Maximizing the Impact of Competitive Relation Based on Node diffusion Model

Huanhuan Zhi¹, Yufeng Liu¹ and Lei Zhu¹

¹ College of Information Science and Engineering, Hunan University, 410082, Changsha, China

² Huanhuan Zhi: hhzh1229@163.com

Abstract. The problem of maximizing influence is to select a small number of nodes (seed nodes) in social networks so that they could maximize the spread of influence. In our paper, we have two contributions. First, we consider the impact of competition among products on nodes based on the Susceptible-Infected (SI) information diffusion model and improve it into a Rival Susceptible-Infected (RSI) information diffusion model. Second, the probability of influence between nodes is different based on different products under the same theme with competitive relations, and in the process of the node spreading one of the products, other products under the same theme have a competitive blocking effect on the node. So a new heuristic algorithm—Influence Reconstitution Algorithm (IRA)—is proposed to consider the impact of product competition on the nodes and the distance between the initial nodes. Our algorithm introduces the k-order core competition influence and coincidence rate P. Through the coincidence rate P, the influence of the initial node is reasonably controlled, and the most influential nodes are found in order. Our experiments based on real data sets shows that the IRA algorithm has better competitive effect than the existing heuristic algorithm in the competitive relational social network.

1. Introduction

In recent years, large-scale social networks (such as Facebook, Sina Weibo, Renren) have developed rapidly. This provides opportunities for "word-of-mouth" and "viral marketing", and a key question arises as to how to mine some nodes from large-scale social networks and maximizes the influence of these nodes in the network [1-4].

In social networks, it is necessary to use the corresponding propagation model to determine the influence of the initial node. Independent Cascade Model (IC) and Linear Threshold Model (LT) are the most basic propagation models. In addition, there are SI, SIR, SIS and other general information diffusion models in the process of social network dissemination. A. Gopalan et al.[5] proposed that the SI information diffusion model was widely used in social networks, and the effect was relatively good. Zhang et al.[6] analyzed the structure of social networks and classified the nodes according to the degree of nodes, and proposed an improved SI information diffusion model. Xu et al.[7] then proposed the SIS information diffusion model. Hacid et al.[8] based on the SIR information diffusion model and the SIS information diffusion model to predict the propagation of influence.

The above diffusion models only considered the spread of one product and ignored the simultaneous spread of multiple products. In our paper, the influence of competition among multiple products on nodes is considered, and the hindrance effect between competing products is proposed. We improve the SI information diffusion model into a RSI information diffusion model.
The study of the maximization of social network nodes influence has a long history. Kemple et al. proposed the greedy hill-climbing algorithm[9], which affected the results to reach 63% of the optimal solution. However, this algorithm had a large amount of computation and high time complexity. A. Goyal[10] and others proposed an effect maximization algorithm Celf ++ after improving the greedy algorithm. Chen et al.[11] proposed a heuristic algorithm Degree Discount. When a node was selected as the initial node, then a certain discount was given when the author calculated the degree of the neighbor nodes of this node. Kitsak et al.[12] proposed a cover-based maximum kernel algorithm and maximum degree algorithm. Estevez et al.[13] proposed the SCG (Slow Crack Growth) algorithm to select a large initial nodes. Considering that the initial nodes with a large degree have duplicate neighbor nodes, the selected initial nodes need to be dispersed as much as possible. Cao Jiuxin et al.[14] proposed a core coverage algorithm based on k-core and used a distance factor d to control the initial node distance. The IRIE algorithm[15] calculated the global influence ranking of each node mainly through the influence ranking algorithm (Influence Ranking), and then selected the influence nodes according to the ranking order. After selecting a seed node, the impact evaluation algorithm (Influence Estimation) was used to estimate the influence of the remaining nodes and then update the ranking of the node influence.

The algorithms mentioned above do not take into account the effect of the propagated product itself on nodes. Our paper considers that there is competition among different products of the same topic in social networks, and the competition relationship between products has a certain influence on the spread of products. Therefore, products with competitive relationships will have different degrees of impact on the nodes during the process of dissemination. In addition to considering the reasonable distance between the initial nodes, an effective heuristic algorithm IRA is proposed.

2. SI Information Diffusion Model

The SI information diffusion model [16] is one of the most important social network information diffusion models. In this model, there are only two states for each node. They are the effect state S(i) and the unaffected state I(i), and the node can only change from the unaffected state to the influence state. The basic idea of the SI propagation model is as follows: First, we obtain the influence state of the node at time t-1, and then calculate the influence probability of all neighboring nodes of the node by the influence state of the neighbor and the influence probability p of the neighbor, and finally obtain the node state at time t.

3. RSI Information Diffusion Model

3.1 Competition between nodes

In a real social network, there are competing relationships between different products of the same topic in the process of communication, such as mobile phone products, Apple, Huawei and Xiaomi compete with each other. And the same node can only be affected by one of the products. Assuming that there are M products that have a competitive relationship, \( M = \{1,2,3,...,m\} \). When a product is affected, other products have a competitive blocking effect on the same node, so the probability of influence between nodes will change due to different products. Figure 1 shows, in social networks \( G = (V,E) \), when there are three products, the impact of the three products based on nodes i and j is \( p_{i,j}^1, p_{i,j}^2 \) and \( p_{i,j}^3 \) respectively.

Based on the SI information diffusion model, this paper considers the impact of the competition relationship between products on the nodes, and proposes the RSI information diffusion model.
3.2 Nodes Influence Process

In a competitive social network, we introduce a competitive damping factor $\alpha' \ (0 \leq \alpha' \leq 1)$ to indicate the impedimental impact of a competing product on the node. The probability of influence between nodes is denoted by $p_{ij}$, $i \in V$ denotes the set of neighbor nodes of node $i$. We assume that there are $M$ products that have a competitive relationship, $M = \{1, 2, 3, ..., m\}$. Then, the probability of influence between nodes based on the product is calculated as follows:

$$ p_{ij} = p_{ij} - \left( \sum_{m \neq \alpha} \alpha' p_{ij} \right) \quad (1) $$

The following describes the process of affecting nodes in the network based on product $m$. In this paper, $p_{ij}$ is used to represent the probability that nodes are affected at time $t$. $p_{ij}$ is mainly related to three factors: 1) The probability that the neighbor nodes of this node will affect it at time $t-1$ to $t$. 2) The probability of influence of this node at time $t-1$. 3) The affected state of the node's neighbor nodes.

We denote $\chi_{i,t}^m$ as the probability that node $i$ is not affected by neighbor node $j$ that has been affected at time $t-1$ to $t$:

$$ \chi_{i,t}^m = ((1 - p_{ij}) p_{ij}) \quad (2) $$

The probability that neighbor node $j$ does not affect at time $t-1$ is $p_{ij}$. We define $\delta_{i,t}^m$ to represent the probability that node $i$ is not affected by neighbor node $j$ at time $t-1$ to $t$:

$$ \delta_{i,t}^m = ((1 - p_{ij}) p_{ij} + (1 - p_{ij})) \quad (3) $$

Node $i$ may have multiple neighbor nodes, so the probability that all neighbor nodes $j \in \mathbb{Z}$ fail to influence node $i$ from time $t-1$ to time $t$ is expressed as $\beta_{i,t}^m$:

$$ \beta_{i,t}^m = \prod_{j \in \mathbb{Z}} \delta_{i,j}^m = \prod_{j \in \mathbb{Z}} (1 - p_{ij} + (1 - p_{ij})) \quad (4) $$

In contrast, we use $\phi_{i,t}^m$ to express the probability that node $i$ is affected by its neighbors at time $t-1$ to $t$:

$$ \phi_{i,t}^m = (1 - \beta_{i,t}^m) = (1 - \prod_{j \in \mathbb{Z}} (1 - p_{ij} + (1 - p_{ij}))) \quad (5) $$

The probability that node $i$ is not affected at time $t-1$ is $1 - \phi_{i,t}^m$. The probability that node $i$ is not affected from time $t-1$ to time $t$ is $1 - \phi_{i,t}^m$. So the probability of a node being affected at time $t$ is:
\( P_{i}^{n} = 1 - (1 - P_{i}^{n})(1 - \phi_{i}) \)  
\( i \in V, m \in M \)  

(6)

Substituting formula (5) into formula (6) can get:

\[
P_{i}^{n} = 1 - (1 - P_{i}^{n})(1 - (1 - P_{j}^{n})) \]

\[
= 1 - (1 - P_{i}^{n})(\prod_{j \in Z}(1 - P_{j}^{n})) \]

\[
i \in V, m \in M \]

(7)

Decomposition of \( \prod_{j \in Z}(1 - P_{j}^{n}) \) gives:

\[
\prod_{j \in Z}(1 - P_{j}^{n}) = 1 - \sum_{j \in Z}(P_{i}^{n}P_{j}^{n}) \]

\[
+ \sum_{j, j' \in Z}(P_{i}^{n}P_{j}^{n}P_{j'}^{n}) \]

\[
- \sum_{j, j' \in Z}(P_{i}^{n}P_{j}^{n}P_{j}^{n}P_{j'}^{n}) \]

\[
i \in V, m \in M \]

(8)

The value following formula (8) is very small, which can be ignored and obtained:

\[
\prod_{j \in Z}(1 - P_{j}^{n}) = 1 - \sum_{j \in Z}(P_{i}^{n}) \]

\[
i \in V, m \in M \]

(9)

From equations (8) and (9), we get:

\[
P_{i}^{n} = 1 - (1 - P_{i}^{n})(1 - \sum_{j \in Z}(P_{i}^{n}P_{j}^{n})) \]

\[
= P_{i}^{n} + \sum_{j \in Z}(P_{i}^{n}P_{j}^{n}) - P_{i}^{n}\sum_{j \in Z}(P_{j}^{n}) \]

\[
i \in V, m \in M \]

(10)

Similarly, the value of \( \sum_{j \in Z}(P_{i}^{n}P_{j}^{n}) \) is very small, rounding off, simplified:

\[
P_{i}^{n} = P_{i}^{n} + \sum_{j \in Z}(P_{i}^{n}P_{j}^{n}) \]

\[
i \in V, m \in M \]

(11)

Where \( \sum_{j \in Z}(P_{i}^{n}P_{j}^{n}) \) represents the impact probability of all the neighboring nodes of node i. The edge between node i and its neighbor node j is denoted by \( b_{i,j} \). In a social network, if two nodes are neighbors, there is an edge \( b \) of 1, otherwise it is 0. We can get:

\[
P_{i}^{n} = P_{i}^{n} + \sum_{j \in Z}(P_{i}^{n}b_{i,j}P_{j}^{n}) \]

\[
i \in V, m \in M \]

(12)

Equation (12) shows the process of the probability of a node changing from \( t-1 \) to \( t \), and also reflects the change of the node's influence state.

3.3 RSI Model Framework

The vector \( P_{i}^{n} = [P_{i,1}^{n}, P_{i,2}^{n}, \ldots, P_{i,n}^{n}] \) is used to represent the influence probability of all nodes at time \( t \) during the propagation of product \( m \). \( P_{i}^{n} = 1 \) indicates that node i has been affected at time \( t \), and \( P_{i}^{n} = 0 \) indicates that node i has not been affected at time \( t \).

We use \( E_{i}^{n} \) to represent the probability of the influence of a node and its neighbors based on the propagation of a product, expressed as follows:
From the diffusion process, we can get:
\[
P_i^m = P_i^{m-1} + \left( E_i^m \right)^r
\]

As shown in Figure 2, \(m_1, m_2, m_3\) represents three different products, and the node has the influence probability of multiple products at the same time, but in the process of product competition, the same node can only be affected by one product. Therefore, at each time \(t\), it is judged whether \(P_i^m\) is equal to or greater than 1. When \(P_i^m > 1\), the node is affected by the product \(m\). If the same node meets multiple product influence conditions at the same time \(t\), then the node selects the product with the maximum value of \(P_i^m\).

Figure 2. The influence of different products existing at time \(t\)

Since the node is affected by a certain product, the maximum probability of affecting the state is only 1, so the control of the probability function is used.

\[
G(x_1, x_2, \ldots, x_n) = \min \{x_1, 1\}, \min \{x_2, 1\}, \ldots, \min \{x_n, 1\}
\]

Substituting equation (14) into equation (15) yields
\[
P_i^m = P_i^{m-1} + \left( E_i^m \right)^r
\]

Formula (16) mainly represents the change process of all nodes in the RSI competitive information diffusion model that are affected by competition. When \(t=0\), each product selects the corresponding initial nodes. Through any time \(t\), we can get the status of the nodes affected by the corresponding products in the entire network.

In addition to the corresponding propagation model, the most important issue is the selection of initial nodes. In the following section, we introduce an algorithm based on the k-order core competition influence to mine the initial node in order to maximize the impact.

4. K-order Competition Influence Algorithm

4.1 K-order Core Sets

**Definition 1:** (k-order core sets) In the social network topology of Figure \(G=(V, S)\), all k-order nodes of node \(i (i \in V)\) can be obtained according to the node cascading feature. The k-core core sets nodes are denoted by \(S^k(i) (i \in V)\).
A network node undirected graph is shown in figure 3. The first-order nodes of node 1 are 4, 8, and 21, the second-order nodes are 2, 5, 7, 11, 12, 13, 14, and 16, and the third-order nodes are 9, 15, 18. When k=2, we can get $S^2(i) = \{4, 8, 21, 2, 5, 7, 11, 12, 13, 14, 16\}$.

4.2 K-order Competition Influence

In a competitive social network, nodes that are spread based on different products have different probability $P_{ij}$. From this we can calculate the k-order influence of the core node based on k-order core nodes. The effect probability $D^n(i, j)$ based on a certain product between nodes is expressed as follows:

$$D^n(i, j) = P_{ij}^n \quad i \in V, \quad j \in Z_i$$

(17)

The impact probability of the core node on the k-th nodes are:

$$D^n(i, j_1, \ldots, j_k, s, q) = D^n(i, j_1) \cdot D^n(j_1, j_2) \cdot \ldots \cdot D^n(s, q) \quad i \in V, \quad j_1, \ldots, j_k \in Z_i, \quad s, q \in Z_s$$

(18)

The first-order influence $L^n(i)$ of the core node is expressed as follows:

$$L^n(i) = \sum_{j \in Z_i} P_{ij}^n$$

(19)

The k-order influence of this core nodes are calculated as follows:

$$I^n_k(i) = L^n(i)$$

$$I^n_k(i) = \sum_{j \in Z_i} D^n(i, j) \cdot L^n(j)$$

$$I^n_k(i) = \sum_{j \in Z_i, \ldots, s \in Z_s} D^n(i, j, s) \cdot L^n(s)$$

And so on

$$I^n_k(i) = \sum_{j \in Z_i, \ldots, s \in Z_s} D^n(i, j, s, q) \cdot L^n(q)$$

(20)

The selected initial nodes consider the distance between the initial nodes in addition to the size of the competition influence of the nodes.

**Definition 2** (coincidence rate) The coincidence rate is defined as the ratio of k-order common nodes sets of node i and node j to the k-order core sets of two nodes, denoted by $P$: 

$$P(i, j) = \frac{\left|\bigcup_{k=1}^{\infty} S^k(i) \bigcap S^k(j)\right|}{\bigcup_{k=1}^{\infty} S^k(i) \bigcup S^k(j)}$$

(\forall i \in S_i, j \in V$$

(21)

The range of coincidence rate controls the influence of the initial node, $P$ is too high, and it is easy to maximize local influence. $P$ is too low, and the connectivity of the edge nodes of the initial node k-level core sets is not easy to be affected. So this paper proposes a suitable method for calculating the coincidence rate as follows:

$$P(i, j) \in (\alpha, \beta) \quad \alpha, \beta \in (0,1)$$

(22)
\[
\alpha = \frac{S'(j)/2'}{(\bigcup_{s \in S} S'(s)) \cup S'(j)} \quad (i \in S, j \in V)
\]
\[
\beta = \left[\sqrt{\frac{\bigcup_{s \in S} S'(s) \cup S'(j)}{S'(j)}} \right] \quad (i \in S, j \in V)
\]

(23)

(24)

The IRA algorithm discussed in this paper is mainly selected based on the k-th order competition influence of nodes, and then determined by the coincidence rate. When k is 1, the new initial node obtained through the coincidence rate range is generally in the vicinity of the first-order node of the previous initial node. The resulting initial sets of nodes do not have enough diffusion, and then it may maximize the local influence, so k is usually selected from two.

4.3 IRA Algorithm

The basic idea of the IRA algorithm is briefly described as follows: 1) According to the k-order core sets, the node with the greatest impact on the core nodes is selected. 2) The selected node is marked and its first-order core set is marked to indicate that it can no longer be selected. 3) The coincidence rate range is judged, and the newly selected node is calculated for the coincidence rate with all selected initial nodes. The control of the coincidence rate provides multiple opportunities for the k-order core set edge nodes of each initial node to be affected.

\begin{algorithm}
\textbf{Algorithm Influence Reconstitution (G, r)}
1: Initialize $S = \emptyset, C_v = \emptyset$
2: for each vertex $i$ do
3: \hspace{1cm} compute $I^n_v(i)$
4: \hspace{1cm} end for
5: select $i = \arg \max \{ I^n_v(i) | i \in V \}$
6: $S = S \cup \{i\}$
7: $C_v = C_v \cup \{i, S'(i)\}$
8: for $t=1$ to $r-1$ do
9: \hspace{1cm} select $j = \arg \max \{ I^n_j(j) | j \in V \}$
10: \hspace{1cm} if $j \notin C_v$
11: \hspace{2cm} compute $p(j, i), \alpha, \beta$
12: \hspace{2cm} if $p(j, i) \in (\alpha, \beta)$
13: \hspace{3cm} $S = S \cup \{j\}$
14: \hspace{2cm} $C_v = C_v \cup \{i, S'(i)\}$
15: \hspace{2cm} $t=t+1$
16: \hspace{1cm} end if
17: \hspace{1cm} end if
18: end for
19: output $S$
\end{algorithm}

The IRA algorithm mainly compares the k-th order competition influence of each node, so the comparison time complexity is $O(n)$; The process of node marking is to mark only the first-order core set of the selected initial nodes, so the time complexity is less than $O(n)$; the comparison process of the coincidence rate and the range comparison is to compare only the selected initial nodes, and the time complexity is much less than $O(n)$.

5. Experiments

5.1 Datasets
we use real data sets provided by Stanford Large Network Datasets Collection. As shown in Table 1, Epinions [17] is a who-trust-whom online social network of a general consumer review site Epinions.com. Enron email [18] communication network covers all the email communication within a datasets of around half million emails.

| Network      | Nodes | Edges | Average clustering coefficient |
|--------------|-------|-------|-------------------------------|
| Epinions     | 75879 | 508837| 0.1378                        |
| Enron email  | 36692 | 183831| 0.4970                        |

5.2 Comparative Experiment
In order to verify IRA algorithm, we compared it with several other representative algorithms. As shown in Table 2. The Degree Discount algorithm proposed by Chen et al., which has a good measure of the influence of a node in the current neighborhood, has considerable performance in most experiments. This experimental parameter setting is from the paper [11]. The IRIE algorithm is a heuristic algorithm that is currently considered to be a better one in terms of running time, memory consumption, impact on propagation range, and the like. Experimental parameters are set from the paper [15]. The influence of CCA algorithm on the influence of force transmission is also good. The experimental parameters of the algorithm come from the paper [14]. In our algorithm, we set the empirical parameters \( k \) and \( \varepsilon \) to 3.

In a competitive social network, there will be multiple products on the same topic. This paper assumes that the competition of the three products of the same subject is represented by \( a \), \( b \), and \( c \) respectively. We set the corresponding competitive damping coefficients to 0.01, 0.02, and 0.03 to analyze and compare the effects of different algorithms on different products. In the experiment, the initial node is selected in turn according to the algorithm corresponding to the product, and the following algorithm should select the initial node that was not selected by the previous algorithm. The probability of Inter-node Influence Using the Weighted Cascading Model[1], \( p_{v,v'} = \frac{1}{d_v} \), where \( d_v \) is the in-degree of \( v \).

| Algorithm    | Algorithm Description                                                                 |
|--------------|---------------------------------------------------------------------------------------|
| Degree Discount[11] | The neighbor node of the initial node performs a discount heuristic algorithm, abbreviated as DD. |
| IRIE[15]     | A heuristic algorithm that combines influence ranking and influence assessment.        |
| CCA[14]      | Heuristic Algorithm Based on the Hierarchical Feature and Influence Radius of the Kernel. |
| IRA          | A heuristic algorithm for selecting different nodes based on \( k \)-order competitive influence and different products |

5.3 Analysis of Results
The evaluation of the experimental result in this paper can be summarized as the following two points: (1) Impact speed. The numbers of nodes affected by the same time. (2) The number of nodes that can ultimately be affected. In the nodal propagation effect quantity map, the x-axis represents the time step.
in the information propagation model, and the y-axis represents the numbers of influences of the node. We can observe the velocity of the nodes. All the legends represent different algorithm comparisons based on different product choices. As shown in Figure 4(a), DD-a indicates that a product is based on algorithm DD to select the initial node, and CCA-b indicates that b product is based on algorithm CCA to select the initial node. IRIE-c indicates that the c product is based on the initial node selected by the algorithm IRIE, and other representations are also so meaningful. In the final influence percentage graph of the node, we can see the final influence of the nodes of the corresponding algorithm. All the algorithms in this paper choose initial nodes m as 5 for comparison experiments.
Figure 4. Nodes influence change on Epinions
Figure 5. Nodes influence change on Enron email

Figure 6. Average percentage of the final influence on Epinions
We compare the four algorithms based on the competition process of three products. In fig.4 and fig.5, time $t$ is from 0 to 100, which is a time period where the influence of the node is relatively critical and is suitable for analyzing the influence speed of the algorithm. Fig. 6 and Fig. 7 are the percentages of the final influence of the algorithm, which can be used to analyze the final effect of these four algorithms based on $a$, $b$, and $c$. From the experimental results of two real data sets, when the algorithm based on a product and b product is the same, we compare the algorithm based on c product. From figure 4 and figure 5, it can be seen that the influence speed of IRA algorithm is better than other algorithms in most cases. Only in the comparison group of fig.4 (a-b) and fig.5 (a-b), it can be seen that the influence speed difference between CCA algorithm and IRA algorithm is not much. IRA algorithm is better than CCA algorithm in other comparison groups. This fully proves that considering the node's level node association characteristics is more important than the node's hierarchy. IRIE and DD also have a considerable speed of influence, but in the comparison experiment, they are still inferior to IRA algorithm. In Fig. 6 and Fig. 7, based on the competition of a product, the algorithm CCA effect is the better, indicating that the algorithm works better in the case of preferential selection. And based on the competition between b products and c products, the effect of IRA is relatively good. This also proves that the IRA algorithm selects the initial node based on the influence of different products, and it can really improve the competitive influence of the nodes. Although the algorithms IRIE and DD are effective in the general social network, they have relatively less impact in such competitive social networks. It indicates that the selected initial node is relatively uncompetitive. Therefore, in competitive social networks, it is necessary to consider different product-to-node impact factors.

**6. Conclusion**

In this paper, an IRA algorithm is proposed to maximize the influence of nodes in competitive social network, according to the competitive influence and coincidence rate of k-order core sets of nodes. In order to solve the problem that the SI information diffusion model does not consider the influence of competition among products, we improve it and get the RSI diffusion model. Experiments were carried out on the RSI information diffusion model and compared with other algorithms. The following conclusions were obtained:1) In terms of the influence speed of nodes, the influence speed of IRA nodes is better.2) In the final impact state of the competitive network, the competition effect of IRA algorithm is the best under three products.

This provides a reference for the initial node that should be selected for the maximum impact of the competitive product.

In the following research, for one thing, machine learning methods are used to more accurately predict the probability of influence between nodes. For another thing, the total number of data set nodes and the effect of k-order core set characteristics on k value are calculated.
References

[1] Chen W, Wang Y, Yang S. Efficient influence maximization in social networks[C]// ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 199-208(2009).

[2] Jiang B, Hegde N, Massoulie L, et al. How to Optimaly allocate your budget of attention in social networks[C]// INFOCOM, 2013 Proceedings IEEE. IEEE, 2373-2381(2013).

[3] Borgs C, Brautbar M, Chayes J, et al. Maximizing Social Influence in Nearly Optimal Time[J]. Computer Science, (2013).

[4] Ou H C, Chou C K, Chen M S. Influence Maximization for Complementary Goods:Why Parties Fail to Cooperate?[J]. 1713-1722(2016).

[5] Gopalan A, Banerjee S, Das A K, et al. Random mobility and the spread of infection[C]// IEEE INFOCOM. IEEE, 999-1007(2011).

[6] Zhang H, Dinh T N, Thai M T. Maximizing the Spread of Positive Influence in Online Social Networks[C]// IEEE, International Conference on Distributed Computing Systems. IEEE, 317-326(2013).

[7] Xu B, Liu L. Information diffusion through online social networks[C]// IEEE International Conference on Emergency Management and Management Sciences. IEEE, 53-56(2010).

[8] Guille A, Hacid H. A predictive model for the temporal dynamics of information diffusion in online social networks[C]// International Conference on World Wide Web. ACM, 1145-1152(2012).

[9] Kempe D, Kleinberg J. Maximizing the spread of influence through a social network[C]// ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 137-146(2003).

[10] Goyal A, Lu W, Lakshmanan L V S. CELF++: optimizing the greedy algorithm for influence maximization in social networks[C]// International Conference Companion on World Wide Web. ACM, 47-48(2011).

[11] Chen W, Wei L, Zhang N. Time-Critical Influence Maximization in Social Networks with Time-Delayed Diffusion Process[J]. 19(5):592-598(2012).

[12] Kitsak M, Gallos L K, Havlin S, et al. Identification of influential spreaders in complex networks[J]. Nature Physics, 6(11):888-893(2010).

[13] Estevez P A, Vera P, Saito K. Selecting the Most Influential Nodes in Social Networks[C]// International Joint Conference on Neural Networks. IEEE,2397-2402(2007).

[14] Cao Jiuxin, Dong Dan, Xu Shun, et al. A social network impact maximization algorithm based on k- kernel [J]. Journal of computer science,38 (2): 238-248(2015).

[15] Jung K, Heo W, Chen W. IRIE: Scalable and Robust Influence Maximization in Social Networks[C]// IEEE, International Conference on Data Mining. IEEE, 918-923(2013).

[16] Zhou,“Initial Spreaders in Online Social Networks,”[C]//IEEE, in Fifty-fourth Annual Allerton Conference. Allerton Conference, UIUC, Illinois, USA September 27 - 30, 180-186(2016).

[17] Richardson M, Agrawal R, Domingos P. Trust Management for the Semantic Web[M]// The Semantic Web - ISWC 2003. Springer Berlin Heidelberg.351-368(2003).

[18] J. Leskovec, K. Lang, A. Dasgupta, M. Mahoney. Community Structure in Large Networks: Natural Cluster Sizes and the Absence of Large Well-Defined Clusters. Internet Mathematics 6(1) 29--123, (2009).