Li H, Cui JT, Ma JF. Social influence study in online networks: A three-level review. JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 30(1): 184–199 Jan. 2015. DOI 10.1007/s11390-015-1512-7

Social Influence Study in Online Networks: A Three-Level Review

Hui Li (李 辉), *Member, CCF, ACM*, Jiang-Tao Cui (崔江涛), *Member, CCF, ACM*, and Jian-Feng Ma (马建峰), *Member, CCF, IEEE*

School of Cyber Engineering, Xidian University, Xi'an 710126, China

E-mail: {hli, cuijt}@xidian.edu.cn; jfma@mail.xidian.edu.cn

Received January 10, 2014; revised April 25, 2014.

Abstract Social network analysis (SNA) views social relationships in terms of network theory consisting of nodes and ties. Nodes are the individual actors within the networks; ties are the relationships between the actors. In the sequel, we will use the term node and individual interchangeably. The relationship could be friendship, communication, trust, etc. These reason is that these relationships and ties are driven by social influence, which is the most important phenomenon that distinguishes social network from other networks. In this paper, we present an overview of the representative research work in social influence study. Those studies can be classified into three levels, namely individual, community, and network levels. Throughout the study, we are able to unveil a series of research directions in future and possible applications based on the state-of-the-art study.

Keywords social network analysis, review, social influence, individual, community

1 Introduction

Social network analysis (SNA) views social relationships in terms of network theory consisting of nodes and ties. Nodes are the individual actors within the networks; ties are the relationships between the actors. The relationship could be friendship, communication, trust, etc. Compared with traditional social scientific studies, which assume that it is the attributes of individual actors that matter, SNA (social network analysis)① puts more emphasis on the relationships and ties between actors within the network. The reason is that these relationships and ties are driven by *social influence*[1-2], which is the most important phenomenon that distinguishes social network from other networks[3].

In this paper, we present an overview of the representative research work in social influence study. Those researches can be classified into three levels. Fig.1 shows the categorization of social influence study in detail. We shall follow this structure and discuss each of them.

SNA provides a both visual and mathematical analysis of human relationships. According to aforementioned discussion, instead of individual actors, SNA puts more focus on the connections between them. Social networks have also been used to examine how organizations interact with each other, characterizing many informal connections that link executives together, as well as associations and connections between individual employees at different organizations. Moreover, SNA gives companies and stakeholders new opportunities to collect information, design marketing strategies, attract customers, and so on through social networks.

In social psychology, social influence[1-2] occurs when an individual's thoughts, feelings or actions are affected by other people. A majority of social ties in social networks such as who-believes-whom, who-emails-whom, who-likes-whom or who-borrows-money-from-whom can be concluded as social influence effect. We refer to these networks which are driven by social influence effect as influence-driven social networks. In this paper, we focus on these networks and study how

Regular Paper

This work was supported by the National Natural Science Foundation of China under Grant Nos. 61173089, 61202179, 61472298, and U1135002, the Scientific Research Foundation for the Returned Overseas Chinese Scholars of State Education Ministry of China, and the Fundamental Research Funds for the Central Universities of China.

① Social network analysis theory and applications. http://train.ed.psu.edu/WFED-543/SocNet_TheoryApp.pdf, Nov. 2014.

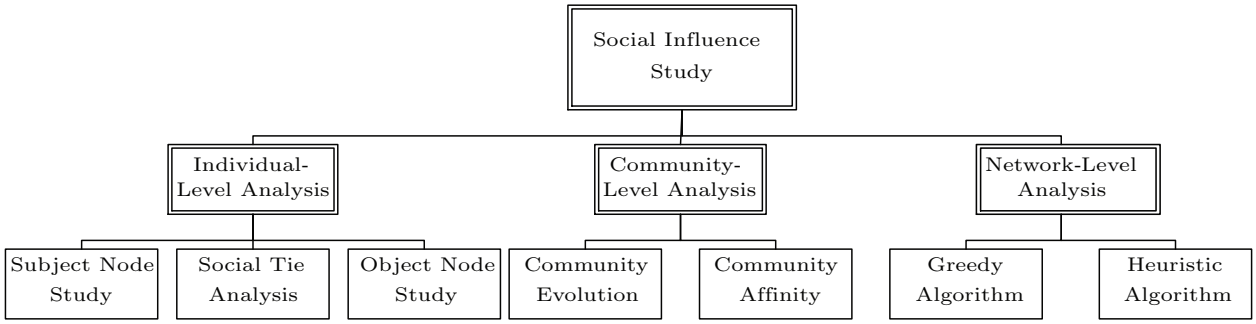


Fig.1. Social influence study categorization.

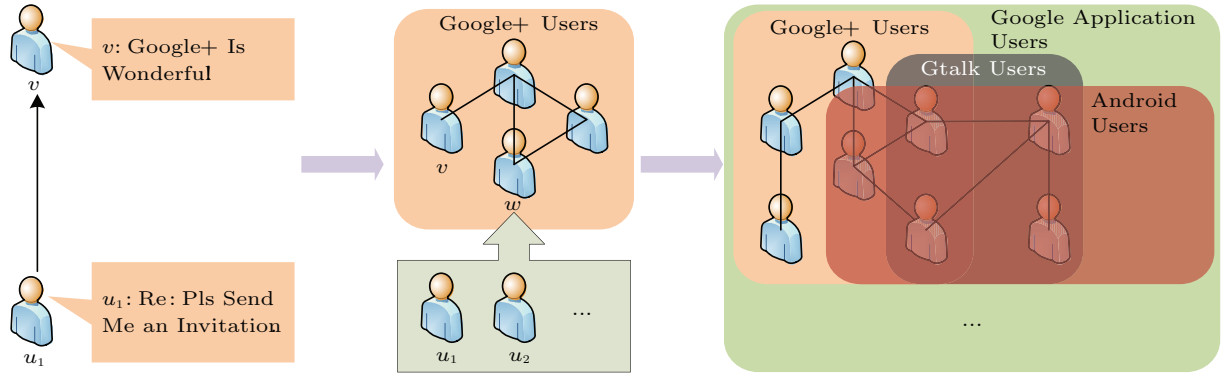


Fig.2. Three levels of social influence within social networks. (a) Atomic social influence phenomenon. (b) Effect of social influence on community. (c) Effect of social influence in the network.

the social influence phenomenon affects them. For example, Fig.2 depicts a conversation between users u_1 and v, which is a common scenario in many kinds of social networks[1,4-9]. Suppose that v recommends a new application of Google+ to his/her neighbors. One of his/her friends u_1 sees the recommendation and decides to have a try. In summary, u_1 is influenced by v to use a new application. Similarly, u_2 is also influenced by w to use this application. Such effects will result in a growth of the Google+ user community, which is shown in Fig.2(b). Thus, the evolution of a community can be traced back to the effect of social influence. Similar to the evolution of Google+ community, Gtalk user and Android user communities also evolve in this way (i.e., Fig.2(c)). Such changes will result in the evolution of the whole Google user network. It is obvious that most of the changes and evolutions of social networks can be seen as the result of the atomic social influence effect in Fig.2(a). Thus, in this paper, we propose a systematical study over the social influence related work and compare representative state-of-the-art approaches such that series of development and further research directions over social influence study can be unveiled. In the following, we shall discuss the existing work and conduct comparisons within each level in sequence.

The rest of this paper is organized as follows. We investigate the individual-level, community-level, and network-level social influence studies in Sections 2∼4 sequentially. Section 5 concludes this paper and discusses the future directions over social influence work.

2 Individual-Level Analysis

There are three main research areas with respect to individual-level social influence study: the subject node which influences others, the social tie where influence flows, and the object node which is influenced. We will discuss each of them in sequence.

2.1 Subject Node

Influential mining problem[10-12] is the focus in the research field with respect to subject node. It aims to answer the following question: given a social network, which are the most important nodes with respect to a specific application? In the following, we review some representative work in this field.

Out-Break. Leskovec *et al.*[13] discussed the problem of out-break detection in networks. The authors answered the following question: which blogs should we read to avoid missing important stories? In the paper, the authors proposed a model to find a set of nodes as sensors so that once out-break happens, the system sensors can detect the out-break as soon as possible. Nodes selected from the model can be referred to as a placement of sensors. According to their approach, the problem is formed as:

$$\max_{A\subseteq\Lambda} R(A) \text{ subject to } c(A) \leqslant B,$$

where $R(A)$ is a *placement score* of a placement A to be maximized, $c(A)$ is the related cost of such a placement A, and B is a given budget. To accomplish the task, they introduced a *penalty function*:

$$\pi(A) = \sum_i P(i)\pi_i(T(i,A)),$$

where for a placement $A \subseteq \Lambda$, $T(i,A) = \min_{s\in A}\{i,s\}$ is the time until event i is detected by one of the sensors in A, P is a given probability distribution over the events, and $\pi_i(t)$ denotes the penalty of detecting event i at time t. $T(i,\infty)$ is set to ∞, and $\pi_i(\infty)$ is set to some maximum penalty incurred for not detecting the event i. Then R can be defined as:

$$R(A) = \sum_i P(i)R(\{A\}) = \pi(\infty) - \pi(A).$$

Following this model, the authors showed by experiments that they can find a series of important nodes in blogosphere such that detecting information cascades with the minimum cost is possible.

Influential Blogger. Agarwal *et al.*[14] proposed a model to measure the significance of blogs. They developed a ranking algorithm to discover influential bloggers with the help of a post influence graph where the influence of a blog post flows along the post-post links. If I denotes the influence of a node (or a blog post p), then *InfluenceFlow* across that node is:

$$\begin{aligned} & InfluenceFlow(p) \\ = \ & \omega_{\text{in}} \sum_{m=1}^{|\iota|} I(p_m) - \omega_{\text{out}} \sum_{n=1}^{|\theta|} I(p_n), \end{aligned}$$

where ω_{in} and ω_{out} are the weights that can be used to adjust the contribution of incoming and outgoing influence, respectively. p_m denotes all the blog posts that link to p, where $1 \leqslant m \leqslant |\iota|$; p_n denotes all the blog posts that are referred by p, where $1 \leqslant n \leqslant |\theta|$; and $|\iota|$ and $|\theta|$ are the total number of in-links and out-links of p. *InfluenceFlow* accounts for the part of a post's influence that comes from in-links and out-links. The overall influence of a blog post p can be defined as

$$I(p) = \omega(\lambda)(\omega_{\text{com}}\gamma_p + InfluenceFlow(p)),$$

where ω is a weight function which rewards or penalizes the influence score of a blog post depending on the length λ of the post. ω_{com} denotes the weight that can be used to regulate the contribution of the number of comments γ_p towards the influence of blog post p.

Hence, for a blogger B, the influence score of each of B's N posts can be calculated as the blogger's *iIndex*:

$$iIndex(B) = \max_i I(p_i),$$

where $1 \leqslant i \leqslant N$. Thus, bloggers within a blogosphere can be ranked according to *iIndex*.

Heat Diffusion. Ma *et al.*[15] used heat diffusion models to find a set of k influential candidates as targets for marketing strategy in social networks. Particularly, the influence propagation is modeled as a heat diffusion process within social networks where the influence a node i receives at a particular time point t follows a heat diffusion formula as the following:

$$\begin{aligned} & \frac{f_i(t+\Delta t) - f_i(t)}{\Delta t} \\ = \ & \alpha(-\tau_i f_i(t) + \sum_{j:(v_j,\, v_i)\in E} \frac{1}{d_j} f_j(t)). \end{aligned}$$

In the above equation, τ_i is a flag to identify whether node i has any outlinks, such that $\tau_i = 0$ if node i does not have any outlinks, otherwise, $\tau_i = 1$. Solving the equation, the influence that nodes receive at time point t can be expressed as the following.

$$f(t) = e^{\alpha t H} f(0), \quad H_{ij} = \begin{cases} 1/d_j, & \text{if } (v_j, v_i) \in E, \\ -\tau_i, & \text{if } i = j, \\ 0, & \text{otherwise.} \end{cases}$$

Based on this idea, the top-k candidates whose heat is diffused to the largest scope are selected using a greedy algorithm.

Twitter Authority. In a different media, Pal and Counts[16] used the count of original tweets, conversational tweets, and re-tweets of a tweeter as features to rank the *authority* of each tweeter in the context of different topics. They employed a Gaussian mixture model to compute the authority score of each tweeter.

Formally, the authority score for twitter i can be computed as the following:

$$R_G(x_i) = \prod_{f=1}^{d} \left[\int_{-\infty}^{x_i^f} N(x; \mu_f, \sigma_f) \right]^{w_f}.$$

In the above equation, w_f is the weight that is put on feature f; x_i^f is the associated value of node i on feature f; and $N(x; \mu_f, \sigma_f)$ is the univariate Gaussian distribution with model parameters as μ_f and σ_f. The authority score defined above helps in devising a total ordering under "$\leqslant$" over all the users. To validate their results, they conducted a survey to rate the authority of the tweeters and use it as the ground truth for authority ranking.

Conformity. Li *et al.*[17] proposed a framework that computes both the influence and the conformity of an arbitrary individual. Conformity is defined as the probability that an individual is willing to accept others' opinion. They proposed an iterative algorithm, of which each iteration computes a pair of indices (influence Φ and conformity Ω) for each individual as follows:

$$\begin{aligned} \Phi^{k+1}(u) &= \sum_{\overrightarrow{vu} \in E^+} \Omega^k(v) - \sum_{\overrightarrow{wu} \in E^-} \Omega^k(w), \\ \Omega^{k+1}(u) &= \sum_{\overrightarrow{uv} \in E^+} \Phi^k(v) - \sum_{\overrightarrow{uv} \in E^-} \Phi^k(w). \end{aligned}$$

Moreover, they proved that the iterative algorithm was guaranteed to converge. In order to justify the result of the indices computation scheme, they applied the indices in link prediction task. Empirical results showed that using the computed indices of both influence and conformity, link prediction accuracy can be improved significantly.

The comparison of all the aforementioned work is summarized in Table 1. They do not address the following issues. First of all, there exist positive edges and negative edges in Twitter where positive edges represent agreement relationships while negative ones represent disagreement relationships. The aforementioned models fail to distinguish negative edges from positive ones. They treat both kinds of edges equally. Secondly, these existing researches lack justification over the influential mining result.

2.2 Social Tie

A common problem that is related with *social tie* is link mining which can be described as follows: given a pair of unconnected nodes i, j, what is the probability that they are connected in a future time t'? Since social networks consist of individuals, the links between the individuals tend to mirror or, in some cases, establish new information propagation kernels. Studying the *social tie* allows the discovery and usage of information dissemination within social networks. We review some representative link mining work in the following and then compare them with our work.

Common Neighbor. Liben-Nowell and Kleinberg proposed an algorithm to solve the problem of link prediction[18]. They developed approaches to link prediction based on measures of the "proximity" of nodes in a network. Experiments on large social networks suggest that information about future interactions can be extracted from network topology alone. Their approach is based on the idea that two nodes x and y are more likely to form a link if $\Gamma(x)$ and $\Gamma(y)$ have large overlap where $\Gamma(x)$ denotes the neighbors of x. It follows the natural intuition that such node pairs represent authors with many colleagues in common, and hence are more likely to come into contact. Thus, the authors set several different score measures to evaluate the probability x and y will cooperate in future.

$$score(x, y) := |\Gamma(x) \cap \Gamma(y)|, \quad (1)$$

$$score(x, y) := |\Gamma(x) \cap \Gamma(y)| / |\Gamma(x) \cup \Gamma(y)|, \quad (2)$$

$$score(x, y) := |\Gamma(x)| \times |\Gamma(y)|. \quad (3)$$

(1) is the concept of common neighbors. Newman[19] has verified a correlation between the number of common neighbors of x and y and the probability that they will collaborate in the future. (2) utilizes *Jaccard coefficient* which is widely used in information retrieval, measuring the similarity of features

Table 1. Influential Mining Models Comparison

	Negative Edge	Edge Type	Text Analysis	Justification
Out-Break[13]	–	Undirected	–	–
Influential blog[14]	–	Directed	✓	–
HeatDiff[15]	–	Directed	–	–
TwitterAuth[16]	–	Directed	✓	Survey
CASINO[17]	✓	Directed	✓	Link prediction

between x and y. (3) is based on *preferential attachment* model[20-21] which claims that the probability a new edge involves node x is proportional to $\Gamma(x)$. The authors also proposed that the probability of connecting x and y was correlated with the product of the number of collaborators of x and y.

Historical Study. O'Madadhain *et al.*[22] proposed an algorithm for prediction and ranking of link existence based on event-based network data. The main contribution of their paper is predicting the probability for a pair of individuals to co-participate in the same event. The problem can be formally described as follows: "given the information of a series of events, will entities v_j and v_k co-participate in at least one event in a future specified interval?" The authors treat this task as a data-driven classification problem (in which "co-participating" is one class, and "not co-participating" is the other). The methods used are primarily probabilistic classifiers, which assign a probability to each class conditioned on the values of a set of specified features, whose nature may vary depending on the dataset. They defined the conditional probability as the following:

$$p(v_j, v_k \in P_{t,\, t+\Delta t} | f(\Upsilon_{1,\, t}, X, Y) = \boldsymbol{w}),$$

where $v_j, v_k \in P_{t,\, t+\Delta t}$ is a binary proposition defining whether entities v_j and v_k co-participate in any event during the period $t, t + \Delta t$, f is a function returning a vector $\boldsymbol{w}$ of feature values, $\Upsilon_{1,\, t}$ is the historical event data up to time t, and X, Y are the relevant entities and event co-variate data. By computing the conditional probability as above, they are able to predict the probability that a pair of individuals will co-participate in a event.

Signed Edge Prediction. In many social networks, there exist different attitudes attached to the edges. For example, in Epinions, the social ties between users may express trust or distrust; in Slashdot, the social tie may indicate agreement or disagreement. The edges attached with trust/agreement can be labeled as *positive*; the edges representing distrust/disagreement can be labeled as *negative*. Leskovec *et al.*[23] investigated some of the underlying mechanisms that determine the signs of links in large social networks where interactions can be both positive and negative. They used logistic regression to predict the signs of edges in signed networks by exploiting a series of features as following.

$$P(+|\boldsymbol{x}) = \frac{1}{1 + e^{-(b_0 + \sum_i^n b_i x_i)}}.$$

Within the above formula, $\boldsymbol{x}$ is a vector consisting of features $(x_1, \ldots, x_n)$ and $b_0, \ldots, b_n$ are the coefficients learned from the training data. All these features are solely extracted from the structure of the network. They showed that their model significantly improves previous approaches.

OOLAM. Cai *et al.*[24] proposed another algorithm called OOLAM (an opinion-oriented link analysis model) by introducing a new feature (i.e., influence) aside from the 7-dimensional degree features in [23]. A PageRank-like algorithm was developed to compute the influence of individual users and then use it as another feature in an SVM classifier to predict the signs of edges. They showed that by taking into account the social influence of individual users, the accuracy of edge sign prediction could be significantly improved. Based on this, they categorized influence personae into *positive persona*, *negative persona*, and *controversy persona*. *Positive* and *negative personae* represent users with high positive and negative influence, respectively. The last kind of *controversy persona* represents a group of individuals who are liable to be challenged or supported by many.

Blog Cascade Affinity. Li *et al.*[25-26] investigated the phenomenon of blog cascading, where they test a series of macroscopic and microscopic features that may affect the probability for an individual to join an arbitrary blog cascade. They showed that the most important feature that affects the cascade joining behavior is the number of friends within a cascade using both ANOVA test and learning task. Moreover, they feed all the features that have been found important to two different learning algorithms, SVM and BiMRF (Bipartite Markov Random Field). Experimental results show that utilizing the features and the learning algorithm, they are able to predict the probability of an arbitrary individual to join a given cascade.

The difference of all the aforementioned algorithms is summarized in Table 2.

2.3 Object Node

The majority of existing work focuses on either the subject who is influencing others or the edge where influence propagates. They have not systematically investigated the object in social influence phenomenon. The only work regarding to the object node which is influenced by others can be found in viral marketing research. The majority of marketers are interested in evaluating the probability for a user to purchase a particular product. It is of much importance for on-line advertising and information propagation.

Table 2. Link Mining Models Comparison

	Negative Edge	Edge Type	Social Influence Feature	Object Node
Common neighbor[18]	–	Undirected	–	–
Historical study[22]	–	Undirected	–	–
Signed edge prediction[23]	✓	Directed	–	–
OOLAM[24]	✓	Directed	Node influence	–
Blog cascade affinity[25-26]	–	Directed	Number of friends, popularity of participants, citing factor, initiator-media link	✓

In summary, most of the aforementioned work in the field of individual-level study has only focused on the subject node and social tie in social influence phenomenon. They have ignored the object node. The research investigating the object node that is influenced by others is currently receiving much attention.

3 Community-Level Analysis

As an effect of social influence phenomenon, people form *communities* in a network. Consequently, a great deal of work has focused on mining implicit communities in online social networks[27]. A *community* is a group of people with some common properties. Community mining is in fact subgraph identification[28] or node clustering[29]. There are two main research directions in this area: community evolution and community affinity.

3.1 Community Evolution

The basic problem to be addressed with respect to community-level study is how to evaluate and extract communities which are highly evolving. To solve this problem, researchers made a well accepted assumption that communities are groups of people who are relatively stable along the evolution of network[30]. Based on this assumption, many frameworks have been proposed to find communities within evolutionary networks[31]. We present some of them in the following.

FacetNet. Lin *et al.*[32] analyzed communities and the evolution of them through a unified process. An *adjacency matrix factorization* approach is introduced in the paper. According to the approach, the adjacency matrix of a network $\boldsymbol{W}$ can be factorized as $\boldsymbol{W} = \boldsymbol{X} \cdot \boldsymbol{\Lambda}^{\mathrm{T}}$ where $\boldsymbol{X} \in \mathbb{R}_{+}^{n \times m}$ and $\sum_i x_{ij} = 1$[33]. In addition, $\boldsymbol{\Lambda}$ is an $m \times m$ non-negative diagonal matrix. According to the paper, $\boldsymbol{X} \cdot \boldsymbol{\Lambda}$ fully characterizes the community structure in the network. Based on the factorization, the authors proposed *snapshot cost* which captures how well the community structure $\boldsymbol{X} \cdot \boldsymbol{\Lambda} \cdot \boldsymbol{X}^{\mathrm{T}}$ fits $\boldsymbol{W}$ at time t. It can be expressed as $\mathcal{CS} = D(\boldsymbol{W} \| \boldsymbol{X} \cdot \boldsymbol{\Lambda} \cdot \boldsymbol{X}^{\mathrm{T}})$ where $D(A\|B)$ is the KL-divergence between A and B. Similarly, the authors proposed another concept called *temporal cost* to measure how consistent the community structure at time t is with respect to that of $t-1$. It can be calculated as $\mathcal{CT} = D(\boldsymbol{Y} \| \boldsymbol{X} \cdot \boldsymbol{\Lambda})$ where $\boldsymbol{Y} = \boldsymbol{X}_{t-1} \cdot \boldsymbol{\Lambda}_{t-1}$. Both of the cost are then linearly combined together into a total cost formula as follows.

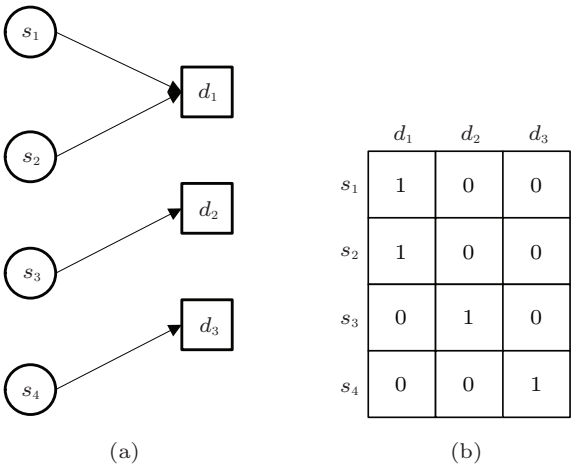
$$\begin{aligned} cost &= \alpha \times D(\boldsymbol{W} \| \boldsymbol{X} \cdot \boldsymbol{\Lambda} \cdot \boldsymbol{X}^{\mathrm{T}}) + \\ &\quad (1-\alpha) \times D(\boldsymbol{Y} \| \boldsymbol{X} \cdot \boldsymbol{\Lambda}). \end{aligned}$$

Following this approach, communities can be detected by minimizing this *cost*. The authors justified this model in both synthetic dataset and real-world datasets.

GraphScope. GraphScope[34] is a parameter-free algorithm where the minimum description length (MDL) principle is employed to extract communities as well as their changes. According to this model, the evolution of a graph G is modeled as a graph stream $\mathcal{G} = \{\boldsymbol{G}^{(1)}, \boldsymbol{G}^{(2)}, \ldots, \boldsymbol{G}^{(t)}, \ldots\}$. The goal of the model is to find good partitions of source and destination nodes that describe the community structure. In order to evaluate the quality of partitions, they proposed an encoding scheme to represent temporal partitions, graphs and subgraphs. For example, Fig.3(a) depicts a social network where circles represent source node and squares denote destination nodes. The adjacency matrix of the graph is shown in Fig.3(b). Conceptually, such a binary matrix can be stored as a binary string with length mn, along with the two integers m and n. For example, Fig.3(b) can be stored as 110000100011 (in column major order), along with two integers 4 and 3. To reduce the space, more scientific schemes such as Huffman coding can be employed to store that string. The code length for that is accurately estimated as $mnH(\boldsymbol{G}^{(t)})$ where $H(\boldsymbol{G}^{(t)})$ is the entropy of binary string of $\boldsymbol{G}^{(t)}$ that can be computed as follows.

$$H(\boldsymbol{G}^{(t)}) = -p(1)\log p(1) - p(0)\log p(0).$$

In the equation, $p(1)$ (resp., $p(0)$) denotes the ratio of 1 (resp., 0) in the entrance of matrix $\boldsymbol{G}^{(t)}$. After encoding the subgraphs using this method, MDL is employed to evaluate the cost between the codes of different subgraphs. In this way, those subgraphs with the minimum cost can be viewed as communities as they do not change much between snapshots.



	d_1	d_2	d_3
s_1	1	0	0
s_2	1	0	0
s_3	0	1	0
s_4	0	0	1

Fig.3. Coding of GraphScope.

MONIC. MONIC[35] models the changes within each individual community. The authors first clustered the graphs at each snapshot. The goal of the model is to find similar clusters across different snapshots of the same graph. These similar clusters are viewed as the same community evolving from one snapshot to another. In order to do this, they proposed a measure to evaluate the overlap from cluster X at time t_i to cluster Y at time t_j $(t_i < t_j)$, which can be described as the following.

$$overlap(X,Y) = \frac{\sum_{a \in X \cap Y} age(a, t_j)}{\sum_{x \in X} age(x, t_j)},$$

where $age(x, t_j)$ describes the weight of record x at time t_j. Using this measure, the best matching cluster, which exhibits the highest overlap value with X, can be selected. The selected cluster is then viewed as the identity cluster that evolves from X. The authors also identified a set of key events, such as *survive*, *split*, *disappear*, which are further studied based on the changes of clusters.

Stable Cluster. Bansal *et al.*[36] also viewed the evolution of a graph as a stream of graphs and clustered each snapshot into partitions. They further used Jaccard similarity to measure the similarity between clusters at different snapshots. It is computed as the intersection of community members at different time points divided by the union of the members.

$$overlap(C_1, C_2) = \frac{|C_1 \cap C_2|}{|C_1 \cup C_2|}.$$

In this way, a *cluster evolution* graph is formed. An example is shown in Fig.4. Each node is a cluster at a particular time point while edge weight is the Jaccard similarity between clusters. The nodes in column "Day 1" represent the clusters identified in the graph at the first snapshot. An edge pointing from node "12" to node "21" indicates that the cluster C_{12} is similar to C_{21} with similarity of 0.5. Such a high similarity means that C_{21} is probably evolved from C_{12}. Thus, from this graph, it is easy to find the most stable cluster by tracing each path from "source" to "sink" and select the one which exhibits the highest accumulated weight.

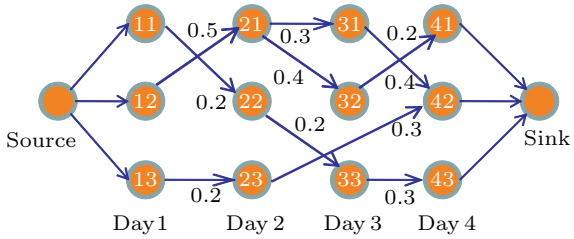


Fig.4. Cluster evolution graph.

Event-Driven. Asur *et al.*[37-38] studied the evolution of graphs to understand particular patterns for communities and individuals over time. They defined the overlap ratio of vertex set for the same community over two timestamps as $overlap(x,y) = \frac{|x \bigcap y|}{\min(|x|,|y|)}$. *Popularity index* for community j at time i is:

$$PI(C_i^j) = \sum_{x=1}^{V_i} Join(x, C_i^j) - \sum_{x=1}^{V_i} Leave(x, C_i^j).$$

Influence index for node x is defined as: $Inf(x) = |moves(companions(x))|/|moves(x)|$.

Affrank. Li *et al.*[39-40] proposed a framework, namely Affrank, to predict the rank of products or topics based on historical information utilizing a series of methods evaluating the evolution of communities. Specifically, they employed a DTW (dynamic time warping) distance that evaluates the distance between the evolutionary statistics from different communities (i.e., topics or products). Moreover, they fed the results of DTW into an ARX (autoregressive-moving-average) model that predicts the ranking of product communities. Empirical results justify that their solution exhibits better performance than state-of-the-art algorithms.

The aforementioned studies have proposed different ways to evaluate the evolution of a community. They are summarized in Table 3.

3.2 Community Affinity

Another area of community-level study is finding the factors that exert effect on a community's ability to attract new members. Formally, the ability of a community to attract new members is referred to as *community affinity*[39-40]. We describe some representative work which investigates community affinity.

Group Formation. Backstrom *et al.*[31] demonstrated that the community size, the connectivity between community members, and the number of friends a user has in a community have strong influence on community affinity. They modeled the affinity problem as a standard classification task: the nodes which eventually join a group are denoted as positive while those which do not are denoted as negative samples. Two real-world social network datasets (LiveJournal and DBLP) were studied in that paper. They extracted several features (friends in the group, clustering coefficient of the group, etc.) for each node and employed a decision tree to predict the sign of node samples. Their empirical results show that a large clustering coefficient is negatively related to the growth of community. Additionally, they also show that community affinity is highly affected by the existence of friends in the target community.

Dynamics in VM. Leskovec *et al.*[41] showed that an individual's probability of buying a DVD increased with the number of recommendations he/she has received. There is a *saturation point* at the value of 10, which means after a person receives 10 recommendations on buying a particular DVD, the probability of buying does not increase anymore. Moreover, by studying the correlation between people's buying behavior and the number of recommendations they have received, the authors found that such correlation is significant in some products (i.e., DVD and Book). Further, a logistic regression model was employed to test the success of recommendation based on the findings on the affinity of buying a product.

Propagation in Flickr. Cha *et al.*[42] conducted a study on Flickr over the same problem. The authors investigated the correlation between pictures' popularity and network topology. Specifically, the number of fans for each picture at different distances from the uploader is examined in the paper. It is reported that the probability for a user to become a fan of a photo increases with the number of his/her friends who are already fans of the photo. On the other hand, the authors also reported that the number of fans a picture has is also affected by the elapsed time since the picture was uploaded.

Affrank. The framework proposed by Li *et al.*[39-40] has utilized a series of novel features, such as affinity rank history, affinity evolution distance, and average ratings. They trained a regression model using ARX technique and predicted the affinity ranks of product communities within social rating networks such as Epinions.

Table 4 summarizes the differences as well as the similarities among all the approaches aforementioned.

4 Network-Level Analysis

Social influence phenomenon has turned social networks into important channels for information propagation. For instance, rumors spreading, product promotion and "word-of-mouth" communications all depend on social influence effect within individuals. Due to the difficulty in tracking a specific information when it is transmitted by people, most current understandings of information spreading in social networks come from models of indirect measurements[43]. It is strongly related to the research of epidemic and contagion problems which study the propagation models of viruses and diseases.

A main research goal in this field is to solve the IM

Table 3. Summary of Community Dynamics Research

	Difference Measure	Parameter-Free	Evolution Pattern Comparison
GraphScope[34]	Minimum description length	✓	–
MONIC[35]	Cluster intersection	–	–
Stable cluster[36]	Jaccard similarity	–	–
FacetNet[32]	KL-divergence	–	–
Event-driven[37-38]	Overlap ratio of vertices	–	–
AffRank[39-40]	ΔAffinity and ΔAffinity rank	✓	DTW distance

Note: Δ means the change of affinity (rank) between consecutive time steps.

Table 4. Summary of Community Affinity Research

	Features Adopted	Method	Target Network
Group formation[31]	Friends in the group, group size, clustering coefficient	Decision tree	Collaboration network
Dynamics in VM[41]	Number of recommendations, price, number of reviews	Logistic regression	Recommendation network
Propagation in Flickr[42]	Number of fans, number of friends, elapsed time	Statistic analysis	Flickr network
AffRank[39-40]	Affinity rank history, affinity evolution distance, average rating besides above features	ARX	Social rating network

(*influence maximization*) problem. Formally, the problem can be described as follows.

Definition 1 (Influence Maximization). *Given a social network $G(V, E)$, a specific cascade model C and a budget number k, the influence maximization problem is to find a set of nodes S (referred to as seeds) in G, which we call as seed set, where $|S| = k$ such that according to C, the expected number of nodes that are influenced (referred to as influence spread) by S (denoted by $\sigma(S)$) is the largest. It can be expressed using the following formula.*

$$S = \underset{S' \subseteq V, |S'|=k}{\arg\max} \sigma(S').$$

The *influence spread* in the network may follow one of the following cascade models.

• *Independent Cascade (IC) Model.* Let A_i be the set of nodes that are influenced in the i-th round and $A_o = |S|$. For any $(u, v) \in E$ such that u is already in A_i and v is not yet influenced, v is influenced by u in the next $(i + 1)$-th round with an independent probability p, which is referred to as the *propagation probability*. Thus, if there are t neighbors of v that are in A_i, then $v \in A_{i+1}$ with a probability $1 - (1 - p)^t$. This process is repeated until A_{i+1} is empty.

• *Weighted Cascade (WC) Model.* Let $(u, v) \in E$. In this model, if u is influenced in round i, then v is influenced by u in round $(i + 1)$ with a probability $1/v.degree$. Thus, if v has t neighbors influenced at the i-th round, then the probability for a node v to be influenced in the next round is $1 - (1 - 1/v.degree)^t$.

• *Linear Threshold (LT) Model.* In this model, each node v has a threshold θ_v uniformly and randomly chosen from 0 to 1. This represents the weighted fraction of v's neighbors that must become influenced (active) in order for v to be influenced. All nodes that were influenced in step $(i-1)$ remain so in step i, and any node v is influenced when the total weight of its influenced neighbors is at least θ_v.

In fact, the WC model is often viewed as a variant of IC model in that it also assumes influence propagation through an edge depends on a probability. In summary, many studies have been proposed to solve IM in both IC (resp., WC) and LT models, which are listed in Table 5.

Table 5. Cascade Models of Different Algorithms

Cascade Model	Algorithms
IC (WC) and their variants	General Greedy[3], CELF[13], MixGreedy[44], CGA[45], CINEMA[46], IPA[47], MSA[48], PMIA[49], MIA-N[50]
LT	General Greedy[3], LDAG[51], SimPath[52]

Notably, influential mining in individual-level analysis discussed in Section 2 aims to evaluate the influence of individuals through their personal features and characteristics, such as expertise, historical action logs, age, interest, and so on. In contrast, influence maximization differentiates from influential mining problem in the following aspects. Firstly, instead of evaluating the individual influence, IM aims to find k nodes whose accumulative influence is maximized. In this process, the influence of each individual does not need to be exactly computed. Secondly, IM focuses on evaluating influence from the topological structure of the network, which depends on the information cascade model. Thirdly, influential mining problems are always modeled as supervised or semi-supervised learning tasks, while IM problems can only be modeled as unsupervised problems and discrete optimization tasks.

To achieve the goal of *influence maximization*, existing studies have proposed many methods to find some nodes in the network to spread the information initially with minimal cost and maximal influence. The algorithmic study towards the problem of influence maximization within social networks can be traced back to the year 2001 when Domingos and Richardson[53-54] proposed a probabilistic method to predict the number of influenced nodes in a network to help companies determine the potential customers for marketing a product. They modeled the customers as a social network and adopted *Markov random field* method to study the propagation of influence. After that, many algorithms have been proposed to solve the problem. These algo-

rithms can be classified into two groups, greedy algorithms and heuristic algorithms.

4.1 Greedy Algorithms

Greedy algorithms are a group of greedy approaches which greedily select the node with the maximum marginal gain towards the existing seeds in each iteration. These algorithms provide high quality results which are within 63% of the optimum solution. However, they suffer from the running time.

General Greedy (*Hill-Climb*). Consider an arbitrary function $f(\cdot)$ that maps subsets of a finite ground set U to non-negative real numbers. Then f is *submodular* if it satisfies a natural "diminishing returns" property: the marginal gain from adding an element to a set S is at least as high as the marginal gain from adding the same element to a superset of S, namely S'. Formally, a function f is submodular if it satisfies the following for $\forall S \subseteq S'$:

$$f(S \cup \{x\}) - f(S) \geqslant f(S' \cup \{x\}) - f(S').$$

The submodular function has been studied extensively in the field of mathematics[55]. Kempe *et al.*[3] proved that several popular cascade models are submodular (e.g., Independent cascade model, weighted cascade model). Particularly, if the cascade model is fixed and the influence function is proved to be submodular and monotone, then they showed that the influence maximization problem could be solved by starting with an empty set and iteratively selecting the nodes which achieve the best marginal gain towards the current seeds. Note that although the minimization of a submodular function is shown to be within polynomial time[56-57], the maximization of that is proved to be NP-hard. Hence, the influence maximization problem is also NP-hard.

Some work focuses on designing approximated algorithms to achieve the maximization task in polynomial time[58]. Kempe *et al.*[3] also proposed an approximate algorithm based on greedy strategy. According to [2], if a greedy maximization algorithm of a submodular function f returns the result A_{greedy}, then the following holds:

$$f(A_{\text{greedy}}) \geqslant (1 - 1/e) \max_{|A| \leqslant k} f(A).$$

That is, a greedy algorithm can give near optimal solution to the problem of maximization of a submodular function. Accordingly, Kempe *et al.* guaranteed that their greedy algorithm could achieve influence spread within $(1-1/e)$ of the optimal influence spread. However, the proposed algorithm takes $O(knmR)$ time to solve the influence maximization problem (see Table 6), which is computationally very expensive for real-world social networks.

Table 6. Time Complexity of Different Algorithms

Algorithm	Time Complexity
General Greedy	$O(knRm)$
CELF	$O(knRm)$
MixGreedy-IC	$O(kRm)$
MixGreedy-WC	$O(kTRm)$
DegreeDiscount-IC	$O(k \log n + m)$
MSA	$O(Tk\bar{d})$
PMIA	$O(nt_{i\theta} + kn_{o\theta}n_{i\theta}(n_{i\theta} + \log n))$
MIA-N	$O(nt_o + kn_o n_i(\min(k, h_{\max})n_i + \log n))$
CGA	$O(m + nRlm' + klRm' + kRm')$
IMCD	$O(\lvert A\rvert nm^2 + k\lvert A\rvert m^2)$
CINEMA	$O(k'm'n' + kTRm')$
IPA	$O(\lvert V\rvert\lvert O_v\rvert n_{vu}) + O\left(\frac{\lvert O_v \cup \{v\}\rvert}{c} n_{vu} + c\right)$

Note: $\bar{d}$ denotes the average degree of nodes.

CELF (*Cost-Effective Lazy Forward*). In order to reduce the computation time of influence maximization problem, Leskovec *et al.*[13] proposed an algorithm called CELF (cost-effective lazy forward) that was reported to be 700 times faster than the simple hill-climbing algorithm proposed by Kempe *et al.* on real networks. CELF is also based on the submodular property of the cascade influence function. They observed that in each round, the hill-climbing algorithm needs to recompute the influence $\sigma(v)$ of each node v. In most cases, the *marginal gain* of a node v, given by $\sigma(v|S) = \sigma(S \cup \{v\}) - \sigma(S)$, may not change significantly between consecutive rounds. Thus instead of recomputing the spread for each node at every round of seed selection, CELF performs a *lazy* evaluation. Initially, it marks all $\sigma(v|S)$ as *invalid*. When selecting the next seed, it scans the nodes in decreasing order of their $\sigma(v|S)$. If the $\sigma(v|S)$ for the top node is *invalid*, then CELF recomputes it and inserts it into the existing order of the $\sigma(v|S)$ (e.g., using a priority queue). In many cases, the recomputation of $\sigma(v|S)$ will lead to a new value which is not significantly smaller. Consequently, often the top element will remain at the top even after recomputation. In the worst case, during each selection, CELF needs to recompute the marginal gain for all the remaining nodes resulting in a worst-case time complexity of $O(kmRn)$ (Table 6).

MixGreedy. Chen *et al.*[44] reduced the computation of marginal gain from $O(mn)$ to $O(m)$. To compute the influence of each node, they processed the graph by removing unnecessary edges according to the cascade model such that computing the marginal gain for all the nodes only requires a linear traversal over the network. Their *random removal* method can be summarized as following. Since in the *independent cascade* (IC) *model*, each edge has the probability p to take effect in the cascade, each edge in the graph G is randomly removed with a probability $1-p$. In this way, G is separated into pieces and each piece is the scope of the node v's influence spread within it. Thus, computing the marginal gain of a node will only require a linear traversal of the scope. As a result, the computation of the marginal gain of a node only requires $O(m)$ operations, which is the complexity of removing edges from G. Similarly, when the network follows the *weighted cascade* (*WC*) *model*, each edge is removed with a probability $1-1/v.degree$. The influence of each node can be computed by adding the gain in R iterations of the random removal process. Based on this, they proposed the *MixGreedy* algorithm which follows the random removal process in computing the marginal gains and then utilized the CELF approach for updates. The time complexities of the MixGreedy approach for the two aforementioned cascade models are $O(kRm)$ and $O(kTRm)$, respectively (Table 6). They empirically demonstrated that the running time of MixGreedy was smaller than that of CELF.

CGA (*Community-Based Greedy Algorithm*). Wang *et al.*[45] proposed a community-based greedy solution to the problem. In order to reduce the running time, they first detected communities based on IC model and then mined the top-k nodes across communities. They developed a cost function that optimizes the community assignment in mobile networks. In fact, their community detection process under IC model takes $O(m + nRlm' + klRm')$ which is time consuming in huge networks. Second, CGA selects each influential node based on a unified optimization formula, which requires the information for all communities stored in a unified space.

IMCD (*Influence Maximization Under Credit Distribution Model*). Goyal *et al.*[59] proposed a *credit distribution* (CD) *model* that leverages on historical *action logs* of a network to learn how influence flows in the network and used this to estimate influence spread. An *action log* is a set of triples $(u, a, t) \in A$ which says user u performed action a at time t. The basic idea is that if user v takes action a and later on v's friend u does the same, then the authors assumed that action a has propagated from v to u. Based on this assumption, the CD model assigns "credits" to the possible influencers of a node u whenever u performs an action. The sophisticated variant of this model distinguishes between different influenceabilities of different users by incorporating a *user influenceability function*. It is defined as the fraction of actions that u performs under the influence of at least one of its neighbors (e.g., v) and is learnt from the historical log data. In contrast to our approach, this model suffers from two key limitations. Firstly, it depends on the availability of a large number of historical action logs to compute influence probability as well as user influenceability. Unfortunately, historical action logs may not be available to end-users in many real-world social networks.

CINEMA. CINEMA[46] is designed for IC and WC (resp. C^2) models. The authors proposed a novel cascade model, namely C^2 model, that takes into account the conformity of individuals, such that the probability for a node to be influenced is proportional to not only the influence of the active node but also the conformity of the target node. Moreover, they proposed a divide-and-conquer scheme that can significantly reduce the running time of the algorithm while preserving high quality seeds result. Empirical results have justified that their model can solve influence maximization problem in large networks within tens of hours.

IPA. Recently, Kim *et al.*[47] proposed an approximate influence maximization algorithm called IPA under the IC model that efficiently approximates influence by considering an independent influence path (influence paths between two nodes) as an influence evaluation unit. A parallelized version of IPA using OpenMP meta-programming framework was also proposed, which fully utilizes multi-core CPU resources. Empirical results reveal that it can solve the influence maximization problem with competitive processing time and less memory usage. The complexity of computing the seeds is $O(|V||O_v|n_{vu})+O(\frac{|O_v \cup \{v\}|}{c} n_{vu}+ c)$, where O_v is the influence area of node v, n_{vu} is the average number of influence paths between v and u, and c is the number of execution units (i.e., CPU cores).

4.2 Heuristic Algorithms

As the greedy algorithms are time consuming, a series of heuristic algorithms which save much time have been proposed recently. Instead of computing the

marginal gain of nodes in each iteration, these heuristic algorithms iteratively select nodes based on a specific heuristic, such as *degree*, *PageRank*[3]. However, the result generated in these approaches is not satisfactory. Researchers have investigated several other heuristics that may improve the result quality of the algorithm.

DegreeDiscountIC. The running time of the aforementioned greedy approaches is still long and may not be suitable for very large social networks. Hence, Chen *et al.*[44] used *degree discount* heuristic, where each neighbor of newly selected seed discounts its degree by one, to reduce the running time. Although this heuristic can be used for all cascade models, they enhanced the degree discount heuristic to make it suitable for the independent cascade model. They demonstrated that the heuristic-based approach is orders of magnitude faster than all greedy algorithms. However, the seed set quality can be inferior, compared with the greedy approaches. Recently, they proposed a new heuristic approach[49] which introduces a parameter to control the balance between the result quality and the running time. However, the result quality is much lower than that of greedy approach although it has improved much over *degree discount*. It is worth mentioning that although the importance of reduction in computation time is undeniable, the seed set quality is more significant to companies as ultimately they would like to maximize the influence spreads of their new products in order to reach out to the largest possible customers.

MSA and SASH (*Simulated Annealing*). Jiang *et al.*[48] proposed an algorithm based on *simulated annealing* (SA) for the influence maximization problem. It is the first SA-based algorithm for the problem. Additionally, two heuristic methods were proposed to accelerate the convergence process of the algorithm. The algorithm is developed specifically for IC model, and it initiates the seeds set by randomly selecting k nodes. In each iteration afterwards, a node in the current seed set is replaced by another one which is not in the seeds, and thus a new seed set is formed. If the new seed set can generate better influence spread than the old one under the IC model, the seed set is updated to the new one. This process is iterated for T times until no such new seed set can be found. The time complexity of MSA is $O(Tk\overline{d})$ where $\overline{d}$ denotes the average degree of nodes. Experimental results have shown that the two heuristic methods, MSA and SASH, generate a better result quality compared with *degree discount* algorithm. The running time of MSA and SASH is similar to that of *degree discount*. However, the improvement in result quality is limited (i.e., 3% to 8%).

PMIA. Chen *et al.* proposed PMIA technique[49] over the IC model, which selects a limited number of paths that satisfy a given threshold θ to compute the influence. This threshold is used to tune the trade-off between influence spread and running time. The authors compared CELF, PMIA and degree discount-based approaches and demonstrated that PMIA improved the influence spread generated by *degree discount* by 3.9%∼6.6% over *Hep* dataset. However, the running time of PMIA is an order of magnitude slower than the degree discount-based technique with the time complexity of $O(nt_{i\theta} + kn_{o\theta}n_{i\theta}(n_{i\theta} + \log n))$ where $t_{i\theta}, n_{i\theta}, n_{o\theta}$ are constants decided by θ.

LDAG and SimPath. Based on the concept of PMIA model, another model called LDAG[51] was developed. It is similar to PMIA except that LDAG is specifically designed for the *linear threshold* model. LDAG also performs slightly better than *degree discount* over the *Hep* dataset in terms of influence spread. More importantly, the authors demonstrated that CELF still outperforms all these three heuristic-based approaches with respect to the quality of influence spread. In order to improve LDAG further, another alternative model called SimPath is developed[52]. Similar to LDAG, it is also designed for selecting influential seeds under linear threshold. However, unlike LDAG, SimPath computes the spread by exploring simple paths in the neighborhood. Using a parameter η, the trade-off between running time and seeds quality can be controlled. It has been shown by experimental results that by selecting proper parameter η, SimPath outperforms LDAG in both efficiency and accuracy. Both LDAG and SimPath are algorithms designed specifically for linear threshold model.

MIA-N. The aforementioned approaches investigate different ways to maximize the influence under the IC, WC and LT models. However, all of the existing models ignore an important aspect of influence propagation. Not only may positive opinions on products and services that we receive propagate through the network, but also negative opinions propagate. To this end, Chen *et al.*[50] proposed a novel model called IC-N (independent cascade model with negative opinions) which introduces a quality factor to control the negative opinion propagation probability. In order to maximize the influence under the IC-N model, a new heuristic algorithm called MIA-N, which borrows the core idea of PMIA, was developed. It defines a maximum influence in-arborescence to estimate the influence to an arbitrary

node v from other nodes. The time complexity of MIA-N algorithm is $O(nt_o + kn_on_i(\min(k, h_{\max})n_i + \log n))$ where $h_{\max}$ denotes the maximum height of the maximum influence in-arborescence. Experimental results have shown that the heuristic algorithm generates influence spread very close to the greedy method while is orders of magnitude faster than the greedy approach. Although IC-N incorporates negative opinions in networks, it suffers from another limitation. It assumes each node has the same influence and consequently exhibits the same quality factor that negative opinions can emerge. However, in real social networks, individuals may exhibit different probabilities to express opposite opinions. Moreover, individuals may show different influence in different topics. In fact, the quality factor proposed in IC-N model to control the negative opinion propagation probability can be viewed as a special case of conformity where an individual negatively follows another. However, the quality factor does not completely capture the negative opinions' propagation in that it assumes individuals show the same quality factor.

Unlike greedy algorithms, the quality of influence spread of these models is not guaranteed to be within 63% of the optimal. Although the aforementioned heuristic algorithms save much more time than greedy algorithms and improve the result quality compared to *degree* and *PageRank* heuristics, the results generated from these heuristic algorithms are poorer than those from the greedy algorithms.

Existing work in influence maximization suffers from either the running time (e.g., greedy approaches[3,13,44]), or the result quality (e.g., heuristic approaches[44,49]). Besides, there are two other directions in influence maximization research. Firstly, IC, WC and LT as well as the other existing models are synthetic ones and may not be suitable for real network propagations. Hence, learning the cascade models from real networks and proposing influence maximization solutions over the learned cascade model may be more realistic to real applications. Secondly, existing studies all assume that there is only one party who is maximizing their influence in a given network. However, in real case, there may be two or more parties maximizing their influence simultaneously, and how to solve the influence maximization problem in a competitive network where there are at least two campaigns within the same network is another problem that needs to be solved.

5 Conclusions

In this paper, we investigated the state-of-the-art research over social influence study within online social networks from three levels, individual-level, community-level, and network-level. We also carried out systematical comparison over the discussed approaches within the study of each corresponding level. Through our study in this paper, a series of potential applications as well as future research directions have been unveiled.

• In the field of individual-level study, state-of-the-art work all aims to find the nodes having the most influence on others. Instead, there is limited work in exploring the individuals who are easily influenced by others. In fact, using a group of network features and social characteristics, we can employ statistical or learning model to evaluate the probability of being influenced by others at the aspect of both local network topology and social characteristics. Taking into account the evolution of networks, if we have enough knowledge of users who are more probable to accept the opinion of others, it may make much sense in the design of next generation recommendation system.

• In the field of community-level study, existing work has investigated ways to detect and evaluate the change of communities. However, how to evaluate and rank the communities' ability of growing has not been answered yet. Moreover, how the community evolution is affected by each individual inside or outside it, and how a community evolves according to the network topological change, are unexplored yet. In another word, how the social influence at the individual level and the network level affects the evolution of communities is an important research direction that may draw much attention in next few years.

• In the field of network-level study, existing work has found different evolution phenomena in networks and tried to utilize these phenomena in information propagation. The most important and popular problem within this field is the influence maximization problem. Many algorithms have been proposed to address this problem. However, existing greedy algorithms have a series of limitations as follows. Firstly, greedy approaches suffer from running time, and it may take days for these algorithms to select 100 seeds from a network with hundreds of thousands nodes. Secondly, heuristic approaches suffer from the seed quality. Although these algorithms are one or two magnitudes faster than greedy ones, they do not guarantee the results quality. Recent efforts in influence maximization all focus

on leveraging the gap between the two aforementioned groups of approaches such that they can find a proper balance between the running time and the results quality. Thirdly, existing studies are based on a series of abstract mathematical models (e.g., the IC, WC, LT models). However, whether influence propagation follows these models is unjustified. For real applications, it may be more accurate if we can learn cascade models from real network propagation historical logs and then propose IM algorithms based on the historical logs. Hence, learning cascade models from real network and proposing real data-driven IM algorithms are another two research directions within this field. Last but not the least, existing studies all assume that there is only one party which is maximizing their influence in a network. However, in real applications, especially in viral marketing, there may be two or more companies trying to maximize their own influence within the same network, simultaneously or non-simultaneously. In these networks, existing IM algorithms are not suitable, as the influence propagation in competitive networks may not be the same with scenarios of non-competitive networks. Hence, solving the influence maximization problem in competitive networks is a big challenge. Moreover, solutions to this scenario can benefit a series of applications in real case.

References

[1] Brown J, Reingen P. Social ties and word-of-mouth referral behavior. *The Journal of Consumer Research*, 1987, 14(3): 350-362.

[2] Nemhauser G L, Wolsey L A, Fisher M L. An analysis of approximations for maximizing submodular set functions. *Math. Programming*, 1978, 14(1): 265-294.

[3] Kempe D, Kleinberg J, Tardos É. Maximizing the spread of influence through a social network. In *Proc. the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Aug. 2003, pp. 137-146.

[4] Cha M, Antonio J, Pérez N, Haddadi H. Flash floods and ripples: The spread of media content through the blogosphere. In *Proc. the 3rd International AAAI Conference on Weblogs and Social Media*, May 2009, pp. 8-15.

[5] Herlocker J L, Konstan J A, Terveen L G, Riedl J T. Evaluating collaborative filtering recommender systems. *ACM Trans. Inf. Syst.*, 2004, 22(1): 5-53.

[6] Koren Y. Factor in the neighbors: Scalable and accurate collaborative filtering. *ACM Trans. Knowl. Discov. Data*, 2010, 4(1): Article No. 1.

[7] Leskovec J, Adamic L A, Huberman B A. The dynamics of viral marketing. In *Proc. the 7th ACM Conference on Electronic Commerce*, Jun. 2006, pp. 228-237.

[8] Liu Y, Huang X, An A, Yu X. ARSA: A sentiment-aware model for predicting sales performance using blogs. In *Proc. the 30th International ACM SIGIR Conference on Research and Development in Information Retrieval*, July 2007, pp. 607-614.

[9] Rogers E M. Diffusion of Innovations (5th edition). New York: Free Press, 2003.

[10] Bakshy E, Hofman J M, Mason W A, Watts D J. Everyone's an influencer: Quantifying influence on twitter. In *Proc. the 4th ACM International Conference on Web Search and Data Mining*, Feb. 2011, pp 65-74.

[11] Bao H, Chang E Y. AdHeat: An influence-based diffusion model for propagating hints to match ads. In *Proc. the 19th International Conference on World Wide Web*, Apr. 2010, pp. 71-80.

[12] Borgatti SP. The key player problem. In *Dynamic Social Network Modeling and Analysis: 2002 Workshop Summary and Papers*, Breiger R, Carley K M, Pattison P(eds.), Washington, DC: National Academies Press, 2004, pp.241-252.

[13] Leskovec J, Krause A, Guestrin C, Faloutsos C, VanBriesen J M, Glance N S. Cost-effective outbreak detection in networks. In *Proc. the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Aug. 2007, pp. 420-429.

[14] Agarwal N, Liu H, Tang L, Yu P S. Identifying the influential bloggers in a community. In *Proc. the 1st ACM International Conference on Web Search and Data Mining*, Feb. 2008, pp. 207-218.

[15] Ma H, Yang H, Lyu M R, King I. Mining social networks using heat diffusion processes for marketing candidates selection. In *Proc. the 17th ACM International Conference on Information and Knowledge Management*, Oct. 2008, pp. 233-242.

[16] Pal A, Counts S. Identifying topical authorities in microblogs. In *Proc. the 4th ACM International Conference on Web Search and Data Mining*, Feb. 2011, pp. 45-54.

[17] Li H, Bhowmick S S, Sun A. Casino: Towards conformity-aware social influence analysis in online social networks. In *Proc. the 20th ACM International Conference on Information and Knowledge Management*, Oct. 2011, pp. 1007-1012.

[18] Liben-Nowell D, Kleinberg J. The link prediction problem for social networks. In *Proc. the 12th ACM International Conference on Information and Knowledge Management*, Nov. 2003, pp. 556-559.

[19] Newman M E. Clustering and preferential attachment in growing networks. *Physical Review E*, 2001, 64(2): 025102.

[20] Albert R, Barabási A L. Topology of evolving networks: Local events and universality. *Physical Review Letters*, 2000, 85(24): 5234-5237.

[21] Barabási A L, Albert R. Emergence of scaling in random networks. *Science*, 1999, 286(5439): 509-512.

[22] O'Madadhain J, Hutchins J, Smyth P. Prediction and ranking algorithms for event-based network data. *ACM SIGKDD Explorations Newsletter*, 2005, 7(2): 23-30.

[23] Leskovec J, Huttenlocher D, Kleinberg J. Predicting positive and negative links in online social networks. In *Proc. the 19th International Conference on World Wide Web*, Apr. 2010, pp. 641-650.

[24] Cai K, Bao S, Yang Z, Tang J, Ma R, Zhang L, Su Z. OOLAM: An opinion oriented link analysis model for influence persona discovery. In *Proc. the 4th ACM International Conference on Web Search and Data Mining*, Feb. 2011, pp. 645-654.

[25] Li H, Bhowmick S S, Sun A, Cui J. Affinity-driven blog cascade analysis and prediction. *Data Mining and Knowledge Discovery*, 2014, 28(2): 442-474.

[26] Li H, Bhowmick S S, Sun A. Blog cascade affinity: Analysis and prediction. In *Proc. the 18th ACM International Conference on Information and Knowledge Management*, Nov. 2009, pp. 1117-1126.

[27] Chin A, Chignell M H. A social hypertext model for finding community in blogs. In *Proc. the 17th ACM Conference on Hypertext and Hypermedia*, Aug. 2006, pp. 11-22.

[28] Xu X, Yuruk N, Feng Z, Schweiger T A J. SCAN: A structural clustering algorithm for networks. In *Proc. the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Aug. 2007, pp. 824-833.

[29] Jo Y, Lagoze C, Giles C L. Detecting research topics via the correlation between graphs and texts. In *Proc. the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Aug. 2007, pp. 370-379.

[30] Kumar R, Novak J, Raghavan P, Tomkins A. On the bursty evolution of blogspace. In *Proc. the 12th International Conference on World Wide Web*, May 2003, pp. 568-576.

[31] Backstrom L, Huttenlocher D, Kleinberg J, Lan X. Group formation in large social networks: Membership, growth, and evolution. In *Proc. the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Aug. 2006, pp. 44-54.

[32] Lin Y, Chi Y, Zhu S, Sundaram H, Tseng B L. Analyzing communities and their evolutions in dynamic social networks. *ACM Trans. Knowl. Discov. Data*, 2009, 3(2): Article No. 8.

[33] Lin Y, Sundaram H, Chi Y, Tatemura J, Tseng B L. Blog community discovery and evolution based on mutual awareness expansion. In *Proc. IEEE/WIC/ACM International Conference on Web Intelligence*, Nov. 2007, pp. 48-56.

[34] Sun J, Faloutsos C, Papadimitriou S, Yu P S. Graphscope: Parameter-free mining of large time-evolving graphs. In *Proc. the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Aug. 2007, pp. 687-696.

[35] Spiliopoulou M, Ntoutsi I, Theodoridis Y, Schult R. Monic: Modeling and monitoring cluster transitions. In *Proc. the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Aug. 2006, pp. 706-711.

[36] Bansal N, Chiang F, Koudas N, Tompa F W. Seeking stable clusters in the blogosphere. In *Proc. the 33rd International Conference on Very Large Data Bases*, Sept. 2007, pp. 806-817.

[37] Asur S, Parthasarathy S, Ucar D. An event-based framework for characterizing the evolutionary behavior of interaction graphs. In *Proc. the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Aug. 2007, pp. 913-921.

[38] Asur S, Parthasarathy S, Ucar D. An event-based framework for characterizing the evolutionary behavior of interaction graphs. *ACM Trans. Knowl. Discov. Data*, 2009, 3(4): Article No. 16.

[39] Li H, Bhowmick S S, Sun A. Affinity-driven prediction and ranking of products in online product review sites. In *Proc. the 19th ACM International Conference on Information and Knowledge Management*, Oct. 2010, pp. 1745-1748.

[40] Li H, Bhowmick S S, Sun A. Affrank: Affinity-driven ranking of products in online social rating networks. *Journal of the American Society for Information Science and Technology*, 2011, 62(7): 1345–1359.

[41] Leskovec J, Adamic L A, Huberman B A. The dynamics of viral marketing. *ACM Trans. Web*, 2007, 1(1): Article No. 5.

[42] Cha M, Mislove A, Gummadi K P. A measurement-driven analysis of information propagation in the flickr social network. In *Proc. the 18th International Conference on World Wide Web*, April 2009, pp. 721-730.

[43] Iribarren J L, Moro E. Information diffusion epidemics in social networks. *CoRR*, 2007. http://arxiv.org/abs/0706.0641, Dec. 2014.

[44] Chen W, Wang Y, Yang S. Efficient influence maximization in social networks. In *Proc. the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Jun. 2009, pp. 199-208.

[45] Wang Y, Cong G, Song G, Xie K. Community-based greedy algorithm for mining top-k influential nodes in mobile social networks. In *Proc. the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Jul. 2010, pp. 1039-1048.

[46] Li H, Bhowmick S S, Sun A. Cinema: Conformity-aware greedy algorithm for influence maximization in online social networks. In *Proc. the 16th International Conference on Extending Database Technology*, Mar. 2013, pp. 323-334.

[47] Kim J, Kim S, Yu H. Scalable and parallelizable processing of influence maximization for large-scale social networks? In *Proc. the 29th IEEE International Conference on Data Engineering*, Apr. 2013, pp. 266-277.

[48] Jiang Q, Song G, Cong G, Wang Y, Si W, Xie K. Simulated annealing based influence maximization in social networks. In *Proc. the 25th AAAI Conference on Artificial Intelligence*, Aug. 2011, pp. 127-132.

[49] Chen W, Wang C, Wang Y. Scalable influence maximization for prevalent viral marketing in large-scale social networks. In *Proc. the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Jul. 2010, pp. 1029-1038.

[50] Chen W, Collins A, Cummings R, Ke T, Liu Z, Rincón D, Sun X, Wang Y, Wei W, Yuan Y. Influence maximization in social networks when negative opinions may emerge and propagate. In *Proc. the 11th SIAM International Conference on Data Mining*, Apr. 2011, pp. 379-390.

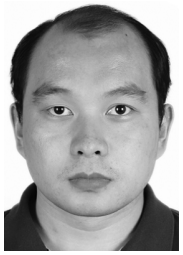
[51] Chen W, Yuan Y, Zhang L. Scalable influence maximization in social networks under the linear threshold model. In *Proc. the 10th IEEE International Conference on Data Mining*, Dec. 2010, pp. 88-97.

[52] Goyal A, Lu W, Lakshmanan L V S. Simpath: An efficient algorithm for influence maximization under the linear threshold model. In *Proc. the 11th IEEE International Conference on Data Mining*, Dec. 2011, pp. 211-220.

[53] Domingos P, Richardson M. Mining the network value of customers. In *Proc. the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Aug. 2001, pp. 57-66.

[54] Richardson M, Domingos P. Mining knowledge-sharing sites for viral marketing. In *Proc. the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Jul. 2002, pp. 61-70.

[55] Nemhauser G L, Wolsey L A. Maximizing submodular set functions: Formulations and analysis of algorithms. *Studies on Graphs and Discrete Programming*, 1981, 11: 279-301.
[56] Iwata S. A fully combinatorial algorithm for submodular function minimization. In *Proc. the 13th ACM-SIAM Symposium on Discrete Algorithms*, Jan. 2002, pp. 915-919.
[57] Iwata S, Orlin J B. A simple combinatorial algorithm for submodular function minimization. In *Proc. the 20th ACM-SIAM Symposium on Discrete Algorithms*, Jan. 2009, pp. 1230-1237.
[58] Feige U, Mirrokni V S, Vondrak J. Maximizing non-monotone submodular functions. In *Proc. the 48th Annual IEEE Symposium on Foundations of Computer Science*, Oct. 2007, pp. 461-471.
[59] Goyal A, Bonchi F, Lakshmanan L V S. A data-based approach to social influence maximization. *Proc. the VLDB Endowment*, 2011, 5(1): 73-84.



Hui Li received his B.E. degree from Harbin Institute of Technology in 2005 and Ph.D. degree in computer engineering from Nanyang Technological University, Singapore, in July 2012. He is an Associate Professor in the School of Cyber Engineering, Xidian University, Xi'an. His research interests include data mining, knowledge management and discovery, privacy-preserving query and analysis in big data. He is a member of CCF and ACM.

Jiang-Tao Cui received his B.E. and Ph.D. degrees in computer science from Xidian University in 1998 and 2005, respectively. He is a professor with the School of Cyber Engineering, Xidian University, Xi'an. His research interests are mainly on high-dimensional database management, multimedia content analysis, and cloud computing. He is a member of CCF and ACM.

Jian-Feng Ma received his M.E. and Ph.D. degrees in computer software and communications engineering from Xidian University, Xi'an, in 1988 and 1995, respectively. He is a member of the executive council of the Chinese Cryptology Society, and a Yangtze River Scholar Especially Hires Professor of the Ministry of Education of China. His research interests include information security, coding theory, and cryptography. He is a member of CCF and IEEE.