Identifying Influential Users by Their Postings in Social Networks

Beiming Sun and Vincent TY Ng

Department of Computing,
Hong Kong Polytechnic University, Hong Kong
{csbsun,cstyng}@comp.polyu.edu.hk

Abstract. Much research effort has been conducted to analyze information of social networks, such as finding the influential users. Our aim is to identify the most influential users based on their interactions in posting on a given topic. We first proposes a graph model of online posts, which represents the relationships between online posts of one topic, so as to find the influential posts on the topic. Based on the influential posts found, the post graph is transformed to a user graph that can be used to discover influential users with improved influence measures. Finally the most influential users can be determined by considering the properties and measures from both graphs. In our work, two types of influences are defined based on two roles: starter and connecter. A starter is followed by many others, similar to a hub in a network; while a connecter is to help bridging two different starters and their corresponding clusters. In this paper, different measures on the graphs are introduced to calculate the influences on the two roles.

1 Introduction

With the rapid development and increased popularity of social networks, more and more interests have been made in obtaining information from social networking websites for analyzing people's behaviors. Our research is focusing on identifying the influential social network users; as it can help to increase the marketing efficiency, and also can be utilized to gather opinions and information on particular topics as well as to predict the trends. In order to find these influential users, the first problem is to measure a user's influence on social networks. In the past, there has been a lot of work on judging the influence of users on a specific social networking site. For example, many measurement metrics have made use of the relationship between users (i.e. follower / followee) in Twitter. However, they mostly ignore the interactions of users in their online posts. Moreover, without the consideration of the contents posted by users, they are not able to tell the influence of users on different topics.

In order to identify the influential users or leaders within a topic, we first obtain a measure of the influence of online posts on that topic. Next, we identify the most influential posts, and then based on their authors we further measure and compare the influence of users. In this paper, there are two types of influences based on the two roles: starter and connecter. A starter is followed by many others, similar to a hub in a network, so it should have certain influence. The connecter is also regarded to be

M. Atzmueller et al. (Eds.): MUSE/MSM 2012, LNAI 8329, pp. 128–151, 2013.

influential when it links starters together. Both types are considered as influential in online posts as well as users.

The approach is to first figure out the relationship between online posts. Usually, posts are considered to be related in a thread or a chain. However, their relationships can be more complicated in certain cases. For example, a post is replying to a previous post while its content refers to a different one. Other than these direct responses as explicit relationships, there is also implicit relationship between online posts. For example, a user has read a post online. Instead of directly replying to it, he writes a new post on this topic. In this scenario, the two posts are considered to be implicitly related, because the action of later posting is influenced by the earlier one [1, 2]. Considering these situations, we proposed a graph model to represent the relationships between online posts on a topic. With the information of the explicit and implicit relationships between posts, the model tries to identify the most influential posts and users based on their direct interactions as well as the underlying relationships on the same topic. Three measurement methods are used to assess the influences of posts and to identify starters and connecters. Based on the influential posts found, we transformed the post graph to the user graph, and then refined the influence measures of users acting as starter and connecter. Finally, the most influential users can be identified by considering the properties and measures from both graphs.

The rest of the paper is organized as follows. Section 2 reviews some related works and a graph model is defined in section 3 to represent the relationships between online posts. After that, three different methods of influence measure are proposed based on the graph model. Section 5 defines the user graph model. The next session presents the conversion from the post graph to user graph, and the measurements of user influences. Section 7 discusses the tests with different cases to verify our models. Finally, we summarize the paper and suggest for future work in the last section.

2 Related Work

Many methods have been proposed to measure users' influence on Twitter. A popular metric of influence is the number of a user's follower [3]. It makes the assumption that all followers will read the contents published by that user. Yet, this method ignores the different ways for users to interact with the online contents. There are also many online tools to measure a user's influence on social network, such as Klout Score [4] and Twinfluence [5]. However, they cannot tell the influences of users on different topics. In [6], the TwitterRank algorithm, which is an extension of PageRank, was proposed to measure the user influence on Twitter taking both the topical similarity between users and the link structure into account. TunkRank [21] is another adaptation of PageRank. It makes the assumption that if a user reads a tweet from his friend he will retweet it with a constant probability. The influence is calculated recursively considering the attention a user can give to his friends, and that their followers could give to them. These methods do not consider users' interaction in posting. Yet, it is interesting to judge their influences not by their relations of friends in static structure, but based on the dynamic interaction in online contents.

As for the work of role detection on social media, Hansen et al. defined the social role of discussion starters based on graph metrics [7]. Discussion starters mostly receive messages often from people who are well-connected to each other, and they can be identified by low in-degree, high out-degree and high clustering coefficient in the graph. This metric does not suit our model, because the clustering coefficient is better to deal with an undirected graph or a directed graph with loops. Mathioudakis and Koudas did similar work [8] in distinguishing starters and followers on blogs. The starter does not mean the first one to open the discussion but the one who triggers an intense discussion. They expected that a blogger, who primarily generates posts that others link (inlinks) over a significant period of time, could be a starter, and the bloggers who primarily generate posts that links to other blog posts (outlinks) would be followers. They compared which bloggers behave more as 'starters' by computing the difference between the number of inlinks and outlinks of their blogs. Their experiments showed that it is possible to identify the top starters for a given query of several topic keywords in BlogScope. In this paper, we adopt the definition of the role of starter. In addition, we also propose connecters that link starters together as they are influential too.

Specially, Shetty and Adibi proposed the Entropy model to identify the most important nodes in a graph [9]. They dealt with the problem of finding leaders in a network. They built the graph so that nodes are representing persons or organizations and edges are representing actions they are involved in. They determined the important nodes by those who have the most effect of the graph entropy when they are removed from the graph. They used the event based entropy that has been similarly defined in [10]. Their experiment showed that comparing to conventional techniques such as betweenness centrality, this method leads to a better result. More important nodes can be discovered based on their effect on graph entropy in the ordered network. However, the graph entropy model claims its results on certain assumptions, like the evidence data is complete and with no noise.

Inspired by their ideas, we propose a method of measuring the influence of online posts through a refined graph entropy approach. In addition, the methods of Degree Measure and Shortest-path Cost Measure are exploited and integrated their results to identify the most influential posts. The details are discussed in Section 4. After the influential posts are identified, their authors are considered as potential influential users whose influence will be finally determined in the user graph model. In Section 6, we describe how to build the user graph model based on the post graph, meanwhile how to measure the users' influences from three aspects.

3 Graph Model of Online Posts

A lot of research work has been carried out in using graph methods to analyze the relationships between users on certain social networking websites [11, 12, 13, 14]. Here, we propose a general model of online posts which can be applied in different social networking sites while the user information is also taken into consideration.

A graph is defined as $G_v(V, E_v)$, where V is the set of posts and E_v is the set of directed edges which represent the relationship between those posts. Each post $v \in V$ can be described as a tuple of the form (n, t, u, c) where n is the node type, t is the timestamp, u is the author of the post and c is its content. Each directed edge $e \in E_v$ can be represented as $(v_i, v_j, p, w_{i,j})$ where v_i, v_j are nodes and e is an edge directed from v_i to v_j which means v_i is related to v_j, p specifies the type of relationship (either explicit or implicit), and $w_{i,j}$ is the weight of edge in range of (0, 1] that measures the strength of their relationship. The relationship is directional and irreciprocal. It is defined that each post can only be related to (point to) earlier posts. Therefore, it is a directed acyclic graph and there should be of single edge connection between any two nodes as shown in Figure 1.

3.1 Types of Relationship

Explicit Relationship: It is given explicitly by the META information data collected from the social media platform, including the relationship of direct reply and some other forms (depending on the functions provided by the social network media, such as "share" on Facebook, "retweet" on Twitter and "citation" on forums). A relevance score $r_{i,j}$, which will be discussed later, is assigned to each edge from v_i to v_j, in order to calculate the edge weight. The score is set to 1 for all explicit relationships to represent full relevance. For example, $r_{i,j} = 1$ if v_i is a reply or retweet to v_j on Twitter.

Implicit Relationship: It is used to connect posts that are not directly related but talking on the same topic. The implicit relationship, $r_{i,j}$, indicates the degree of content relevancy from v_i to v_j that can be determined by measuring the content similarity score. The score should be in the range of (0, 1]. The conditions of building an implicit relationship can be different and depending on the features of the social networks applied on. In general, it is restricted by the time interval between two posts, as their relationship should weaken or dissolve when the time interval exceeds a certain time (called expiration time). For some forums in which only members within a group can see the posts of each other, the user's identity is also a restriction. For the blogging sites such as Facebook and Twitter, where one's posts can only be seen by friends or followers, the building of implicit relationships of posts is limited to their authors' friendship network.

3.2 Types of Posts

The type of a post is determined by the role it plays. The posts can be characterized in four types: root, follower, starter and connecter. Among them starters are certainly considered to be influential. Many researchers have tried to identify starters in a network as stated before. As for connecters, they are considered as bridges that connect two or more peaks in centrality analysis [15]. Also, a bridge node is also important in a network if it connects starters. Therefore, we define connecter to represent this type of nodes which are influential in a different way. Noted that in our definitions, follower, starter and connecter are referring to the type of posts.

Root: It is the first post discussing a topic or a subtopic within a certain period, so it is not related to any others. In the graph, roots are the nodes who are not pointing to others (with no out-degree).

Follower: It is a response (e.g. reply, comment or share) of a post or a new post talking on the same topic as another post before, which means it is explicitly or implicitly related to others. In a graph, followers are the nodes who are pointing to others (with some out-degree).

Starter: It is identified when it received a large number of explicit or implicit responses (followers); meanwhile the less it behaves as follower, the more it acts like a starter. In a graph, conversation starters are the nodes who point to a few but be pointed by many others, i.e., they are of high in-degree and low out-degree. Moreover, it is better for a starter to have followers also followed by many others. In a graph, it can be observed as having a high in-degree of followers.

Connecter: It connects two or more starters as a bridge, which means some starters will be disconnected without this node. It should be noticed that a post may play multi-roles at the same time. It is also possible that the roles of posts can change over time. The details of the identification of the node types will be discussed in Section 4.

3.3 Edge Weight

The weight assigned to each edge is the degree of relevance between two posts and high weight edges indicate strong relationships. Edge weight is measured by two factors: the content relevance and the time interval between posts.

$$w_{i,j} = \alpha_T \cdot r_{i,j} \tag{1}$$

An example is shown in Figure 1, where T is the time interval between the root post and its first reply. α_T is a factor used to diminish the relevance degree based on T. The details of calculating for content relevance between posts and edge weights have been introduced in our previous paper [22].

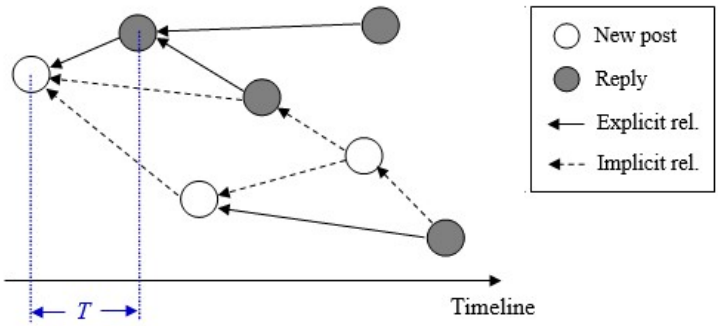

Fig. 1. A post graph with the timeline

4 Influence Measurements

4.1 Degree Measure

As mentioned above, the degree of a node can be used to identify starters. Since a starter is supposed to have a lot of followers and it is not a follower of many others, we first compute the difference between the in-degree and out-degree of each node. The in-degree of a node v denoted as $deg^+(v)$ is the sum of weight of the incoming edges incident to the node v, and the out-degree $deg^-(v)$ that is the sum of weight of its outgoing edges. The difference $d(v)$ is measured as one factor [8]:

$$d(v) = deg^+(v) - deg^-(v) \tag{2}$$

Another factor is the weighted average of its follower in-degrees to reflect the popularity of its followers:

$$\begin{aligned} s(v_i) &= \frac{\sum_{v_j \in Fol(v_i)} w_{i,j} \cdot deg^+(v_j)}{\sum_{v_j \in Fol(v_i)} w_{i,j}} \\ &= \frac{\sum_{v_j \in Fol(v_i)} w_{i,j} \cdot deg^+(v_j)}{deg^+(v_i)} \end{aligned} \tag{3}$$

Then we can identify a node $v_i \in V$ as a starter when both $d(v_i)$ and $s(v_i)$ reach a threshold:

$$d(v_i) \geq \sigma_1 \wedge s(v_i) \geq \sigma_2$$

4.2 Shortest-Path Cost Measure

The basic idea of this method is to judge a node's influence by measuring how many other nodes would be affected and how much the influences are if the target node is removed from the graph. It should be noted that in a graph the relationship edges are built from later posts to earlier ones; conversely the influences traverse in reverse directions from earlier posts to later ones.

In our definition, a post should have influence on its followers, as the followers are responses (e.g. replies, citation and share) that are somehow activated by the original post (followee). These followers may also have influence on their own followers. As a result, a post may have indirect influences on its followers' followers, and so on. In a graph $G(V, E)$, the descendant set $Des(v)$ of a node $v \in V$ includes its followers directly pointing to it and other descendants that can reach it through paths. For every $v_d \in Des(v)$, there is at least one directed path from v_d to v in the graph.

If the path from node v_d to v_n is $(v_d, v_{d+1}, v_{d+2}, \ldots, v_n)$, the *relationship strength* from v_d to v_n can be measured as the accumulative weight:

$$W(v_d, v_n) = \prod_{i=d}^{n-1} w_{i,i+1} \tag{4}$$

where v_i is pointing to v_{i+1} and $w_{i,i+1}$ is their edge weight. If more than one path from v_d to v_n exist, the maximum accumulative weight is taken as their relationship strength value. By doing this, the value of weight between any two nodes can be constrained in the range (0, 1]. The reason not to do summation and normalization of $w_{i,i+1}$ is that it will induce new weights with too small variance, which is difficult to differentiate afterwards. On the other hand, the ancestor set $Anc(v_d)$ of a node v_d is defined accordingly: $v_a \in Anc(v_d)$ when $v_d \in Des(v_a)$.

The algorithm of finding ancestors is similar to the one of finding the shortest path with respect to cost between nodes in a graph, except that we calculate the path cost as the product of the weights instead of the sum. It is assumed that each node would have influence on its descendants in the graph. To measure the influence of a node, we remove it from the graph and capture the change of path cost between these descendant nodes and their ancestors. The path cost $c(v_d)$ of a node v_d to its ancestors $v_a \in Anc(v_d)$ is the average of their relationship strength value:

$$c(v_d) = \frac{1}{\left|Anc(v_d)\right|} \sum_{v_a \in Anc(v_d)} W(v_d, v_a) \tag{5}$$

Here we take the average in order to reduce the benefit for the nodes in later time, because later posts may have more ancestors. When a node v_i is removed from the graph, its adjacent edges are also removed. Its descendants $v_d \in Des(v_i)$ may be disconnected from some of their original ancestors. Even if they can reach their ancestors through other paths, their relationship strength may be weakened if the removed node is on their shortest path. Suppose v_i is on the path with the shortest cost between v_d and its ancestor $v_a \in Anc(v_d)$. After v_i is removed, a new path should be found with the new relationship strength value that $W'(v_d, v_a) \leq W(v_d, v_a)$. If no path can be traced between v_d to v_a, it means v_d is disconnected from v_a, and their relationship strength will be set to 0 ($W'(v_d, v_a) = 0$). If v_i is not on that path, the relationship strength between v_d and v_a will not change: $W'(v_d, v_a) = W(v_d, v_a)$.

Let $C(v_d, G, v_i)$ be the average shortest-path cost between the node v_d and its ancestors after removing v_i from the graph G. The influence of a node $v_i \in V$, $Inf_c(v_i)$, in the graph is then:

$$Inf_c(v_i) = \sum_{v_d \in Des(v_i)} (C(v_d, G, \varnothing) - C(v_d, G, v_i)) \tag{6}$$

Compared to the degree measurement, this method considers multi-level relationship between posts, even if they are not on the same path. For example, as shown in Figure 2 (explicit relationship denoted by solid arrow and implicit relationship denoted by virtual arrow). Suppose node A is removed to see the influence on B and C. Then, B will be disconnected from any other nodes, while C can be still connected to D. Hence, A has a larger influence to B than to C. In this

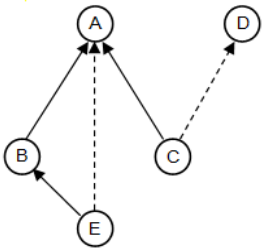

Fig. 2. An example graph of related posts

case, the influence measure of node A also considers the relationship between C and D, which is not considered in the degree measurement. Another advantage is the avoidance of duplicate counting on node E when measuring the influence of node A in multi-levels.

4.3 Graph Entropy Measure

Based on the graph model proposed, a graph can be considered as an ordered network with the node types of root, follower, starter and connecter defined. Shetty and Adibi [9] showed their success in finding important nodes through graph entropy in an ordered network. The graph entropy can be defined differently for various problems and we adopted a similar approach as in Dehme [18]. In their work, the entropy of a network is defined by using the local information graph, where metrical graph properties are used for defining information functional of each vertex.

Consider a graph with arbitrary node labels. In order to determine the probability value for each node so that it can be used to calculate the graph entropy, we first need to define the local vertex functional. Generally, the information functional is used to quantify structural information based on a given probability distribution. In our case, we define the information functional as the centrality of nodes.

For the graph $G = (V, E)$ where $v_i \in V$, graph entropy is defined by:

$$E(G, P) = \sum_{i=1}^{|V|} p(v_i) \log(1 / p(v_i)) \tag{7}$$

The probability for each node is defined as:

$$p(v_i) = \frac{f(v_i)}{\sum_{j=1}^{|V|} f(v_j)} \tag{8}$$

f represents an arbitrary information functional. Unlike traditional centrality measurement, such as closeness centrality, betweenness centrality and eigenvector centrality, in our model the centrality of a node only looks at the nodes that point to it or can be reached through paths. Recalling the term of “descendant” that is defined in last section, a node’s descendants is used to measure its distance-weighted centrality.

$$f(v_i) = \sum_{v_d \in Des(v_i)} \frac{1}{d(v_d, v_i)} \tag{9}$$

$d(v_d, v_i)$ is the distance between the node v_i and its descendant v_d. If there is an edge that directly links to them, their distance can be calculated as the reciprocal of the edge weight.

$$d(v_d, v_i) = \frac{1}{w(v_d, v_i)} \tag{10}$$

Otherwise, if v_d can reach v_i through a path (v_d, v_{d+1}, v_{d+2},…, v_i), then the distance between v_d to v_i will be the sum of edge distance along the path. In case that more than one path exists, the shortest path distance will be taken.

The steps of measuring node influence through graph entropy are shown below.

1. Compute the entropy of each node v_i as:

$$E(i) = -p(v_i) \cdot \log(p(v_i)) \tag{11}$$

2. Remove v_i and its edges from the graph
3. Calculate the entropy of remaining graph as $EN(i)$
4. Calculate the influence of node v_i as:

$$Inf_e(v_i) = \frac{EN(i)}{\log(EN(i) / E(i))} \tag{12}$$

The formula (12) is referred from [9], which proved to be able to identify important nodes in the network built of Enron (company) emails. We adopt it to measure the influence of node v_i by $E(i)$ and $EN(i)$, and try to find nodes which have higher centrality and more effect in the graph after they are removed from the graph.

4.4 Identify Influential Posts

To find the influential nodes, we ranked the nodes based on their influence scores from different measurements. Starters and connecters can be identified first as the preliminary result. Starters are determined by degree measure, and connecters are identified by using the other two methods. In our proposal, a connecter should fulfill two conditions: (i) Have a higher rank in the measurements of shortest-path cost or graph entropy. (ii) Connect two starters by different authors.

As we have defined influence from the aspects of starter and connecter, the influential nodes are either starters or connecters. Based on the combination of three measures, we are able to determine the most influential posts. The following are the heuristic used to determine the influential posts and potential influential users:

1. Remove the starters from the list of influential nodes if they are ranked low in all measurements.
2. Remove the connecters from the list accordingly if their connected starters are not influential.

3. Consider the connecters not critically influential if there are candidate connecters between the same set of starters.
4. Other starters and connecters are considered as influential posts, and their authors are considered as potential influential users.

5 User Graph Model

Although the influential starters and connecters are identified from the post graph, we still have the problem on determining the influential users. Consider the cases that (i) for a starter many of its followers are actually from a small group of users (one user can reply several times); (ii) a connecter links with two starters who have a large set of common follower users. In these cases the influence may be wrongly judged in the post graph model. Therefore we proposed the user graph model to refine the influence measures of potential influential users.

A user graph can be converted from the post graph. However, we are not going to build a complete user graph due to high computational complexity. As we are more interested in users who have made influential posts, we select the authors of starters and connecters in the post graph as seeds, then look at their neighbours and finally find possible connections between distant starters. The process of converting post graph to user graph will be discussed in next session.

The user graph is defined as $G_u(U, E_u)$, where U is the set of users, E_u is the set of directed edges which represent the relationship between users. Each node $u_k \in U$ is the author of post v_i^k in the Post Graph G_v.

Node types: There are three types defined in the user graph: starter, connecter and follower. Each node can belong to one or more types. At first, the type of a user is the summation of types of his posts. For example, starter users are the authors of starter posts identified in the post graph. But connecter is a special type. The author of a connecter in the post graph may no longer play the same role in the user graph. On the contrary, some new nodes could be detected as connecters in the user graph, even though none of their posts connect two starters in the post graph. Therefore the type of connecter will be determined after the graph conversion and measurement.

Edge types: $e(u_k, u_j) \in E_u$ is the edge directed from u_k to u_j representing that u_k is related to u_j, which means u_k has replied or responded to u_j either explicitly or implicitly (as defined in the post graph model). Besides, there are another type of virtual edge $e'(s_k, s_j)$ defined between two starters, to represent the directed path from s_k to s_j. The virtual edges are built when there are at least one directed paths between two starters, and their distance is very long. In this case, we will keep the shortest path length as the weight of virtual edge. The nodes on the paths are not important so it is not necessary to show them in the user graph. Moreover, the edges are directed and will be considered as two separate edges when they link two nodes in opposite directions.

Edge weight: $w(u_k, u_j)$ is the weight of edge $e(u_k, u_j)$ that measures the strength of their relationship. It is affected by the times of interactions and the relevancy of their conversations. For the virtual edges defined to link two starters, the edge weight $w'(s_k, s_j)$ is calculated as the shortest path length from s_k to s_j in user graph as described before.

6 Graph Conversion and Measures

In order to capture the influence of users, a user graph is needed. The next step is to convert the post graph to the user graph. Instead of processing the complete graph, we use a biased sampling method starting with potential influential users, who have posts as starters or connecters identified in the post graph. Then we propose several measurements to capture the influences of u-starters and u-connecters in different respects. (The terms "**u-starter**" and "**u-connecter**" are used to refer to the starters and connecters in user graph model.) For the rest of the section, we discuss the details of how to convert the post graph to user graph and measure the influences of u-starters and u-connecters. Figure 3 shows the overall flow of the operations and measurements on user graph.

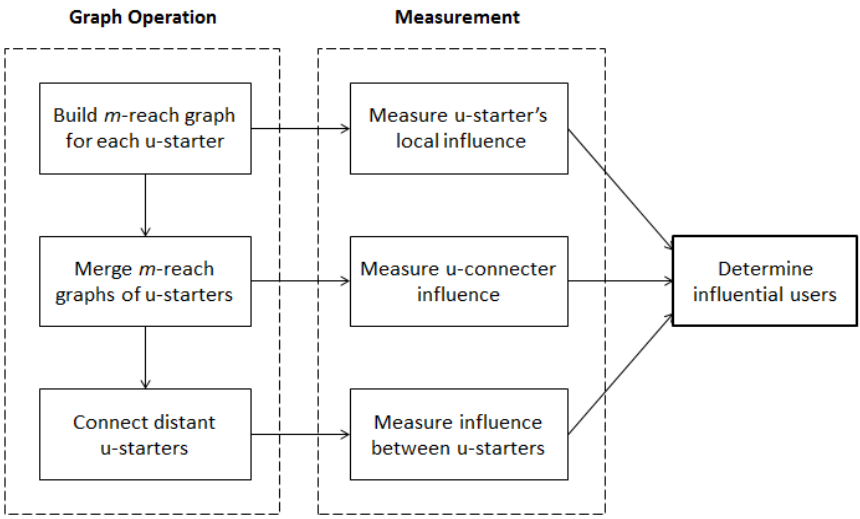

Fig. 3. Workflow of graph operation and measurement

6.1 Build *m*-Reach Graph for Each u-Starter

In the post graph model, a starter is observed when it has obtained a large amount of followers and descendants. However, it is hard to measure its influence on users, because one user may write a number of posts, or reply several times within a discussion. Moreover, if a user has several posts as starters, it is necessary to consolidate all the followers and descendants in terms of users. For this reason, we need the conversion from the post graph to a user graph where each user is represented as one node. But if the user graph is directly built for all discussions from different u-starters, some of their descendants will be merged and their influence may not be accurately judged. Therefore we first built an m-reach user graph for each u-starter in order to capture its local influence.

M-Reach Graph

"*M*-reach" is a measure defined by Borgatti[19] that counts the number of unique nodes reached by a given node in *m* links or less. In our user graph, $g^m(u_k)$ is u_k's *m*-reach graph which consists of nodes that can reach u_k via a path of length *m* or less. Here the path length is defined as the number of hops to go though without consideration of edge weights.

Discussion Thread and Discussion Chain

In the post graph, a starter together with its descendants forms a discussion thread. In the Post-reply Opinion Graph by Memon and Alhajj [20], they clearly defined the discussion chain which is different from discussion thread: "The discussion chains consist of the paths in the graph whose starting node is a root and ending node is a leaf when we inverse the direction of the edges." In Figure 4, a post-reply graph shows the difference between discussion chains and threads.

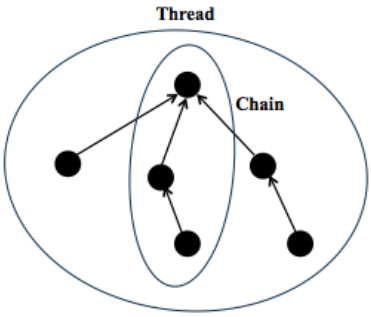

Fig. 4. Discussion threads and discussion chains

Algorithm of Building m-Reach Graph

Suppose the set of starters found in post graph $G_v(V, E)$ is $S \subset V$, each starter (post) $s_i^k \in S$ has an author u_k, then u_k is a u-starter. An *m*-reach user graph $g^m(u_k)$ will be built for each u-starter u_k. For each starter s_i^k by user u_k, the discussion thread in post graph will be converted to user graph $g^m(u_k)$. Here the value of *m* will be determined during the experiment.

In order to keep the information of distances (as defined in Section 4.3) from the starter to its descendants in a discussion chain, depth-first search (DFS) starting from s_i^k is conducted in the post graph G_v. For each descendant v_a^x of s_i^k (with authors u_x and u_k respectively), the shortest distance between v_a^x and s_i^k is notated as $d_a^{(x,k)}$.

While the distance considers edge weights in the post graph, the path length is defined differently for "*m*-reach". That is, the path length from v_a^x to s_i^k is the minimum number of distinct users on the path for v_a^x to reach the starter s_i^k. It is represented as $m_a^{(x,k)}$, and used to control the depth of searching. Suppose the value of *m* is given as m_0, the pseudo code is shown below:

```
1 for each starter s_i^k by user u_k
2    label s_i^k as visited, set m_i^(k,k) to 0
3    let S be a stack
4    S.push(s_i^k)
5    while S is not empty
6        v_a^x := S.top()
7        for each v_a^x 's unvisited follower v_b^y in G_v
8            label v_b^y as visited
9            if there is a visited node v_o^y with author u_y
10               m_b^(y,k) := m_o^(y,k)
11           else
12               m_b^(y,k) := m_a^(x,k) + 1
13           if m_b^(y,k) <= m_0
14               update g^m(u_k) with node v_b^y and edge e(v_b^y, v_a^x)
15               S.push(v_b^y)
16           continue at 5
         /*Reset the node v_a^x as unvisited after all its followers
           are visited, so that it can be visited in other path*/
17       delete m_a^(x,k) and label v_a^x as unvisited
18       S.pop()
```

The *m*-reach user graph $g^m(u_k)$ is built and updated during the process of DFS in the post graph (as shown in Step 14 above). In our user graph model, there are two basic attributes: node type and edge weight. The process of updating *m*-reach graph actually refers to changing the values of these attributes. The pseudo code below shows how to build and update for $g^m(u_k)$.

```
1 add node u_k with type (starter) in g^m(u_k)
2 for each node v_b^y and edge e(v_b^y, v_a^x) obtained from DFS in G_v
3    if v_b^y is not visited
4        if there is no user node u_y in g^m(u_k)
5            add a new node u_y in g^m(u_k)
6        add v_b^y 's type in u_y's type
7    if e(v_b^y, v_a^x) is not visited
8        if there is no edge from u_y to u_x in g^m(u_k)
9            build the edge e(u_y, u_x)
10           w(u_y, u_x) := w(v_b^y, v_a^x) /*initialize edge weight*/
11       if there is an edge e(u_y, u_x) in g^m(u_k)
12           w(u_y, u_x) := w(u_y, u_x) + w(v_b^y, v_a^x) /*update edge weight*/
```

An example of building *m*-reach graph is illustrated in Figure 5: (a) is a post graph showing the relationship between six posts (node 1 to 6) with four authors A, B, C and D; (b) is the *m*-reach user graph converted from (a), each node represents a user with its post IDs labeled in the bracket. In (a), node 1 is a starter with author A, the other nodes are its descendants. The edge weights are labeled beside the edges.

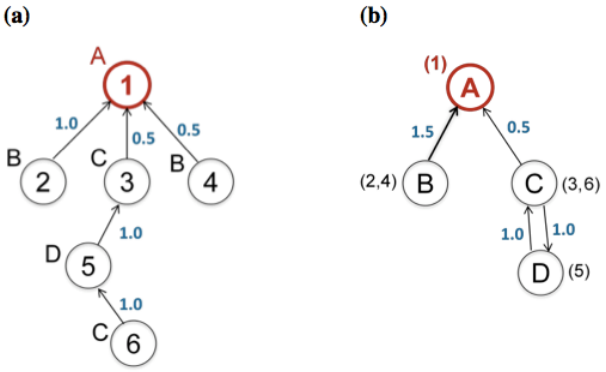

Fig. 5. Example of building *m*-reach user graph from post graph

Suppose we want to convert this post graph to an *m*-reach user graph with $m = 2$, node 2 and 4 are in 1-reach, and they belong to the same user B, so the two nodes are merged into one node B in (b), while the weight of the edge B→A is the sum of the weights for 2→1 and 4→1. If we look at the chain of nodes 1, 3, 5 and 6, the path lengths for the descendants to reach the starter are: $m_3^{(C, A)} = 1$; $m_5^{(D, A)} = 2$; $m_6^{(C, A)} = 1$. It should be noticed that the author of node 6 is C, which is the same as node 3, so the path length for node 6 is reduced to 1. If node 6 has followers by other users, those followers will have the path length equal to 2, therefore they will also be considered within *m*-reach.

As for the connecters, because they are defined as bridges to link with starters, they are certainly in 1-reach to a starter. This means all the connecters will be included some starter's *m*-reach graphs as long as $m \geq 1$.

6.2 Measure the Local Influence of u-Starter

The m-reach graph can be used to measure the local influence of u-starters. We proposed three measures to calculate a u-starter's influence in its m-reach graph from three aspects.

The distance-weighted centrality of a node has been defined in (10). It is a measurement that counts the number of its descendants with the weight reciprocal to their distances. The distance information can be obtained from the post graph and used for calculating the influence score of a u-starter on its descendants. It is defined that for the u-starter u_k, the maximum value of its influence on each user u_x is 1. $d_a^{(x, k)}$ is the shortest distance between v_a^x and s_i^k in G_v, then the influence score of u_k on u_x is:

$$I_k(u_x) = \mathrm{Min}\left(\sum_{v_a^x \in Des(s_i^k)} \frac{1}{d_a^{(x,k)}},\ 1 \right) \tag{13}$$

The centrality influence of u_k is the sum of influences on all its descendants in the m-reach graph $g^m(u_k)$. Let $C(u_k)$ be the centrality influence score of u_k.

$$C(u_k) = \sum_{u_x \in g^m(u_k)} I_k(u_x) \tag{14}$$

The users in an m-reach graph actually consist a community. Graph density is used to measure how many of the users within the community have interactions with many others. $|E_k^m|$ is the number of edges in the m-reach graph $g^m(u_k)$, $|V_k^m|$ is the number of nodes, and $|V_k^m|(|V_k^m| - 1)$ is the maximum possible number of edges in a directed graph. Let $D(u_k)$ be the graph density of u_k's m-reach graph.

$$D(u_k) = \frac{\left|E_k^m\right|}{\left|V_k^m\right|\left(\left|V_k^m\right|-1\right)} \tag{15}$$

The third factor considers how strong the interactions are in the u-starter's community. It is measured by the sum of weights of all the edges in the m-reach graph.

$$N(u_k) = \sum_{g^m(u_k)} w(u_y, u_x) \tag{16}$$

The three factors are summarized into an influence score of the u-starter u_k using the formula below:

$$M_S(u_k) = \frac{\alpha \cdot C(u_k) + \beta \cdot D(u_k) \cdot (\left|V_k^m\right| - 1) + \gamma \cdot N(u_k) / \left|V_k^m\right|}{\alpha + \beta + \gamma} \tag{17}$$

Each factor is weighted depending on user's need and the feature of real data. Besides, there are normalization factors associated with $D(u_k)$ and $N(u_k)$.

6.3 Merge *m*-Reach Graphs of u-Starters

Since the m-reach graph is built for each u-starter separately, it is possible that one user exists in several m-reach graphs. In this step we would like to merge the common user nodes as well as their edges in different m-reach graphs. Figure 6 shows an example of merging two u-starters' m-reach graphs ($m = 2$).

The process of merging nodes includes the combination of node types for the same user. Their associative edges will be added together. The edge weight will remain the same if there is only one edge from one user to another. In the cases that more than one edges exist between two users with the same direction, these edges will be merged and the maximum edge weight among them will be taken as the merged edge weight. They are associative operations. So the overall action of merging is associative, which means the result is unique no matter what the merging sequence is.

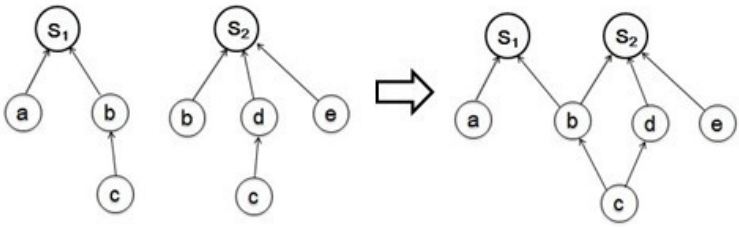

Fig. 6. Merge 2-reach graph of two u-starters

6.4 Measure the Influence of u-Connecter

After merging the *m*-reach graphs of different u-starters, the u-connecters should be in a graph joining all *m*-reach graphs from u-starters they connect.

First, if a u-connecter links two u-starters that already directly connected in the user graph, it is determined no longer a connecter as there is no need to have a connecter here.

For the existing u-connecters, there should be a way to measure and compare their influences. We would like to adopt the method of Shortest-path Cost Measurement used in the post graph model which approved useful to identify connecters. The basic idea is to remove the u-connecter from the user graph and measure the impact on the influence propagation from the u-starters. The same formula is used to calculate the influence of a u-connecter u_k:

$$M_C(u_k) = \sum_{u_d \in Des(u_k)} (C(u_d, G_u, \varnothing) - C(u_d, G_u, u_k)) \tag{18}$$

However, this formula has a different meaning, as the ancestors are replaced with a u-starter here. Let $C(u_d, G_v, u_k)$ be the sum of the *relationship strength* (as defined in Section 4.2) from u_d to the u-starter after removing u_k from the graph G_v. This u-starter should be the parent of the u-connecter u_k. In the case that u_k has several u-starters as parents, the sum of measuring results for several u-starters will be taken as the final influence score of u-connecter u_k.

Besides the existing ones, some new u-connecters may be found as a broker to link two u-starters (one is his parent and the other is his child in the user graph). We can also use the above method to measure their influences. But the new u-connecters do not have a post as connecter in the post graph, which means they have not behaved as connecter within a discussion, they are only considered as potential connecters who should have the ability but have not conducted.

6.5 Connect Distant u-Starters

After merging the *m*-reach graphs, still there may be disconnected subgraphs, or some isolated *m*-reach graphs of u-starters. In order to connect them and discover inter-starter influences, we built virtual edges between distant u-starters.

The u-starters are defined as distant when they do not exist in each other's *m*-reach graphs (e.g. S_1 and S_3 shown in Figure 7). To find possible connections between distant u-starters, we first looked up in the post graph and determined the existence of directed path between them. For example, if u_j and u_k are not in the *m*-reach graph of each other, first we want to check in the post graph if there is a directed path from u_j's post to u_k's. Let v_i^k (i = 1,2,3,…) be u_k's posts in G_v. For each v_i^k, it has a descendant set $Des(v_i^k)$. We need to find out whether u_j has a post v_d^j in $Des(v_i^k)$.

Once the condition is met, it means at least one path exists from u_j to u_k, then we will build an virtual edge from u_j to u_k. The edge weight is calculated as the shortest path length (number of distinct users) between them. Similarly, we can check if the inverse path from u_k to u_j exists. The edges are considered as different in opposite directions.

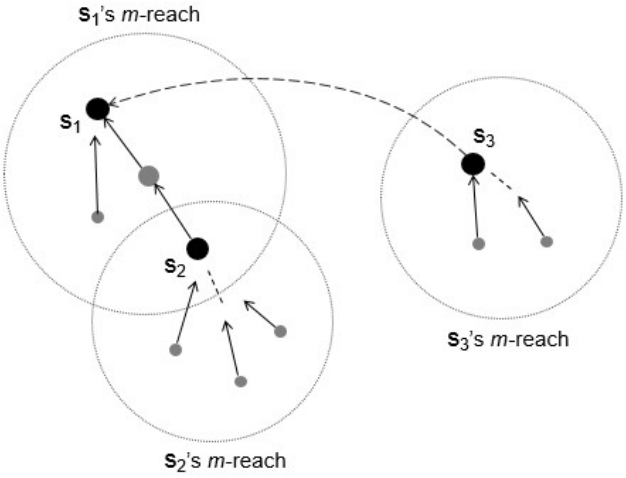

Fig. 7. Connect distant u-starters (from S_3 to S_1)

6.6 Measure the Influence between u-Starters

After all the possible virtual edges are built, the influence of one u-starter s_k on another one s_j can be calculated by:

$$I_k'(s_j) = \begin{cases} 1 & \text{if } s_j \text{ is in } g^m(s_k) \\ \min\left(\dfrac{m}{w'(s_j, s_k)}, 1\right) & \text{if } e'(s_j, s_k) \text{ exists} \end{cases} \tag{19}$$

where the value of m will be determined in the experiment. An example is shown in Figure 7 that S_1, S_2 and S_3 are u-starters, and S_1 has influence on S_2 and S_3. The influence of S_1 on S_2 counts as 1 as S_2 is in S_1's *m*-reach graph, while S1's influence

on S3 is measured by the second formula. Finally, the influence of the u-starter s_k on other starters is the summation of influences on each one:

$$M_I(s_k) = \sum I_k'(s_j) \tag{20}$$

7 Experiment

7.1 Case Study for Post Graph Model

Our proposed model can be applied for different social media. Both explicit and implicit relationships can be identified between text-based posts. We chose Twitter to conduct the experiments as it has many users and its data are easy to collect.

In order to find the most influential posts and their respective authors during the information diffusion within a topic, we select a general user (neither famous people nor public media) who has written some posts on a topic, find the user's friends who have responded to the posts or also talked on this topic, then dig out the friends of friends and so on. General sampling method is not suitable here, because we need the data from users with more connections between them so that the graph can be well formed. Tweet data are collected on the topic of "Steven Jobs and iPhone 4s". The keyword set is defined as {"iPhone 4", "iPhone 4s", "iPhone 5", "iPhone Mini", "Steve Jobs", "Apple", "ios 5", "Siri"}. Table 1 gives the data description for the experiment.

Table 1. Description of data

Platform	Twitter
Topic	Steven Jobs and iPhone 4s
Time	11/10/2011 - 31/10/2011
Location	Hong Kong
No. of users	158
No. of tweets	211

Preliminary Results

Preliminarily, starters and connecters can be found after the three influence measurement methods are applied. As mentioned before, degree measure can be used to identify starters. Two factors are calculated: (i) the degree of each node $d(v)$; (ii) the weighted average of its follower in-degrees $s(v)$. The top nodes that $d(v) + s(v) > 2$ are selected. The results are plotted in the diagram shown in Figure 8.

It is observed that the results of the two factors are not aligned most of the time. The reason is that a node with higher degree should have more followers, and it becomes difficult for all its followers to have a high in-degree. On the contrary, there exist some nodes with only a few followers, but most of the followers have high in-degree. These nodes can be detected by high score of $s(v)$. For our work, we finally selected the 10 nodes with $d(v) > 3$ and $s(v) > 0.1$ as starters (Node 1 – 10 labeled in Figure 9).

As for the connecters, we integrate the results from Shortest-path Cost Measure (SCM) and Graph Entropy Measure (GEM). After calculating the influential scores $Inf_c(v_i)$ and $Inf_e(v_i)$, all the nodes are ranked. After examining the top ranking nodes, besides the found starters, other nodes which connect starters are considered as connecters (Node 11 – 20 in Figure 9). The connecters discovered by each method are listed below in ranking order.

- SCM: 11, 14, 13, 12, 15, 16, 20, 17 (nodes labeled in Figure 9)
- GEM: 11, 14, 12, 15, 13, 16, 20, 17, 18, 19 (labeled in Figure 9)

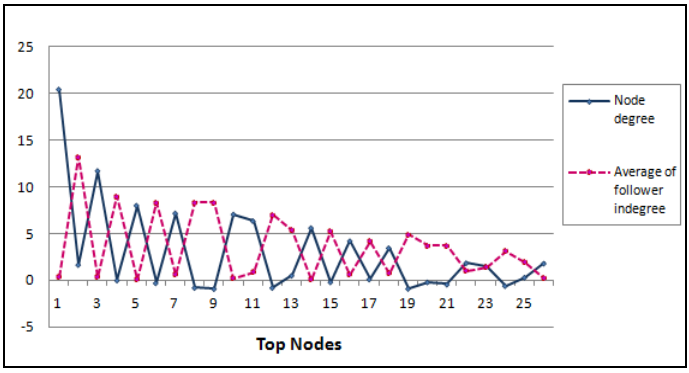

Fig. 8. Degree measures

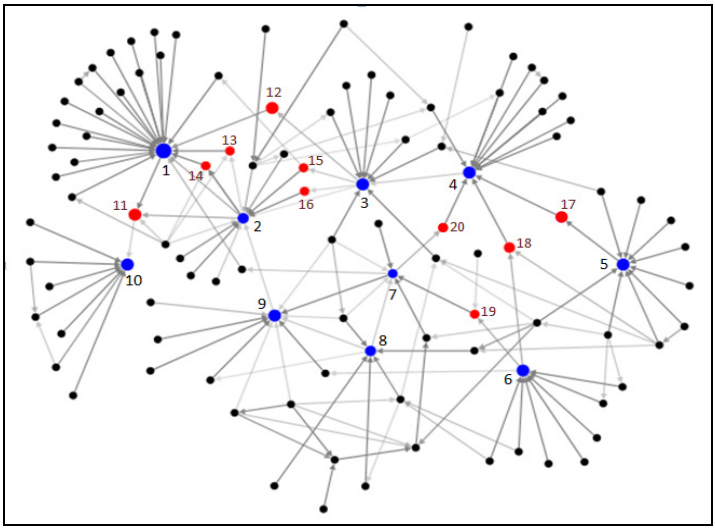

Fig. 9. Graph of starters and connecters

Discussion and Final Results

In comparison, SCM only identifies 4 starters in its top 10 ranking nodes, and is able to find all starters in top 21; while GEM can find 7 starters in top 10 and all starters in top 14. It is because GEM looks into both node entropy $E(i)$ and remnant graph entropy $EN(i)$ in calculating the influence score, which is aimed to achieve high node centrality as well as large effect in the graph after removal. As for the SCM algorithm, we can see that its influence score is in the range from 0 to the number of the node's descendants. There is no difference between its close followers and distant descendants when measuring a node if the weights are all 1. As a result, it is more likely to find the nodes with more descendants, whereas GEM can find the nodes with more ancestors or descendants.

Finally, we can find the most influential posts considering the results of all measurement. For the starters, node 7 and node 8 by different authors are ranked low by SCM and GEM, so they are not considered to be influential in the final result. Since node 7 is not influential any more, we look at the connecters 19 and 20 that connect node 7. It is found that they also have relatively low rankings. Therefore they are also removed from the influential list.

Noted that not every node that connects two starters can be a connecter, the connecters are detected by the two measurements, which means their removal from the graph will have a certain impact on the information transmission, and they should have some followers to make them more influential. In Figure 9, we can see that nodes 13 and 14 are actually connecting the same starters 1 and 2, and so are the nodes 15 and 16 which connect starters 2 and 3. In this case, we consider them not to have critical influences.

7.2 Case Study for User Graph Model

In order to compare the results in finding influential users in post graph and user graph, a larger data set is needed for experiment. In this case study, we collected more than 1700 tweets from 915 users, on the topic of "Sichuan Lushan earthquake" (an earthquake happened in China on April 20, 2013) and "H7N9 influenza". More description on the data set is shown in Table 2.

Table 2. Description of data

Platform	Twitter
Topic	"Sichuan Lushan earthquake", "H7N9 influenza"
Time	31/03/2013 - 30/04/2013
Location	China, Hong Kong, Japan

Table 3. Top 5 influential users in post graph

Rank	1	2	3	4	5
No. of starters and connecter	6	6	5	3	2

Influential Users in Post Graph

Similar actions are taken as in previous case study to identify starters and connecters in the post graph. Finally, 49 starters and 5 connecters are found in the posts. Some starters or connecters are actually written by the same authors, so we identified 29 users as influential in total. If we rank the influential users found in post graph according to the number of starters/connecters they have, the top 5 users are listed below: (For those with the same number of starters and connecters, they are ranked based on the highest ranking of their posts.)

In order to justify our user graph model, we converted the post graph into user graph, and then measured the users' influences in the user graph of two types: u-starter and u-connecter.

U-starter Influence

The local influence of a u-starter is measured by three factors as shown in (18). In this experiment, we put more weight on the centrality measure, set $\alpha = 2$, $\beta = 1$, $\gamma = 1$. The value of m is decided by the distance between close starters in the post graph. We tried to make more m-reach graphs contain only one starter, meanwhile have common descendant nodes so that they are connected after the merging operation. For the cases that the starters are far away from each other, we suggested the m value not larger than 5. The influence between u-starters is also taken into consideration. When some u-starters have similar local influence, their ranking will be judged by the inter influence measure. After all, the ranking of influential starters is a little different from that in post graph. In top 5 influential users found:

- Top 2 users keep the same.
- A new influential user is identified on rank 3 in user graph.
- The user on rank 4 in post graph does not rank on top in user graph.

The new influential user found in the user graph only has one post as starter. But this starter has a large number of followers, and these followers have interactions with each other, which makes its local influence score higher. Figure 10(a) and 10(b) shows the post graph and corresponding user graph of this node and its descendants within 5-reach. For the user falling off the top 5 list, the main reason is that his followers or friends are from a small community, and there are no connecters to propagate their discussion to another community.

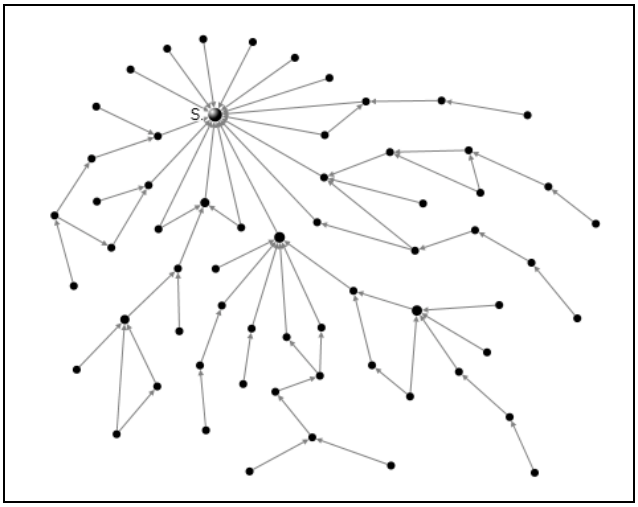

Fig. 10(a). Post graph of an influential starter

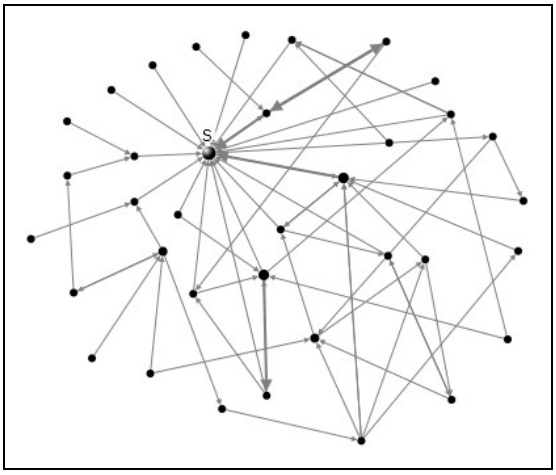

Fig. 10(b). User graph of the starter

U-Connecter Influence

In the post graph there are 5 connecters identified. But after the graph is converted into user graph, it is found that 2 of them are not connecters anymore, because the u-starters they link with are directly connected. The remaining 3 u-connecters are determined to be influential users.

However, 1 new u-connecter is found in the user graph, who link with 2 different u-starters. As stated above, it is only considered as a potential connecter. The result proves that in the post graph the connecters already identified can be refined and some new connecters may be found. The new connecters are not that influential as

they are just supposed to have the ability but have not acted as a connecter in our data set. Therefore the identification of influential connecters will be more accurate and complete if the data set is large enough.

8 Conclusion and Future Work

In this paper, we dealt with the problem of finding influential users based on their interactions in social networks. Different from other's work, we tried to identify the most influential users in different roles through their posting on the same topic. Additional contributions are the following:

- We proposed a general graph model showing the relationship between posts that can be applied in different social media platforms.
- We presented three methods to measure the influences of online posts to distinguish starters and connecters in the graph. We specially defined the node centrality and graph entropy for our model.
- We converted the post graph to user graph using biased sampling, and proposed different measurements to clarify the influences of starters and connecters.

Our graph model has its advantage in dealing with online posts and users with more interactions. Therefore we carried out case studies and visualize the graph to validate the model. In future, we will apply the model on different social media platform to carry on experiments on larger data set. Furthermore, this model can be more effective if it is integrated with advanced text mining techniques, so that the relevance between posts can be judged more accurately.

References

1. Bakshy, E., Mason, W.A., Hofman, J.M., Watts, D.J.: Everyone's an Influencer: Quantifying Influence on Twitter. In: Proceedings of the 4th ACM International Conference on Web Search and Data Mining, WSDM 2011, Hong Kong, China, pp. 65–74 (2011)
2. Bakshy, E., Karrer, B., Adamic, A.: Lada. Social Influence and the Diffusion of User-Created Content. In: 10th ACM Conference on Electronic Commerce, Stanford, California. Association of Computing Machinery (2009)
3. Leavitt, A., Burchard, E., Fisher, D., Gilbert, S.: The Influentials: New Approaches for Analyzing Influence on Twitter. Web Ecology Project (2009), http://tinyurl.com/lzjlzq
4. Klout Score, http://klout.com/home
5. Twinfluence, http://twitterfacts.blogspot.com/2008/10/twinfluence.html
6. Weng, J., Lim, E., Jiang, J., He, Q.: TwitterRank: Finding Topic-sensitive Influential Twitterers. In: Proceedings of the 3rd ACM International Conference on Web Search and Data Mining, WSDM 2010, pp. 261–270 (2010)

7. Hansen, D.L., Shneiderman, B., Smith, M.A.: Visualizing Threaded Conversation Networks: Mining Message Boards and Email Lists for Actionable Insights. In: An, A., Lingras, P., Petty, S., Huang, R. (eds.) AMT 2010. LNCS, vol. 6335, pp. 47–62. Springer, Heidelberg (2010)
8. Mathioudakis, M., Koudas, N.: Efficient Identification of Starters and Followers in Social Media. In: Proceedings of the 12th International Conference on Extending Database Technology: Advances in Database Technology, EDBT 2009, pp. 708–719 (2009)
9. Shetty, J., Adibi, J.: Discovering Important Nodes through Graph Entropy. In: The 11th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (2005)
10. Nobel, C., Cook, D.J.: Graph-based anomaly detection. In: Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 631–636 (2003)
11. Ilyas, M.U., Radha, H.: A KLT-inspired Node Centrality for Identifying Influential Neighborhoods in Graphs. In: Conference on Information Sciences and Systems, pp. 1–7 (2010)
12. Tang, L., Liu, H.: Graph Mining Applications to Social Network Analysis. Managing and Mining Graph Data. In: Managing and Mining Graph Data, pp. 487–513 (2010)
13. Sala, A., Cao, L., Wilson, C., Zablit, R., Zheng, H., Zhao, B.Y.: Measurement-calibrated Graph Models for Social Network Experiments. In: Proceedings of the 19th International Conference on World Wide Web, WWW 2010, pp. 861–870 (2010)
14. Wilson, C., Boe, B., Sala, A., Puttaswamy, K.P.N., Zhao, B.Y.: User interactions in social networks and their implications. In: Proceedings of EuroSys, pp. 205–218 (April 2009)
15. Scott, J.: Centrality and Centralization. In: Social Network Analysis: a Handbook. SAGE Publications, London (2000)
16. Lipkus, A.H.: A proof of the triangle inequality for the Tanimoto distance. J. Math. Chem. 26(1-3), 263–265 (1999)
17. Sun, B., Ng, V.T.Y.: Lifespan and Popularity Measurement of Online Content on Social Networks. In: Social Computing Workshop of IEEE ISI Conference, pp. 379–383 (2011)
18. Dehme, M.: Information processing in complex networks: Graph entropy and information functionals. In: Applied Mathematics and Computation, vol. 201, pp. 82–94 (2008)
19. Borgatti, S.P.: Identifying Sets of Key Players in a Social Network. Comput. Math. Organiz. Theor. 12, 21–34 (2006)
20. Memon, N., Alhajj, R.: From Sociology to Computing in Social Networks
21. Tunkelang, D.: A Twitter Analog to PageRank (2009), http://thenoisychannel.com/2009/01/13/a-twitter-analog-to-pagerank/
22. Sun, B., Ng, V.T.Y.: Identifying Influential Users by Their Postings in Social Networks. In: Proceedings of the 3rd International Workshop on Modeling Social Media, pp. 1–8 (2012)