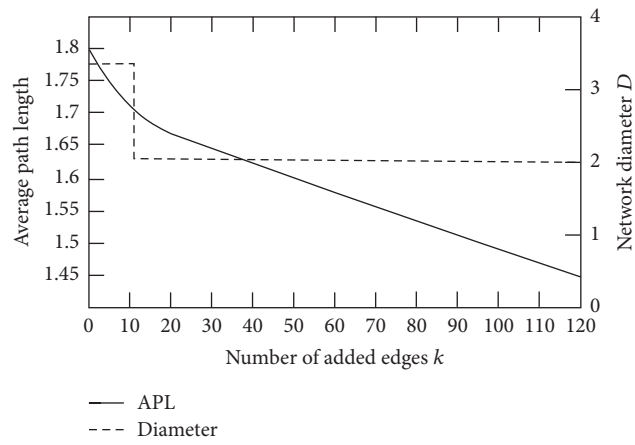


Figure 1: The value of APL and network diameter D as the number of added edges increases.

In this paper, we define the network diameter at the turning point as the "critical diameter," and analyze both this critical diameter and APL in the process of adding edges. We transform the problem of optimizing APL into the problem of optimizing the critical diameter. Specifically, we focus on adding the minimum number of shortcut edges to make the network diameter decrease to 2. Research on predicting missing links has attracted much attention in recent years, the algorithms of which can extract missing information or identify spurious interactions [26–29]. Gao et al. analyzed the feature of predicted network, and they found the network diameter and APL shows a negative linear relation to all of the tested prediction methods [29]. Therefore, our research can also provide some a priori knowledge in designing the method of link prediction. In the next section, we introduce the critical diameter and explore the special relationship between critical diameter and APL; the algorithm for optimizing the critical diameter is proposed in Section 3; Section 4 gives results of testing our method on generated networks; our conclusions and further work are presented in Section 5.

2. Critical Diameter and APL

Network diameter, the maximum path length for all pairs of nodes, is closely related to APL; they both contain information about connectivity and transfer efficiency [30, 31]. Imase and Itoh gave the inequalities $D/2 \leq \text{APL} \leq D$ to describe the static relationship between network diameter, D, and APL [32]. In the dynamic process of adding shortcut edges, there exists a turning point, as shown in Figure 1. APL declines nonlinearly with the number of added edges k until a turning point and then decreases linearly as k increases further. We compute the path length between every pair of nodes and find that the longest path length of the network is larger than 2 before k reaches the turning point (i.e., the network diameter $D > 2$); when k reaches the turning point, the diameter equals 2 ($D = 2$). This is because if $D = 2$, a new added edge between a pair of nodes can only change the path lengths between these two nodes from 2 to 1 but cannot change the path lengths of other pairs of nodes, and the APL can be reduced by just $2/n(n-1)$ for each added edge, which constitutes a linear decline.

In fact, the APL can be computed when the network diameter declines to 2. For a network $G = (V, E)$ with n nodes and m edges, when the diameter equals 2, the path length of every pair of nodes will be equal to or less than 2. If we add k edges to the network, the number of pairs of nodes whose path length equals 1 will be $m + k$ and the number of pairs of nodes whose path length equals 2 will be $n(n-1)/2 - m - k$. Therefore, the value of APL achieved by adding k edges with the network diameter $D = 2$ is

$$\begin{aligned} \text{APL} &= \frac{(m+k) + (n(n-1)/2 - m - k) \times 2}{n(n-1)/2} \\ &= 2 - \frac{2(m+k)}{n(n-1)}. \end{aligned} \tag{1}$$

In the process of adding edges, APL will ultimately decrease linearly and will become equal to the term on the right of (1). Figure 2 shows the results of 20 simulations adding edges randomly to decrease APL; the maximum, mean, and minimum APL of these 20 runs are shown. The three curves become overlapping and linear when the number of added edges becomes large enough. The curve of the minimum APL becomes linear earliest, while the curve of maximum APL is the last to become linear.

In this case, if we attempt to minimize APL by adding a large number of shortcut edges, we can find a solution for which adding a small number of edges has decreased the diameter to 2 and add the remaining edges randomly. Therefore, the problem of optimizing APL can be transformed into the problem of finding shortcut edges that quickly decrease the diameter to 2.

Here, we propose a formal definition for the diameter when APL begins to decline linearly.

Definition 1. In adding edges to a network, the network diameter declines to 2. The network diameter in this case is defined as the "critical diameter," denoted as D_c.

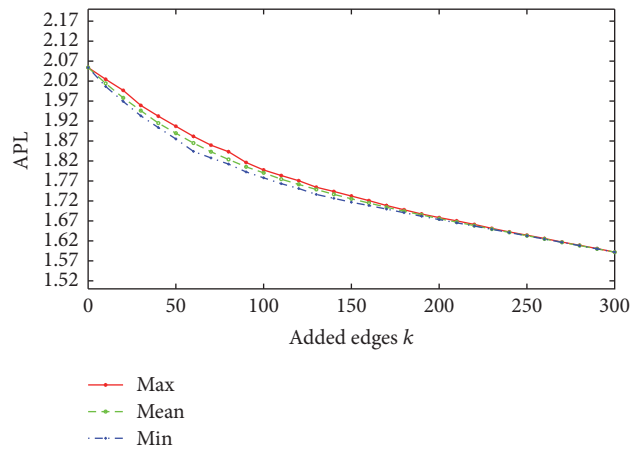
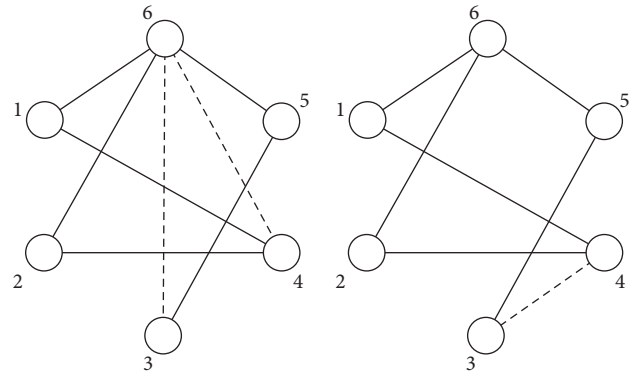


Figure 2: The decrease in APL caused by randomly adding edges.



(a) HDN (b) Other edging method

Figure 3: Different methods to optimize network diameter. The solid lines represent edges of the initial network, while the dashed lines represent added edges.

If we add k shortcut edges to make the network diameter become D_c, the set of these k shortcut edges must be the most optimal solution for minimizing APL by adding k edges. If there exists a solution which can make APL lower by adding another set of k edges, the number of pairs of nodes whose path length equals 1 must be bigger than $m + k$, which cannot be realized by adding only k edges. This kind of relationship between APL and D_c suggests that if we can minimize the number of added edges k to reduce the diameter to D_c, the APL can efficiently decrease to its lowest level.

In this paper, we focus on how to decrease the diameter to 2 by adding the minimum number of edges; this problem is defined as "optimizing the critical diameter." The objective function can be formulated as

$$\begin{aligned} \min \quad & k \\ \text{s.t.} \quad & \forall i \neq j, \quad d_{ij} \leq 2, \end{aligned} \tag{2}$$

where k represents the number of added edges and d_{ij} represents the path length between node i and node j.

It should be noted that the problem of optimizing the critical diameter is an NP-hard problem. Given a connected network $G = (V, E)$ with n nodes and m edges, there can be $\binom{n(n-1)/2-m}{k}$ ways of adding k shortcut edges. Since computing the network diameter costs at least $O(n^3)$, finding the best set of shortcut edges requires $O(n^{2k+3})$, which is high even for a small network.

An efficient way to optimize D_c is to establish connections between the highest-degree node and the rest of nodes, which adds $n - 1 - k_{\max}$ edges (n is the number of nodes and $k_{\max}$ is the highest degree of the network). Then the network will definitely become a Star Network with the highest-degree node at the center. We call this method "HDN" (connecting to the highest-degree node).

However, HDN fails to generate the globally optimal solution. As shown in Figure 3, node 6 is the highest-degree node of the network. To decrease the diameter to 2 by HDN, we should add two edges connecting nodes 3 and 6 and nodes 4 and 6; but if we establish a connection between nodes 3 and 4, the network diameter can also become D_c. Thus we need to design a more efficient method to decrease the network diameter to 2.

(1) Input: the maximum iteration number: $I_{\max}$; population size: S_{pop}; mating pool size: S_{pool}; tournament size: S_{tour}; crossover probability: P_c; mutation probability: P_m; the initial network adjacency matrix: A.
(2) $P \leftarrow$ Initial_Population(S_{pop});
(3) Repeat
(4) $\quad P_{\text{parent}} \leftarrow$ Tournament_Selection($P, S_{\text{pool}}, S_{\text{tour}}$);
(5) $\quad P_{\text{offspring}} \leftarrow$ Genetic_Operation($P_{\text{parent}}, P_c, P_m$);
(6) $\quad P'_{\text{offspring}} \leftarrow$ Local_search($P_{\text{offspring}}$);
(7) $\quad P \leftarrow$ Update_Population($P, P'_{\text{offspring}}$);
(8) Until Termination($I_{\max}$)
(9) Output: the number of added edges, the position of added edges.

Algorithm 1: Framework of our algorithm.

3. The Algorithm for Optimizing Critical Diameter

Memetic algorithms combined with techniques of long-distance and short-distance search have proved to be effective in solving NP-hard problems [33, 34]. In this section, we introduce a memetic algorithm that combines a genetic algorithm and a heuristic local search to optimize critical diameter. We call the method "MA-CD."

3.1. Framework. The framework of MA-CD is shown in Algorithm 1. We first input some necessary parameters such as the maximum iteration number and the population size as well as the adjacency matrix of the network. We generate a population P by the function Initial_Population( ). Next, we repeat the process for optimizing D_c until the number of iterates is $I_{\max}$, or the objective function remains unchanged for 50 iterations. In repeating this process, we first use Tournament_Selection( ) to select the parent population for genetic operations; then we apply two-point crossover and one-point mutation to generate offspring chromosomes by Genetic_Operation( ); we apply some a priori knowledge to carry out a local search on the offspring chromosomes by Local_Search( ); Update_Population( ) is used to construct a new population with better performing chromosomes. Finally, we output the results.

3.2. Representation and Initialization. We aim to find those positions at which we should add edges to optimize D_c. To this end, we find all the positions of these nonexistent edges and encode them as genes $x_i \subseteq X$ in the chromosome $X \in \{0, 1\}$. $x_i = 1$ represents adding a new edge to the corresponding position, while $x_i = 0$ represents not adding an edge. Figure 4 shows an illustration of the representation. We identify the nonexistent edges between nodes 1–3, 1–4, 2–4, 2–5, and 3–5. For the initial network, all the genes are assigned 0 because there are no added edges. If we assign 1 to the first and second gene as shown in Chromosome 2, then the edges between nodes 1–3 and 1–4 will be added. Similarly, when we assign 1 to the first and fourth gene, the edges between 1–3 and 2–5 will be added.

In the initialization, we generate a population of chromosomes and randomly assign 0 or 1 to every gene in the chromosomes.

3.3. The Genetic Operation. The genetic operation consists of two-point crossover and one-point mutation. The crossover operation is described in the appendix. Given two parent chromosomes X_{parent1} and X_{parent2}, we randomly choose two points, and then the parent chromosomes are divided into three parts by two chosen points. Next, we randomly select one part, and all the genes in this part are swapped between X_{parent1} and X_{parent2} with probability P_c (the crossover probability), to generate two offspring chromosomes. Mutation is also described in the appendix, where we choose gene x_i with some probability and reassign $1 - x_i$ to it.

3.4. Local Search. By incorporating some a priori knowledge, a local search can efficiently reduce useless exploration and speed up the convergence of algorithms [35]. We find most optimal networks appear to be disassortative in experiments to optimize APL, and we propose a "disassortativeness-learning" technique to apply this knowledge into local search. Then, we use "Edge-Adding Learning" and "Edge-Dropping Learning" to find the local minimum of our solutions.

3.4.1. Disassortativeness Learning. The detailed algorithm is described in the appendix. We first find the added edge which has the minimum sum of the degrees of the two connected nodes and drop it. Then we find the nonexistent edge which has the maximum difference in degree between the two disconnected nodes and add this new edge to the network.

3.4.2. Edge-Adding Learning. As described in the appendix, we first judge if the diameter of the updated network has decreased to 2. If the diameter exceeds 2, we randomly add edges until the diameter equals 2. Then we output the offspring chromosomes with the updated network diameter equal to 2.

3.4.3. Edge-Dropping Learning. We select every added edge and check whether dropping the added edge will leave the network diameter unchanged. If the drop cannot increase the

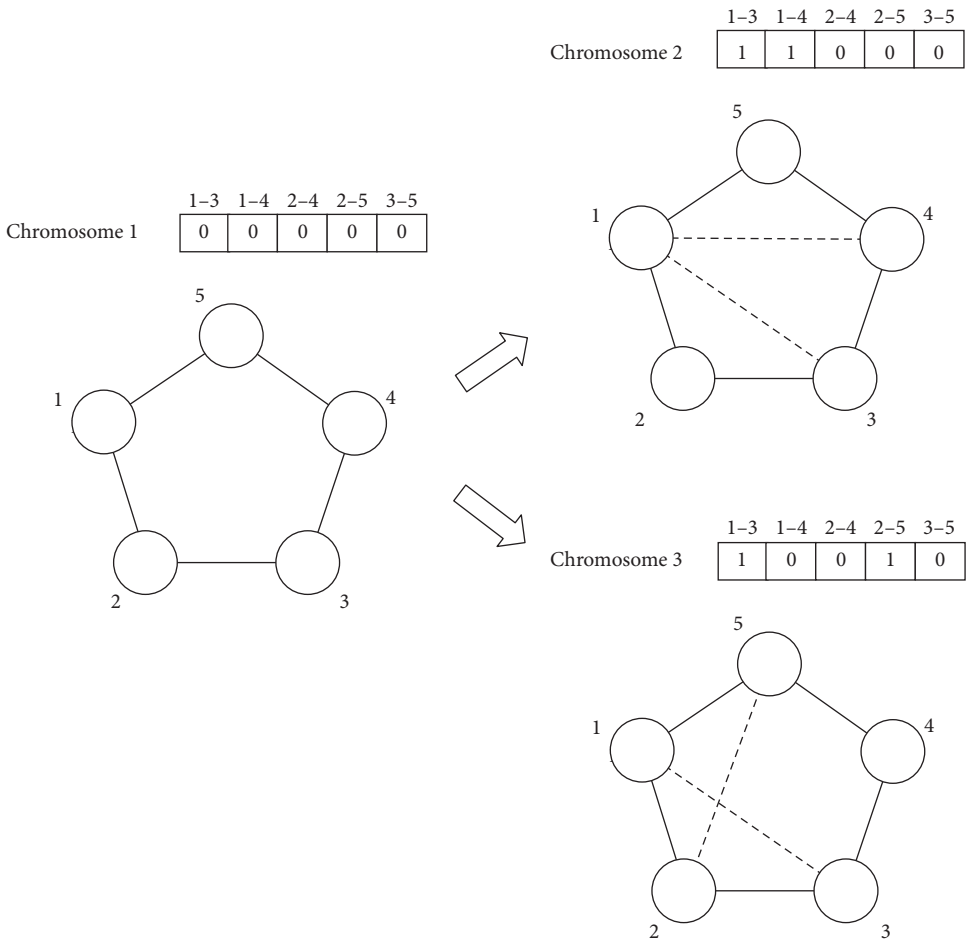


Figure 4: Illustration of representation. The numbers above the chromosomes represent the positions of corresponding nonexistent edges. The solid lines represent initial network edges, while dashed lines represent added edges, and the numbers next to nodes represent the node number.

diameter, we drop the added edge; if the drop increases the diameter, we do not drop the added edge. As a result, some useless added edges may be dropped. The appendix gives the specific procedure of Edge-Dropping Learning.

3.5. Complexity Analysis. The time complexity of MA-CD with the network size n, the number of edges of initial network m, and added edges k can be formulated as follows. Each iteration requires $S_{\text{pool}}/2$ times for crossover and S_{pool} times for mutation, where S_{pool} is the size of mating pool for the genetic operation. Since computing the network diameter costs $O(n^3)$, the total time of genetic operation is $O(S_{\text{pool}}(k + n^3))$. For local search, when executing Disassortativeness Learning, the time to update the matrix is $O(k)$; finding the added edge with the minimum sum of degree requires $O(kn)$; finding the nonexistent edge with the maximum difference in degree requires $O(n^2)$. The time for Disassortativeness Learning is $O(n^2 + kn + k)$. To perform Edge-Adding Learning and Edge-Dropping Learning, we should check at most $n(n-1)/2 - m$ genes for each chromosome and it will cost at most $O(n^5)$ to compute the updated diameter of all changed genes. Therefore, the overall time complexity of MA-CD for each iteration is $O(n^5)$.

Table 1: Parameters of the experiments.

Parameter	Meaning	Value
$I_{\max}$	The maximum iteration number	3000
S_{pop}	Population size	200
S_{pool}	Mating pool size	100
S_{tour}	Tournament size	2
P_c	Crossover probability	0.5
P_m	Mutation probability	0.5

4. Experiments

In this section, we test the performance of MA-CD on different computer-generated networks. The experiments were carried out on a 2.40 GHz CPU, 4.00 GB Memory, and Windows 10 operating system. We use MATLAB to execute the procedure. Table 1 shows the parameters necessary for the experiments.

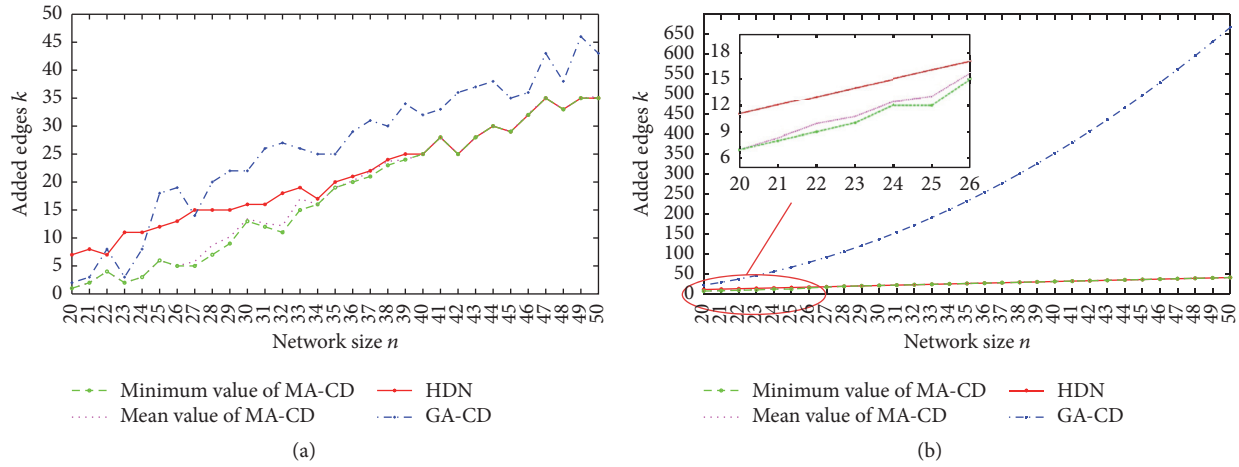


FIGURE 5: Results of optimizing the critical diameter using different methods. (a) is the result for the random network; (b) is the result for the regular network.

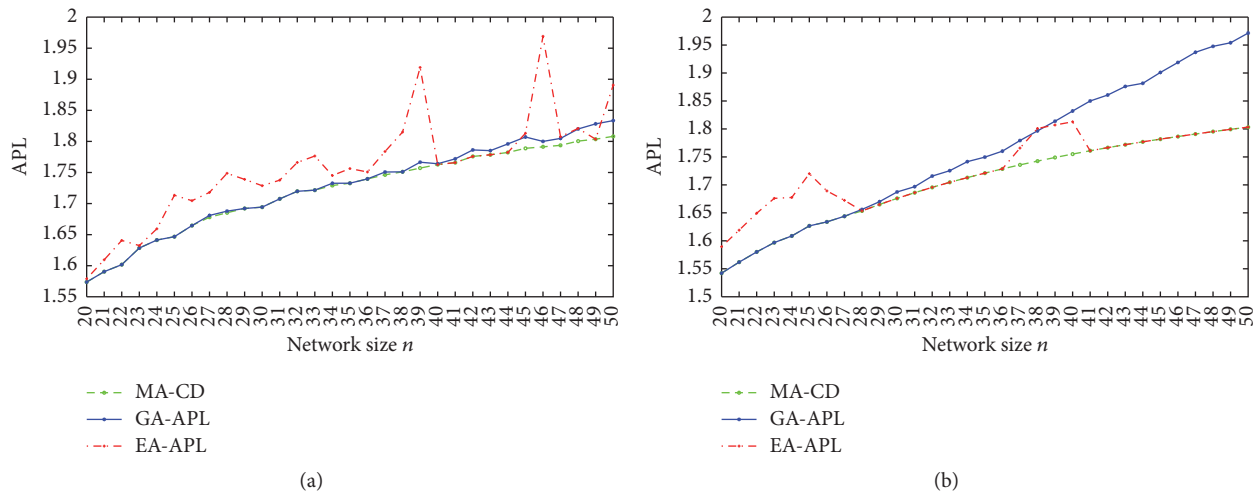


FIGURE 6: Results of optimizing APL using different methods. (a) is the result for the random network; (b) is the result for the regular network.

Our proposed algorithm is carried out ten times on two different network structures: a random network structure and a regular network structure. The detailed information of the two network structures is shown in Appendix. We compare the solution of MA-CD with that of two other methods: (1) adding edges between the highest-degree node and the other nodes as described in Section 2, denoted as HDN; (2) a kind of greedy algorithm which adds edges one by one, with each of the added edges giving the minimum diameter, denoted as "GA-CD" (greedy algorithm for optimizing critical diameter). We compare the minimum and mean value of MA-CD with the minimum value of HDN and GA-CD (the minimum and mean value of these two methods are equal).

Compared with the other two methods for optimizing the critical diameter, MA-CD can always find fewer edges to make the network diameter become the critical diameter, as shown in Figure 5. Further, we find that the results for MA-CD and HDN appear to become the same as the network size becomes larger. In other words, HDN becomes more efficient for larger networks in optimizing D_c. Compared with MA-CD and HDN, GA-CD has worse performance in optimizing D_c, especially for regular networks, even though the greedy strategy performs well in decreasing APL or diameter [23–25].

We show that our proposed method can also efficiently decrease APL. We compute the optimal networks' APL with k shortcut edges added by the MA-CD method, and then this APL is compared with that obtained using the other two methods with the same number k of edges added: (1) the EdgeEffect Algorithm, which maximizes the effect of edge insertion to optimize APL [23], denoted as "EA-APL"; (2) a greedy algorithm adding k edges one by one, with each of the added edges minimizing APL, which has previously been shown to be effective [24], denoted as "GA-APL."

The optimal networks' APL is shown in Figure 6. EA-APL performs worse for random networks, while GA-APL becomes less efficient in regular networks. MA-CD gives the best performance; it can always decrease the APL to its

Table 2: Detailed information of the two network structures.

Network structure	Network size	Number of edges	Density	Diameter	APL
Random networks	20	80	0.42	3	1.58
	21	84	0.40	3	1.62
	22	88	0.38	3	1.66
	23	92	0.36	3	1.64
	24	96	0.35	3	1.67
	25	100	0.33	4	1.74
	26	104	0.32	3	1.72
	27	108	0.31	3	1.73
	28	112	0.30	3	1.77
	29	116	0.29	3	1.78
	30	120	0.28	3	1.80
	31	124	0.27	3	1.80
	32	128	0.26	3	1.80
	33	132	0.25	3	1.84
	34	136	0.24	3	1.87
	35	140	0.24	4	1.88
	36	144	0.23	3	1.86
	37	148	0.22	4	1.94
	38	152	0.22	4	1.95
	39	156	0.21	4	1.98
	40	160	0.21	4	1.97
	41	164	0.20	4	1.98
	42	168	0.20	3	1.97
	43	172	0.19	3	1.97
	44	176	0.19	4	2.01
	45	180	0.18	3	2.02
	46	184	0.18	4	2.04
	47	188	0.17	4	2.03
	48	192	0.17	3	2.03
	49	196	0.17	4	2.05
	50	200	0.16	4	2.04
Regular networks	20	80	0.42	3	1.74
	21	84	0.40	3	1.80
	22	88	0.38	3	1.86
	23	92	0.36	3	1.91
	24	96	0.35	3	1.96
	25	100	0.33	3	2.00
	26	104	0.32	4	2.08
	27	108	0.31	4	2.15
	28	112	0.30	4	2.22
	29	116	0.29	4	2.29
	30	120	0.28	4	2.34
	31	124	0.27	4	2.40
	32	128	0.26	4	2.45
	33	132	0.25	4	2.50
	34	136	0.24	5	2.58
	35	140	0.24	5	2.65

Table 2: Continued.

Network structure	Network size	Number of edges	Density	Diameter	APL
	36	144	0.23	5	2.71
	37	148	0.22	5	2.78
	38	152	0.22	5	2.84
	39	156	0.21	5	2.89
	40	160	0.21	5	2.95
	41	164	0.20	5	3.00
	42	168	0.20	6	3.07
	43	172	0.19	6	3.14
	44	176	0.19	6	3.21
	45	180	0.18	6	3.27
	46	184	0.18	6	3.33
	47	188	0.17	6	3.39
	48	192	0.17	6	3.45
	49	196	0.17	6	3.50
	50	200	0.16	7	3.57

(1) Input: The parent chromosomes X_{parent1} and X_{parent2}.
The number of nonexistent edges of the initial network: N_{non}. Crossover Probability: P_c.
(2) $X_{\text{offspring1}} = X_{\text{parent1}}$;
(3) $X_{\text{offspring2}} = X_{\text{parent2}}$;
(4) randomly generate two positions a and b, which obey: $1 \le a < b \le N_{\text{non}}$;
(5) randomly generate $p \in [0, 1]$;
(6) if $p < P_c$
(7) randomly generate $q \in (0, 1]$;
(8) if $0 < q \le 1/3$
(9) for $i = 1; i \le a; i^{++}$
(10) $temp = X_{\text{offspring1}}(i)$;
(11) $X_{\text{offspring1}}(i) = X_{\text{offspring2}}(i)$;
(12) $X_{\text{offspring2}}(i) = temp$;
(13) end for
(14) end if
(15) if $1/3 < q \le 2/3$
(16) for $i = a + 1; i \le b; i^{++}$
(17) $temp = X_{\text{offspring1}}(i)$;
(18) $X_{\text{offspring1}}(i) = X_{\text{offspring2}}(i)$;
(19) $X_{\text{offspring2}}(i) = temp$;
(20) end for
(21) end if
(22) if $2/3 < q \le 1$
(23) for $i = b + 1; i \le N_{\text{non}}; i^{++}$
(24) $temp = X_{\text{offspring1}}(i)$;
(25) $X_{\text{offspring1}}(i) = X_{\text{offspring2}}(i)$;
(26) $X_{\text{offspring2}}(i) = temp$;
(27) end for
(28) end if
(29) end if
(30) Output: $X_{\text{offspring1}}$ and $X_{\text{offspring2}}$.

Algorithm 2: Crossover operation.

(1) Input: The parent chromosome X_{parent3}.
The number of nonexistent edges of the initial network: N_{non}. Mutation Probability: P_m.
(2) $X_{\text{offspring3}} = X_{\text{parent3}}$;
(3) randomly generate $p \in [0, 1]$;
(4) if $p < P_m$
(5) randomly generate $q \in (0,1]$;
(6) $mt = \text{ceil}(N_{non} * q)$;
(7) for $i = 1$; $i \leq mt$; i++
(8) randomly generate $r \in (0,1]$;
(9) $j = \text{ceil}(N_{non} * r)$;
(10) $X_{\text{offspring3}}(j) = 1 - X_{\text{parent3}}(j)$;
(11) end for
(12) end if
(13) Output: $X_{\text{offspring3}}$

Algorithm 3: Mutation operation.

(1) Input: the offspring chromosome: $X_{\text{offspring}}$. The adjacency matrix of the initial network: A.
(2) $X_{\text{offspring2}} = X_{\text{offspring}}$;
(3) Update the matrix A by decoding $X_{\text{offspring}}$;
(4) Find the element i of $X_{\text{offspring2}}(i) = 1$ with the minimum sum value of nodes pair degrees;
(5) $X_{\text{offspring2}}(i) = 0$;
(6) Find the element j of $X_{\text{offspring2}}(i) = 0$ with the "maximum" difference value of nodes pair degrees;
(7) $X_{\text{offspring2}}(j) = 1$;
(8) Output: $X_{\text{offspring2}}$.

Algorithm 4: Disassortativeness Learning.

(1) Input: the offspring chromosome: $X_{\text{offspring2}}$. The adjacency matrix of the initial network: A.
(2) $X_{\text{offspring3}} = X_{\text{offspring2}}$;
(3) Update the matrix A by decoding $X_{\text{offspring2}}$;
(4) Repeat
(5) randomly choose a gene $X_{\text{offspring3}}(i) = 1$;
(6) Until the diameter of the updated matrix $D = 2$
(7) Output: $X_{\text{offspring3}}$.

Algorithm 5: Edge-Adding Learning.

lowest level compared with the other two algorithms. Thus, we conclude that MA-CD can be used to optimize APL. If we can add a large number of edges to decrease APL, we just need to find a solution for optimizing D_c and then add the remaining edges randomly.

5. Conclusion

In this paper, we find a critical case in which the network diameter declines to 2 when a new edge is added to the network in the process of solving the shortcut-selection problem. Using the relationship between APL and the network diameter, we transform the problem of optimizing APL into the problem of finding shortcut edges to quickly decrease the diameter to 2, which we define as the problem of optimizing the critical diameter. Further, we suggest a method to solve this problem based on a memetic algorithm. The experimental results show that our proposed method can efficiently optimize the critical diameter and is efficient in solving the shortcut-selection problem to decrease the APL.

Appendix

See Algorithms 2, 3, 4, 5, and 6 and Table 2.

(1) Input: the offspring chromosome: $X_{\text{offspring3}}$.
The adjacency matrix of the initial network: A. The number of nonexistent edges of the initial network: N_{non}.
(2) $X_{\text{offspring4}} = X_{\text{offspring3}}$;
(3) Repeat
(4) *islocal* ← TRUE;
(5) rearrange the sequence number of the chromosome *seq*=randperm(N_{non});
(6) for $i = 1$; $i \leq N_{\text{non}}$; i++
(7) if $X_{\text{offspring4}}(seq(i)) = 1$
(8) $X_{\text{offspring4}}(seq(i)) = 0$;
(9) if the diameter of updated network $D = 2$
(10) *islocal* ← FALSE;
(11) else
(12) $X_{\text{offspring4}}(seq(i)) = 1$;
(13) end if
(14) end if
(15) end for
(16) Until *islocal* is TRUE;
(17) Output: $X_{\text{offspring4}}$.

Algorithm 6: Edge-Dropping-Learning.

Conflicts of Interest

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

Acknowledgments

This work is jointly supported by Key Project of the National Social Science Foundation of China (Grant no. 12AZD110) and Humanities and Social Science Talent Plan, Fundamental Research Funds for the Central Universities (Grant no. 2011jdgz08).

References

[1] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world' networks," *Nature*, vol. 393, no. 6684, pp. 440–442, 1998.

[2] A.-L. Barabási and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, no. 5439, pp. 509–512, 1999.

[3] R. Albert and A.-L. Barabási, "Statistical mechanics of complex networks," *Reviews of Modern Physics*, vol. 74, no. 1, pp. 47–97, 2002.

[4] M. E. Newman, "The structure and function of complex networks," *SIAM Review*, vol. 45, no. 2, pp. 167–256, 2003.

[5] A.-L. Barabási, "Scale-free networks: a decade and beyond," *Science*, vol. 325, no. 5939, pp. 412–413, 2009.

[6] M. Cai, H.-F. Du, and M. W. Feldman, "A new network structure entropy based on maximum flow," *Acta Physica Sinica*, vol. 63, no. 6, Article ID 060504, 2014.

[7] D. S. Callaway, M. E. J. Newman, S. H. Strogatz, and D. J. Watts, "Network robustness and fragility: percolation on random graphs," *Physical Review Letters*, vol. 85, no. 25, pp. 5468–5471, 2000.

[8] M. Girvan and M. E. J. Newman, "Community structure in social and biological networks," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 99, no. 12, pp. 7821–7826, 2002.

[9] F. Yu, Y. Li, and T.-J. Wu, "A temporal ant colony optimization approach to the shortest path problem in dynamic scale-free networks," *Physica A: Statistical Mechanics and Its Applications*, vol. 389, no. 3, pp. 629–636, 2010.

[10] H. Ma and A.-P. Zeng, "Reconstruction of metabolic networks from genome data and analysis of their global structure for various organisms," *Bioinformatics*, vol. 19, no. 2, pp. 270–277, 2003.

[11] B. Wang, H. W. Tang, C. H. Guo, Z. L. Xiu, and T. Zhou, "Optimization of network structure to random failures," *Physica A: Statistical Mechanics and Its Applications*, vol. 368, no. 2, pp. 607–614, 2006.

[12] D. J. Ashton, T. C. Jarrett, and N. F. Johnson, "Effect of congestion costs on shortest paths through complex networks," *Physical Review Letters*, vol. 94, no. 5, Article ID 058701, 2005.

[13] L. F. Lago-Fernández, R. Huerta, F. Corbacho, and J. A. Sigüenza, "Fast response and temporal coherent oscillations in small-world networks," *Physical Review Letters*, vol. 84, no. 12, article 2758, 2000.

[14] P. M. Gade and C. Hu, "Synchronous chaos in coupled map lattices with small-world interactions," *Physical Review E*, vol. 62, no. 5, pp. 6409–6413, 2000.

[15] J. Jost and M. P. Joy, "Spectral properties and synchronization in coupled map lattices," *Physical Review E*, vol. 65, no. 1, Article ID 016201, 2002.

[16] H. Hong, M. Y. Choi, and B. J. Kim, "Synchronization on small-world networks," *Physical Review E*, vol. 65, no. 2, Article ID 026139, 2002.

[17] M. Barahona and L. M. Pecora, "Synchronization in small-world systems," *Physical Review Letters*, vol. 89, no. 5, Article ID 054101, 2002.

[18] J. Jost and M. P. Joy, "Spectral properties and synchronization in coupled map lattices," *Physical Review E*, vol. 65, no. 1, part 2, Article ID 016201, 2002.

[19] H. Hong, M. Y. Choi, and B. J. Kim, "Synchronization on small-world networks," *Physical Review E: Statistical Nonlinear & Soft Matter Physics*, vol. 65, no. 2, pp. 95–129, 2002.

[20] Q. Xuan, Y. Li, and T.-J. Wu, "Optimal symmetric networks in terms of minimizing average shortest path length and their sub-optimal growth model," *Physica A: Statistical Mechanics and its Applications*, vol. 388, no. 7, pp. 1257–1267, 2009.

[21] O. Keren, "Reduction of the average path length in binary decision diagrams by spectral methods," *IEEE Transactions on Computers*, vol. 57, no. 4, pp. 520–531, 2008.

[22] A. Meyerson and B. Tagiku, "Minimizing average shortest path distances via shortcut edge addition," in *Proceedings of the International Workshop on Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, pp. 272–285, Springer, 2009.

[23] N. Parotsidis, E. Pitoura, and P. Tsaparas, "Selecting shortcuts for a smaller world," in *Proceedings of the SIAM International Conference on Data Mining*, British Columbia, Canada, 2015.

[24] S. H. Lee and P. Holme, "A greedy-navigator approach to navigable city plans," *European Physical Journal: Special Topics*, vol. 215, no. 1, pp. 135–144, 2013.

[25] M. Papagelis, "Refining social graph connectivity via shortcut edge addition," *ACM Transactions on Knowledge Discovery from Data*, vol. 10, no. 2, article no. 12, 2015.

[26] D. Liben-Nowell and J. Kleinberg, "The link prediction problem for social networks," in *Proceedings of the 12th International Conference on Information and Knowledge Management*, pp. 556–559, ACM, New Orleans, La, USA, November 2003.

[27] Z. Liu, Q.-M. Zhang, L. Lü, and T. Zhou, "Link prediction in complex networks: a local naïve Bayes model," *EPL*, vol. 96, no. 4, Article ID 48007, 2011.

[28] L. Lü and T. Zhou, "Link prediction in complex networks: a survey," *Physica A: Statistical Mechanics and Its Applications*, vol. 390, no. 6, pp. 1150–1170, 2011.

[29] F. Gao, K. Musial, C. Cooper, and S. Tsoka, "Link prediction methods and their accuracy for different social networks and network metrics," *Scientific Programming*, vol. 2015, Article ID 172879, 13 pages, 2015.

[30] E. D. Demaine and M. Zadimoghaddam, "Minimizing the diameter of a network using shortcut edges," in *Proceedings of the Scandinavian Conference on Algorithm Theory*, pp. 3–5, Springer, Bergen, Norway, 2010.

[31] J. Peng and G. Xu, "Average path length for Sierpinski pentagon," *International Journal of Advancements in Computing Technology*, vol. 5, no. 5, p. 724, 2013.

[32] M. Imase and M. Itoh, "Design to minimize diameter on building-block network," *IEEE Transactions on Computers*, vol. 30, no. 6, pp. 439–442, 1981.

[33] F. Neri and C. Cotta, "Memetic algorithms and memetic computing optimization: a literature review," *Swarm and Evolutionary Computation*, vol. 2, pp. 1–14, 2012.

[34] Y.-S. Ong, M. H. Lim, and X. Chen, "Memetic computation-past, present future," *IEEE Computational Intelligence Magazine*, vol. 5, no. 2, pp. 24–31, 2010.

[35] M. Gong, B. Fu, L. Jiao, and H. Du, "Memetic algorithm for community detection in networks," *Physical Review E*, vol. 84, no. 5, Article ID 056101, 2011.