Follow Link Seeking Strategy: A Pattern Based Approach

Agus Trisnajaya Kwee, Ee-Peng Lim, Palakorn Achananuparp, Feida Zhu
School of Information Systems
Singapore Management University
{aguskwee, eplim, palakorna, fdzhu}@smu.edu.sg

ABSTRACT

The way *Twitter* and other microblogging networks work is to have users create follow links among one another, and create short messages to their followers. Most of the time, the creation of follow links to other users does not require approval from the latter. Therefore, it is very easy for a user to create such links. On the other hand, the same cannot be said for seeking incoming follow links which is useful in some application scenarios. In this paper, we therefore study the *Follow Link Seeking Problem* that aims to find the strategies for a source user to maximize the likelihood of receiving a follow link from a target user. We formulate this problem as a recommendation task and generate a set of strategies from well known *follow link patterns*. Using the confidence scores of follow link patterns, we derive the success probability of each strategy. Finally, we present FRIENDER, a working recommender system for follow link seeking strategies. The system performs localized crawling of the target user, computes the required strategies on the fly, and presents the strategies visually.

Keywords

Twitter, follow link seeking, follow link patterns

1. INTRODUCTION

Motivation. Twitter is a highly popular microblogging service with users posting 140-character long messages (also known as *tweets*). When a user is followed by other users, the latter will be able to read her tweets on their public timelines. In general, it does not require any permission for a user to create follow links to other users. Users are instead encouraged to follow others so as to obtain interesting information. Due to its popularity and realtime tweet generation and dissemination, Twitter is able to pick up breaking news on events that have not yet been published by official news media. For example, the death of Osama bin Laden was picked up by Twitter before traditional news media[1]. It is therefore not a surprise that millions of users today depend on Twitter to receive realtime updates of news and events.

A Twitter user can follow other users and becomes their *follower*. On the other hand, a user can be followed by other users and becomes their *followee*. Followers create follow links to their followees for various reasons. One possible reason is that the latter provides information that interests the former. Another reason is that they are friends or family members. A Twitter user having many followers enjoys popularity which is extremely important when the user wants his information to reach out to a wide audience. Examples of such users include celebrities and world leaders. In some marketing applications, a marketing agent may even want to seek follow links from specific user or user groups so as to provide targeted information to the user or groups.

Given the importance of follow links, we therefore need to address the research question: What is a quick way for a user to gain follow link from specific target users? In this paper, we define the *Follow Link Seeking Problem* to be one that aims to find strategies for a source user to maximize the likelihood of receiving a follow link from a target user. A strategy here refers to a series of actions one should perform to achieve the desired outcome of being followed.

The follow link seeking problem is novel and has not been studied earlier. Its solution can be divided into two parts: (a) generation of strategies of user actions; and (b) ranking of strategies. There are different user actions one can consider. In Twitter, the possible user actions include: (i) user follows another user; (ii) user generates a tweet; (iii) user retweets another user; (iv) users mentions another user; (v) user unfollows another user; etc. In this paper, we shall confine to (i) (i.e., user follows another user) only given that it is a very common action, requires least effort, and does not involve content analysis. A more comprehensive study of strategies involving other user actions will be carried out as part of our future work.

The follow link seeking problem is also distinctive from the more frequently studied *followee recommendation*[6, 5, 12, 11] problem. The followee recommendation problem focuses on recommending new follow links to specific users personalizing to their interests. The results of followee recommendation is a ranked list of candidate followees. The follow link seeking problem, in contrast, is defined to have a specific followee and desired follower. It requires a list of strategies to be recommended to the followee so that the desired follower can be acquired. The result consists of a ranked list of strategies (instead of users).

To solve the follow link seeking problem, one may consider

The 6th SNA-KDD Workshop '12 (SNA-KDD'12), August 12, 2012, Beijing, China. Copyright 2012 ACM 978-1-4503-1544-9 ...$15.00.

a trivial strategy of getting the user to follow the desired target user hoping that the latter will return a follow link. This solution has several shortcomings. It assumes that the target user will always reciprocate follow links. The assumption may not hold as the follow behavior varies with users. If a target user does not practise reciprocity, the above strategy will deem to fail. Even if the target user practises reciprocity, one may still want to adopt another strategy if direct following the target user is not a preferred action.

The other extreme strategy is to follow the desired target user and all other users following or being followed by the target user. This may increase the likelihood the target user following back but it comes with a cost, i.e., to find all the followees and followers of the target user and follow them. There is also the additional cost of having to receive all tweets generated from these users, which is also known as the information overloading cost. What is needed is therefore a more principled solution approach.

Overview of our approach. In a social-information network such as Twitter, different users demonstrate different follow behaviors. Any strategy to be recommended for seeking follow link should therefore be personalized to the target user. Our proposed approach deals with this by learning follow link patterns of each target user, rather than the follow link patterns of all users. The pattern learning step involves only local structures making it possible to perform recommendation on the fly. Strategies are generated from these follow link patterns and their success probabilities are derived from the confidence of these patterns using a probabilistic framework. While our method does not make use of any tweet content and other user actions, it can be easily extended to cover both content and other user actions and shall be included in our future work.

Contributions. The contributions of this paper are as follow:

- We formally define the follow link seeking problem and develop method to generate and rank strategies for a user to maximize the likelihood to get a follow link from a specific target user. Our method is data driven as it uses past follow link data to determine the extent to which the target user practises different well known follow link patterns. It is probabilistic as every strategy is generated with some likelihood value. It is also designed to utilize local relationship patterns only so as to be efficient.

- We conduct experiments to evaluate our proposed method against a range of baseline methods. The result shows that our method can predict more accurate source users whom a target user will link to than using the common neighbor methods. We also observe that users demonstrate different behaviors of following others which have been captured by the confidence of their follow link patterns.

- We develop FRIENDER, an interactive graphical user interface, to visualize the follow strategies generated by our proposed method for a given Twitter network. The user interface uses on-the-fly computation to generate the follow link patterns and their confidence scores. For each strategy, it animates the follow steps included in the strategy.

Paper Outline. The rest of the paper is organized as follows. Related work is presented in Section 2. Follow link patterns serving as the backbone of our method is described in Section 3. Sections 4 and 5 describe the proposed method, and experiments and its results, respectively. A graphical user interface to visualize our method is described in Section 6. Finally, Section 7 concludes this paper.

2. RELATED WORK

Follow link seeking is a novel problem which is closely related to followee recommendation. The main difference between two problems is that the former is about suggestion of actions while the latter is about suggestion of users. Follow link seeking task has a goal of getting connected with some target users, while the goal of followee recommendation is to get connected with any relevant target users. Underlying the two problems, however, are some common principles which foster the formation of follow links. Hence, we chose to survey some link formation research works below, particularly those related to Twitter.

Yin et al. mentioned that there are both social and information needs that motivate users to follow other users in Twitter[13]. As users have friends in their real life, they want to stay socially connected to these friends in Twitter as well. To receive interesting information, users are also motivated to follow a user who represents a source of information [11]. By following the latter, users can gain access information conveniently and quickly.

In [12], a structural follow link pattern approach for predicting links was proposed for Twitter. The proposed pattern-based link prediction method captures both social and information interest of users and it outperforms other link prediction methods, such as PropFlow [8], Katz [7], and Adamic/Adar [3]. Similar study on directed triadic closure in a small random sample of Twitter users was conducted by Romero and Kleinberg [11]. A directed triadic closure is essentially a follow link pattern with time ordered links. The work concluded that the presence of directed triadic closures in Twitter is higher than that of egocentric networks.

Nguyen et al. [9] complement Romero and Kleinberg's finding [11] by describing two important constraints in the directed triadic closure. First, the existence of the specific *pre-condition* relationship(s) related to start node u_1 and/or end node u_2. In the previous example, these pre-condition relationships are the links from u_1 to u_3 and from u_3 to u_2. Second constraint is the *temporal constraint.* It says that the relationship between u_1 and u_2 must be formed after pre-condition relationships are formed.

Golder and Yardi [6] described four relationship ties in Twitter in the *attention-information* perspective. They are shared interest, shared audience, transitivity, and reciprocity. Golden and Yardi found that transitivity and reciprocity ties are the important factors that increase the likelihood of forming relationships in Twitter. Brzozowski and Romero compared different types of structural closures and concluded that certain types of structural closures outperformed traditional followee recommendation methods, such as collaborative filtering, behavioral, and similarity based methods[5].

3. FOLLOW LINK PATTERNS

The core knowledge used in our follow link seeking method is a set of follow link patterns adopted by Twitter users. Follow link patterns are some local structural patterns often observed in directed social networks. The follow link patterns are also known as relationship patterns [13], link formation patterns[9], and structural closures (which include dyadic and triadic closures)[10, 5] as reviewed in Section 2.

In the follow link seeking problem, we have a source user u_s wanting to seek a follow link from a target user u_t. The follow link patterns that tell us how u_t creates follow links to others will be most relevant to determining the strategies for u_s. We define each follow link pattern p to be in the form *pre-pattern* $\Rightarrow u_x \leftarrow u_y$ where u_x and u_y represent variables corresponding to source and target users respectively. Here, we use u_x, u_y and u_z as user variables. The pre-condition pattern $pre - pattern$ represents a local structure that represents the set of follow links that connect u_x with u_y before u_x receives a follow links from u_y (i.e., $u_x \leftarrow u_y$).

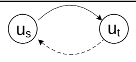
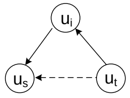
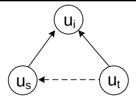
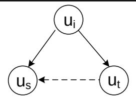
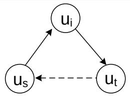
In the previous work, researchers have studied follow link patterns that are derived from the well studied dyadic and triadic structures in social network research [9, 13, 12, 6]. These patterns are shown in Table 1 and we also adopt them as follows:

- *Reciprocity* ($u_x \rightarrow u_y \Rightarrow u_x \leftarrow u_y$): The reciprocity pattern involves some user u_s following u_t as the pre-condition pattern before u_t follows u_s back. The two users may already know each other socially and hence the mutual follow links. When u_t finds out about u_s as a new follower, she may also find u_s's tweets interesting and decide to follow u_s. Even if u_t may not know anything about u_s, there is a good chance u_t follows u_s back to show appreciation of being followed.

- *Transitivity* ($u_x \leftarrow u_z \leftarrow u_y \Rightarrow u_x \leftarrow u_y$ for some intermediate user variable u_z):
  Suppose u_s and u_t and u_i are users taking the roles of u_x, u_y and u_z respectively. The follow link $u_s \leftarrow u_i$ and $u_i \leftarrow u_t$ suggest that u_i is interested u_s's tweets and may forward (retweet) some of them to u_t. In this way, u_i filters u_s's tweets before retweeting them to his followers including u_t[4]. u_t, a follower of u_i may decide to receive a full set of u_s's tweets by directly follow u_s, the information source. The transitive pattern, according to [6] is the most significant pattern that contributes to new follow links.

- *Common Followee* ($u_x \rightarrow u_z \leftarrow u_y \Rightarrow u_x \leftarrow u_y$ for some intermediate user variable u_z):
  Suppose u_s and u_t and u_i are users taking the roles of u_x, u_y and u_z respectively. When u_s and u_t follows u_i, both u_s and u_t expect to receive interesting tweets from u_i[13]. We may infer that u_s and u_t may have similar interests. This common interest may motivate u_t to follow u_s[5]. As the number of u_i's increases, the common interest between u_s and u_t are expected to strengthen. As the result, the likelihood of new follow link between u_t and u_s also increases.

- *Common Follower* ($u_x \leftarrow u_z \rightarrow u_y \Rightarrow u_x \leftarrow u_y$ for some intermediate user variable u_z):
  Suppose u_s and u_t and u_i are users taking the roles of u_x, u_y and u_z respectively. As the follower of u_s and u_t, u_i becomes their common audience [6][12], as all u_s and u_t's tweets are received by u_i. Sharing the same follower suggests that both u_s and u_t may have similar interests that may motivate them to form a link between them. This will happen even more likely when the number of common followers is large.

Pattern	Name
	Reciprocity rcp
	Transitivity trt
	Common Followee cfe
	Common Follower cfr
	Cycle cyc

Legend
⇢ final link
⟶ pre-condition link

Table 1: Follow Link Patterns

- *Cycle* ($u_x \rightarrow u_z \rightarrow u_y \Rightarrow u_x \leftarrow u_y$ for some intermediate user variable u_z):
  Suppose u_s and u_t and u_i are users taking the roles of u_x, u_y and u_z respectively. Given that u_s follows u_i and u_i follows u_t, there could be some social relationships among them. Similar to reciprocity, u_t may therefore get to know u_s and decide to follow u_s later.

While the above patterns are common, they are not adopted the same way by all Twitter users. Depending on the users and their neighbors, some patterns can be more likely adopted than others. The behaviors of following others are therefore different among users. For example, there may be a user who always reciprocate follow links from others while another user hardly does so.

In this paper, we would like to use the above follow link patterns to recommend strategy for follow link seeking. Instead of assuming every user behaves identically, we allow each user to adopt follow link patterns with personalized preference. Hence, for each user, we would like to determine the likelihood that he will adopt every follow link pattern. This further distinguishes our approach from the existing methods[5, 13, 12, 6].

4. PROPOSED FOLLOW LINK SEEKING METHOD

Our proposed follow link seeking method consists of three steps that begin with taking a source user u_s and a target user u_t as user given input, and end with a set of strategies for u_s to gain the follow link from u_t. A *follow link seeking strategy* is a series of one or more follow actions to be carried out by u_s. Each follow action in the strategy is expected to satisfy the pre-condition pattern(s) of one or more follow link patterns that leads to the formation of a new follow link (e.g., $u_t \rightarrow u_s$) with some likelihood value. The user

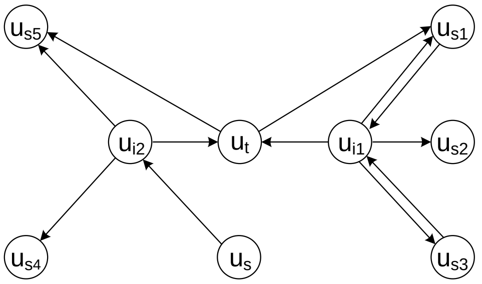


Figure 1: Neighborhood Network Example

u_s can then refer to the likelihood of these follow actions to decide which strategy to adopt.

The three steps are:

1. *Crawling neighborhood network*: This entails downloading the neighborhoods of u_t. The set of users and links covered by this neighborhood provides the historical data for learning the follow link behaviors of u_t and other users around him.

2. *Learning follow link behaviors*: The past follow link behaviors of users tell us how much they adopt the five follow link patterns. In this step, we determine the user-specific confidence of adopting these follow link patterns.

3. *Constructing follow link seeking strategies*: This step examines the different user action options that can be taken by u_s and computes the likelihood of u_s acquiring the follow link from u_t for each action.

We will elaborate the latter two steps in the following subsections.

4.1 Follow Link Behavior Learning

Neighborhood network of target user. We learn the follow link behavior from the neighborhood of u_t so as to tell how u_t adopts follow link patterns. We define the neighborhood of u_t (denoted by G_{u_t}) to be the set of users connected to u_t within two hops and the set of follow links among them. Figure 1 shows the direct and the two-hop neighborhoods of u_t. Each follow link is associated with a timestamp which allows the follow links to be ordered by time.

The source user u_s may or may not exist in G_{u_t}. The example neighborhood network of u_t in Figure 1 illustrates one which includes u_s. From the network, we extract *follow link pattern instances* for each pattern mentioned in Section 3 by instantiating u_y in the pattern by u_t, and u_s and u_z by other users. That is, we want to find those instances involving u_t as the target user node in order to determine the confidence (or likelihood) of u_t adopting each pattern.

Pattern instances, target-specific support and confidence. Given any follow link pattern p, we denote the instance set of p in a network $I(p)$ by the set of instances in the network after instantiating variable(s) in p by actual nodes. The **support** of p is thus defined to be size of $I(p)$, i. e., $Sup(p) = |I(p)|$. The **confidence** of p is defined by $Conf(p) = \frac{Sup(p)}{Sup(q)}$ where q is the pre-condition of p. Next, we define **target-specific support and confidence** for a given pattern. These are the support and confidence of pattern after instantiating u_y with u_t and u_z (for triadic patterns only) with some other user.

For example, we represent the reciprocity pattern by $p_{rcp} = u_x \rightarrow u_y \Rightarrow u_x \leftarrow u_y$. When we instantiate the pattern with specific target user u_t while leaving the source user to be variable, the pattern can be associated with a set of instances denoted by $I(p_{rcp}(u_y = u_t))$ each containing a specific source user having two way links with u_t. With the above target user instantiation, we define target specific support of $p_{rcp}(u_y = u_t)$ to be $Sup(p_{rcp}(u_y = u_t)) = |I(p_{rcp}(u_y = u_t))|$ and target user specific confidence to be $Conf(p_{rcp}(u_y = u_t)) = \frac{Sup(p_{rcp}(u_y=u_t))}{Sup(q_{rcp}(u_y=u_t))}$. Without any loss of semantics, the target specific support and confidence can be written as $Sup(p_{rcp}(u_t))$ and $Conf(p_{rcp}(u_t))$ respectively.

For a triadic pattern such as transitivity pattern, $p_{trt} = u_x \leftarrow u_z \leftarrow u_y \Rightarrow u_x \leftarrow u_y$, the target specific support and confidence are defined as:

$$\begin{aligned} Sup(p_{trt}(u_t, u_i)) &= Sup(p_{trt}(u_y = u_t, u_z = u_i)) \\ &= |I(p_{trt}(u_y = u_t, u_z = u_i))| \end{aligned} \quad (1)$$

$$\begin{aligned} Conf(p_{trt}(u_t, u_i)) &= Conf(p_{trt}(u_y = u_t, u_z = u_i)) \\ &= \frac{Sup(p_{trt}(u_y = u_t, u_z = u_i))}{Sup(q_{trt}(u_y = u_t, u_z = u_i))} \end{aligned} \quad (2)$$

Unlike the reciprocity pattern, the transitivity behavior of u_t is also determined by the intermediate user u_i. As there may be different intermediate users who have direct follow links with the target users, different target specific support and confidence are defined for each of the intermediate users. The target specific support and confidence of other triadic patterns are similarly defined.

This paper assumes that different patterns contribute independently on the computation of confidence scores. Eq 2 can be revised to accommodate the dependency of different patterns and it shall be included in our future work.

Consider the common follower and cycle patterns in our example neighborhood network of u_t in Figure 1. The common follower pattern instances include $u_{s5} \leftarrow u_{i2} \rightarrow u_t$ and $u_{s1} \leftarrow u_{i1} \rightarrow u_t$. The instances $u_{s4} \leftarrow u_{i2} \rightarrow u_t$, $u_{s2} \leftarrow u_{i1} \rightarrow u_t$, and $u_{s3} \leftarrow u_{i1} \rightarrow u_t$ are not common follower instance but they are the instances of the pre-condition of common follower pattern. The target specific support and confidence of this pattern are:

$$Sup(p_{cfr}(u_t, u_{i1})) = 1, Sup(p_{cfr}(u_t, u_{i2})) = 1$$
$$Sup(p_{cyc}(u_t, u_{i1})) = 1$$

$$Conf(p_{cfr}(u_t, u_{i1})) = \frac{1}{3}, Conf(p_{cfr}(u_t, u_{i2})) = \frac{1}{2}$$
$$Conf(p_{cyc}(u_t, u_{i1})) = \frac{1}{2}$$

The target specific support and confidence of other patterns can be derived in a similar manner.

4.2 Follow Link Seeking Strategy

Single action strategy. Once we have the target specific pattern confidence of the target user, we determine the likelihood of different actions causing the formation of $u_s \leftarrow u_t$ link. By composing these user actions in some order, we derive a follow link seek strategy.

A user action a, whether in the form of $u_s \rightarrow u_t$ or some $u_s \rightarrow u_i$, can be treated as a new follow link added to the neighborhood network of u_t, i.e., $G_{u_t} \cup a$. With this addition, we may find links that satisfy the pre-condition of one or more follow link pattern each involving no intermediate user (in the case of reciprocity pattern), one or more intermediate user (for triadic patterns). The more pre-condition instances are found between u_s and u_t, we would expect the likelihood of creating the $u_s \leftarrow u_t$ link to be higher.

As we have five patterns rcp, trt, cfr, cfe and cyc, we will have five sets of pre-condition instances involving the same u_s and u_t users, but different intermediate users. Let $I^q(u_t, u_s, p, a)$ denote the set of pre-condition instances for pattern p in the u_t's neighborhood network with the additional user action link a. $I^q(u_t, u_s, p, a)$ can possibly be empty. The likelihood of a user action a creating the $u_s \leftarrow u_t$ is therefore:

$$Pr(u_s \leftarrow u_t|a) = 1 - \prod_{u_i} \prod_{I^q(u_t,u_s,p,a)\neq\phi} (1 - Conf(p(u_t, u_i))) \qquad (3)$$

For example, in Figure 1, the likelihood of the user action $a_1 = u_s \rightarrow u_{i1}$ creating the $u_s \leftarrow u_t$ link is computed as:

$$\begin{aligned} Pr(u_s \leftarrow u_t|a_1) &= 1 - (1 - Conf(p_{cyc}(u_t, u_{i1}))) \\ &\quad (1 - Conf(p_{cyc}(u_t, u_{i2}))) \\ &= 1 - (1 - \frac{1}{2})(1 - 0) = 0.5 \end{aligned}$$

Suppose u_s decides to follow u_t directly instead and let this action be a_2. Unfortunately, as u_t has not adopted reciprocity pattern in following others (i.e, $Conf(p_{rcp}(u_t)) = 0$) and a_2 does not create any new pre-condition instances for other triadic patterns. Hence, $Conf(p_{rcp}(u_i)) = 0$.

Suppose $u_s \leftarrow u_{i1}$ is a link that exists in the network, and we consider action a_1 by u_s again. $Conf(p_{cfr}(u_t, u_{i1}))$ will be revised as $\frac{1}{4}$ due to an additional pre-condition instance which exists before a_1. The likelihood of creating the $u_s \leftarrow u_t$ link is now:

$$\begin{aligned} Pr'(u_s \leftarrow u_t|a_1) &= 1 - (1 - Conf(p_{cyc}(u_t, u_{i1}))) \\ &\quad (1 - Conf(p_{cyc}(u_t, u_{i2}))) \\ &\quad (1 - Conf(p_{cfr}(u_t, u_{i1}))) \\ &= 1 - (1 - \frac{1}{2})(1 - 0)(1 - \frac{1}{4}) \\ &= \frac{5}{8} \end{aligned}$$

The new higher likelihood is reasonable ($Pr'(u_s \leftarrow u_t|a_1) > Pr(u_s \leftarrow u_t|a_1)$) given that there are more pre-condition instances satisfied by the existing links and a_1.

So far, we only examine single action strategy that requires u_s to create a new follow action. The time complexity here is governed largely by the size of G_{u_t}. The follow action considered here is one that intentionally creates some additional pre-condition instances for the three patterns rcp, cfe and cyc. u_s is not able to create additional pre-condition instances for patterns trt and cfr as the pre-condition instances of these two patterns require some u_i to form link to u_s, which u_s has not direct control over. Interestingly, we can view this as yet another instance of follow link seeking problem, and apply our pattern based approach recursively. This also lead to our generalized strategy that consists of multiple actions.

Multi-action strategy. Our multi-action strategy is derived by a recursive application of the single action strategy to seek a series of new follow links that ends with $u_s \leftarrow u_t$. As mentioned in the single action strategy, u_s has to seek a follow link from an intermediate user u_i in order to create additional pre-condition instances for trt and cfr patterns. To solicit the $u_s \leftarrow u_i$ link, u_s can consider taking some follow link action that creates the pre-condition instances of some patterns that help to create $u_s \leftarrow u_i$. Here, we have u_i taking the place of u_t. The difference here is that there can be different u_i's to be considered.

For these u_i's, we have to start crawling the neighborhood network of u_i (i.e., G_{u_i}) and compute their target specific pattern confidence values, i.e., $Conf(p_{rcp}(u_i))$, $Conf(p_{trt}(u_i))$, etc.. We then derive for each candidate user action a, the likelihood of $u_s \leftarrow u_i$ is created, i.e., $Pr(u_s \leftarrow u_i|a)$.

Consider that u_s in Figure 1 wants to create $u_s \leftarrow u_{i1}$ as the first step of the strategy. A direct follow action $a_0 = u_s \rightarrow u_{i1}$ will have the likelihood of 1 as shown below:

$$\begin{aligned} Pr(u_s \leftarrow u_i|a_0) &= 1 - (1 - Conf(p_{rcp}(u_t, u_{i1}))) \\ &= 1 - (1 - 1) = 1 \end{aligned}$$

Hence, we now have a multi-action strategy $\langle a_0, a_1 \rangle$ which has the combined likelihood of:

$$\begin{aligned} Pr(u_s \leftarrow u_t|\langle a_0, a_1 \rangle) &= Pr(u_s \leftarrow u_i|a_0)Pr'(u_s \leftarrow u_t|a_1) \\ &= \frac{5}{8} \end{aligned}$$

Depending on the size of G_{u_i}, the time complexity will have include the computation of target-specific pattern confidence and choice of user actions.

5. EXPERIMENTS

In this section, we describe the experiments that are conducted to evaluate the effectiveness of our proposed follow link seeking method. Ideally, these experiments should be conducted directly with the Twitter users but serious practical issues such as getting source users willing to participate in the experiments had prevented us from doing so. Instead, our experiments focus on evaluating the accuracy of predicting which source users to follow by a target user given that the source users are connected to the target user by some pre-condition patterns. If the prediction accuracy is high using the likelihood values generated in our proposed method, it will indirectly imply the effectiveness of our proposed follow link seeking method which is heavily based on the follow link patterns and the likelihood function.

5.1 Dataset

We conducted our experiments on a Twitter dataset consisting of follow links of 151,128 users who declared to be in Singapore in their profile pages, and other users they follow or following them. The dataset was crawled at regular intervals over the period from January 8, 2012 to April 3, 2012 (87 days). Multiple crawls were needed because Twitter API does not release the timestamp information of follow links. While we still did not have the exact timestamps of the follow links, the timestamp of a data crawl was assigned to all the new follow links found at the crawl. These assigned timestamps can permit at least partial ordering of follow links in our dataset.

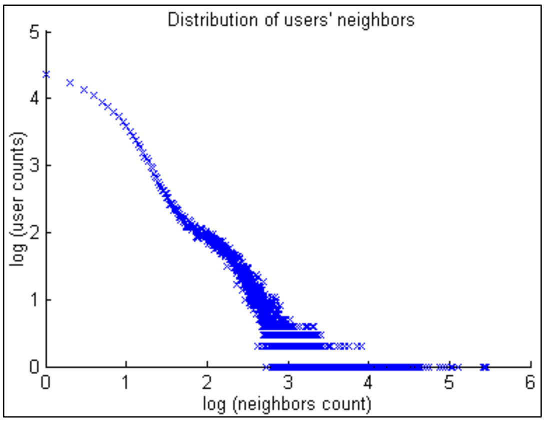


Figure 2: Distribution of Singapore users' neighborhood

Number of crawls/timestamps		2887
Number of Singapore users		151,128
Number of users		3,754,408
Number of links		14,465,305
Followees	Minimum	0
	Maximum	157,382
	Average	48.76
Followers	Minimum	0
	Maximum	286,738
	Average	58.50
Neighbors	Minimum	0
	Maximum	287,549
	Average	81.90
After Preprocessing		
Number of Singapore users		130,708
Number of users		1,425,505
Number of links		5,703,030

Table 2: Data Statistics

The statistics of this dataset is summarized in Table 2. We also show the degree distribution of users in Figure 2. Table 2 shows that we have an average of 2887/87 = 33 crawls per day. Note that each crawl only downloaded a subset of users' follow links as we crawled using multiple machines. The table also shows that there are 3.8M total number of users including users who are neighbors of Singapore users, and 14.5M links among these 3.8M users.

To ensure the representativeness of users, we further filtered peculiar users from our dataset. We removed users who satisfy any of the criteria below:

- *Users without neighbors.* As our method depends largely on the neighborhood networks of users to learn their follow link behaviors, users without neighbors do not have such observed follow link behaviors and should be excluded.

- *Users with too many neighbors.* Figure 2 shows the degree distribution of users is very skewed toward zero or one follower or followee. There are very few users who have huge number of neighbors (followers and followees). We consider these users outliers and they may not behave like other general users. To prevent our experiments to be biased by these users, we removed the top 1% of users ranked by degree and their follow links.

Neighbor count	Number of users
1 - 377	127,901
378 - 755	2,199
756 - 1,133	608

Table 3: Distribution of Singapore users in equal-width bins

After the above preprocessing, we have 130,708 Singapore users having the degrees between 1 and 1,133 as shown in Table 2. The total number of users including the neighbors of these Singapore users is **1,425,505**. There are 5.7M follow links among them.

Among the Singapore users, we would like to select a subset of them as our target users. As shown in Figure 2, the degree distribution of Singapore users is very skewed towards very small values. Instead of sampling mostly users with small degree, we stratified the Singapore users into three equal-width bins as shown in Table 3. We randomly picked 500 users from each bin to be our target users. Each target user must also satisfy all the following criteria:

- Follower count + Followee count ≥ 10, Follower count > 0, and Followee count > 0: This is to ensure that we have enough information to learn user's behavior in follow link formation.

- Public user account: This ensures that we can extract the neighborhood network of every target user.

Finally, we selected $N = 1{,}500$ Singapore Twitter users as the target users for our experiments.

5.2 Evaluation Procedure

Candidate source user selection. For each target user, a set of candidate source users was selected for the target user to follow. A candidate source user has to satisfy all following criteria:

- The source user should have pre-condition links of any pattern(s) with the target user.

- The source user has not been followed by the target user in the training data, which will be elaborated shortly.

We set aside the latest 10% followees who are also candidate source users of every target user as our test data, and the rest for training. The training data is used to compute the target specific pattern confidence scores and the likelihood of generating a $u_s \leftarrow u_t$ follow link.

We would like to predict which candidate source users are to be followed by each target user in the test data. The set of candidate source users actually followed by u_t in the test data is denoted by $H(u_t)$. Due to time constraints, we only retrieved a maximum of 15,000 candidate source users for each target user (3,000 for each pattern).

Metrics. For each target user u_t, we computed the likelihood of u_t following each candidate source user using our proposed method. We then ranked the candidate source users having pre-condition links with u_t (denoted by $H_{cand}(u_t)$) by likelihood score. Let $H_{\langle method \rangle}(u_t, k)$ denote the set of

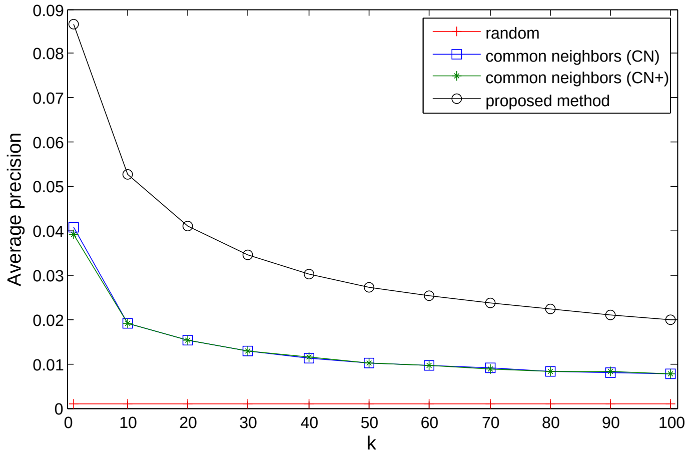


Figure 3: Average Precision@k

top k ranked candidate source users for u_t by some method. We measure the accuracy of prediction of the method by **Precision at** k (**P@k**) of u_t as follows:

$$P@k(u_t) = \frac{|H_{\langle method \rangle}(u_t,k) \cap H(u_t)|}{k} \qquad (4)$$

The **Average Precision at** k ($\overline{P@k}$) of all target users is therefore:

$$\overline{P@k} = \frac{\sum_{u_t} P@k(u_t)}{N} \qquad (5)$$

Baseline methods. Three methods were used as baselines, namely the RANDOM and two COMMON NEIGHBOR methods. The RANDOM baseline method randomly selects k candidate source users from $H_{cand}(u_t)$ for each target user u_t. The expected average precision at k for RANDOM is therefore $\frac{1}{N}\sum_{u_t}\frac{H(u_t)}{H_{cand}(u_t,t)}$, which is independent of k.

There are two COMMON NEIGHBOR methods which do not make use of the follow link direction. The first version CN ranks candidate source users based on the number of common neighbors between the candidate source users and the target user. The second version of the common neighbor method, CN+, considers the direct source user-to-target user link (if exists) as an additional common neighbor.

5.3 Results

Other than RANDOM, we performed three different runs of other baseline and our proposed methods. Each run comes with a different sample set of target users and the average performance over all runs was reported. This also gives rise to a more stable result. Figure 3 shows the average precision at k ($\overline{P@k}$) of all methods using different k values, from 1 to 100.

We make the following observations in our results:

- Our proposed pattern-based method performs better than the three baseline methods, while the common neighbor methods CN and CN+ outperform the RANDOM. Both CN and CN+ have very similar average precision. Our method achieves average precision between 20 to 87 times than that of RANDOM for different k values.
- The best average precision is achieved when $k = 1$. The average precision of all methods except RANDOM, demonstrate a decreasing trend as we increase k.

The above results show that our proposed pattern-based method is better than the common neighbor based methods in predicting the source users to be followed by a target user. With a better prediction accuracy, we believe that the strategy recommending actions to the source users will perform better than a strategy using common neighbor method for recommending source user actions.

6. VISUALIZATION

In this section, we describe the visualization of follow link seeking strategy recommendation in a web-based graphical user interface system called FRIENDER[1]. The objective of this system is to allow users to specify a target user from whom follow link is to be sought. FRIENDER generates a step-by-step recommended user action to be adopted by the source user, making it easier to observe the recommended strategies. To simplify user interaction further, FRIENDER provides a visual walkthrough of user actions within a strategy relating them to the follow link patterns.

At the backend, FRIENDER crawls the target user's neighborhood network on-the-fly when the target user is given by the user. This design has both advantages and disadvantages. The advantages include (i) no special storage for pre-crawled dataset; (ii) latest follow patterns are analyzed; and (iii) no restriction on which target and source users to use.

On the other hand, a single crawl on-the-fly prevents us to obtain the time order information about the follow links. Hence, we have to relax the restriction of the source-to-target follow link coming after the pre-condition links. Without the restriction, the confidences of the patterns become weaker.

FRIENDER's main interface is shown in Figure 4. The interface is divided into five main panels.

1. *Input* panel allows users to input the desirable target's screen name, source's screen name, and to select the desired patterns. If no source's screen name is provided, we assume that the source user is a completely new user not connected to any existing users.

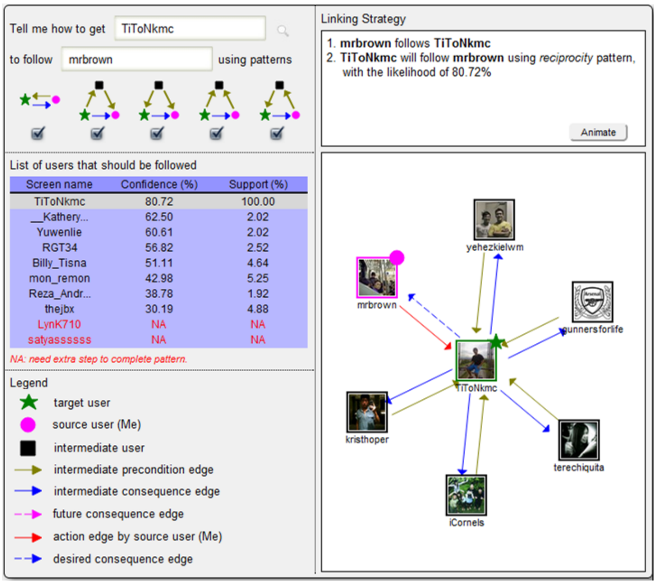
2. *Result* panel shows the top 10 follow actions that a source user can take to seek the follow link from the target user and they are ordered by confidence score. Each follow action is associated with a user (target or intermediate) that the source node needs to follow and the related pattern which has a support score. By following one of these users, there is a likelihood that target user will follow the source user.

3. *Legend* panel explains the symbols used.

4. *Instruction* panel lists the series of actions the user should perform and the expected outcome, together with the likelihood scores.

5. *Neighborhood* panel shows the source and target users, and other users involved in the neighborhood networks of the target users.

[1] http://research.larc.smu.edu.sg/palanteer/friender/index.php



Tell me how to get TiToNkmc

to follow mrbrown using patterns

☑ ☑ ☑ ☑ ☑

List of users that should be followed

Screen name	Confidence (%)	Support (%)
TiToNkmc	80.72	100.00
__Kathery...	62.50	2.02
Yuwenlie	60.61	2.02
RGT34	56.82	2.52
Billy_Tisna	51.11	4.64
mon_remon	42.98	5.25
Reza_Andr...	38.78	1.92
thejbx	30.19	4.88
LynK710	NA	NA
satyasssss	NA	NA

NA: need extra step to complete pattern.

Legend

- ★ target user
- ● source user (Me)
- ■ intermediate user
- → intermediate precondition edge
- → intermediate consequence edge
- - → future consequence edge
- → action edge by source user (Me)
- - → desired consequence edge

Linking Strategy

1. **mrbrown** follows **TiToNkmc**
2. **TiToNkmc** will follow **mrbrown** using *reciprocity* pattern, with the likelihood of 80.72%

Figure 4: Friender's user interface

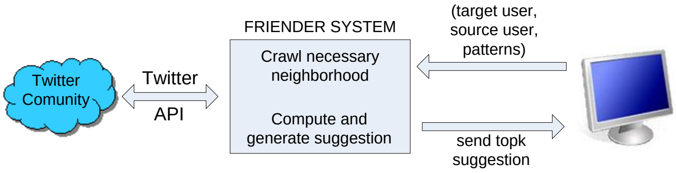
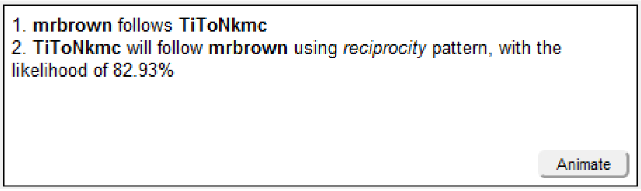

FRIENDER SYSTEM
Crawl necessary neighborhood
Compute and generate suggestion
Twitter Comunity
Twitter
API
(target user, source user, patterns)
send topk suggestion

Figure 5: Friender system

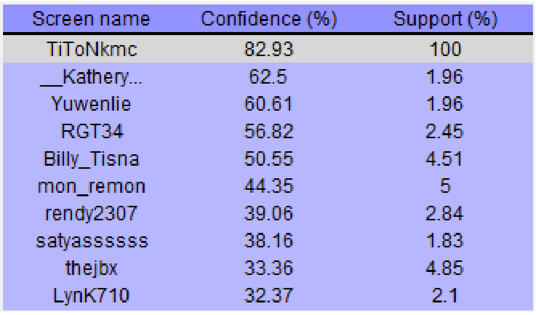
As shown in Figure 5, FRIENDER first takes input from users, such as target user's screen name, source user's screen name, and a set of follow link patterns to be considered. At least one pattern is needed for the system to suggest some strategies. FRIENDER then communicates with Twitter API to crawl the neighborhood network of the target user for confidence computation. FRIENDER generates top k actions contributing to the formation of follow link from the target user to the source user.

Due to the restriction on number of concurrent connections to Twitter API by a single browser[2] and also the time taken to perform crawling of Twitter data, we limit FRIENDER to crawl maximum of 200 neighbors for a given target user. If a target user has more neighbors, only the latest 100 followees and 100 followers are crawled (as crawling followees and followers require different APIs).



1. **mrbrown** follows **TiToNkmc**
2. **TiToNkmc** will follow **mrbrown** using *reciprocity* pattern, with the likelihood of 82.93%

Figure 6: User actions for `mrbrown` to follow `TiToNkmc`



Screen name	Confidence (%)	Support (%)
TiToNkmc	82.93	100
__Kathery...	62.5	1.96
Yuwenlie	60.61	1.96
RGT34	56.82	2.45
Billy_Tisna	50.55	4.51
mon_remon	44.35	5
rendy2307	39.06	2.84
satyasssss	38.16	1.83
thejbx	33.36	4.85
LynK710	32.37	2.1

Figure 7: Suggested User Actions

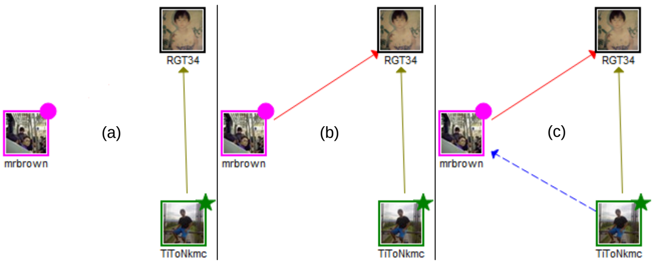


Figure 8: Sequence of follow actions between `TiToNkmc` and `mrbrown` using common followee pattern

Figure 7 shows several suggested actions to be performed by `mrbrown` to gain a follow link from `TiToNkmc`. The top action suggested is for `mrbrown` to follow `TiToNkmc`, our target user as shown in Figure 6. This action comes with a confidence of 82.93% to gain the required follow link. This action is ranked highly because of `TiToNkmc`'s more active use of reciprocity pattern than any other pattern. On average, out of 100 users who follow `TiToNkmc`, the latter follows back 83 of them. This suggested action is also the simplest as it does not involve any intermediate user.

The other suggested actions require at least one intermediate user to gain the follow link from`TiToNkmc`. For example, the second suggested action requires `mrbrown` to follow `RGT34`. By following `RGT34`, `mrbrown` satisfies the precondition of common followee pattern. Therefore, there is a 62.5% chance that `TiToNkmc` will follow him back. The sequence of follow actions of involving these three users using common followee pattern is animated by FRIENDER as shown in Figure 8.

Now, we show an example using multiple actions. At the beginning, we have three users, `mrbrown`,`TiToNkmc` and `thejbx` with `TiToNkmc` following `thejbx` (See Figure 9(a)). Suppose `mrbrown` decides to gain a follow link from `TiToNkmc` using transitivity pattern through the user `thejbx`. This is only possible if there is a follow link from `thejbx` to `mrbrown`. At this point, FRIENDER recommends `mrbrown` to first follow another intermediate user `via_louhan` who is a follower of `thejbx` (See Figure 9(b)). Using the cycle pattern, `thejbx` will follow `mrbrown` with a likelihood of 55.54% (See Figure 9(c)). Afterwards, `TiToNkmc` will follow `mrbrown` using transitivity with an overall likelihood of 33.36% (= 55.54% × 56.99%) (See Figure 9(d)).

7. CONCLUSION

This paper proposes a follow link seeking method to recommend strategies to be performed by a source user so as

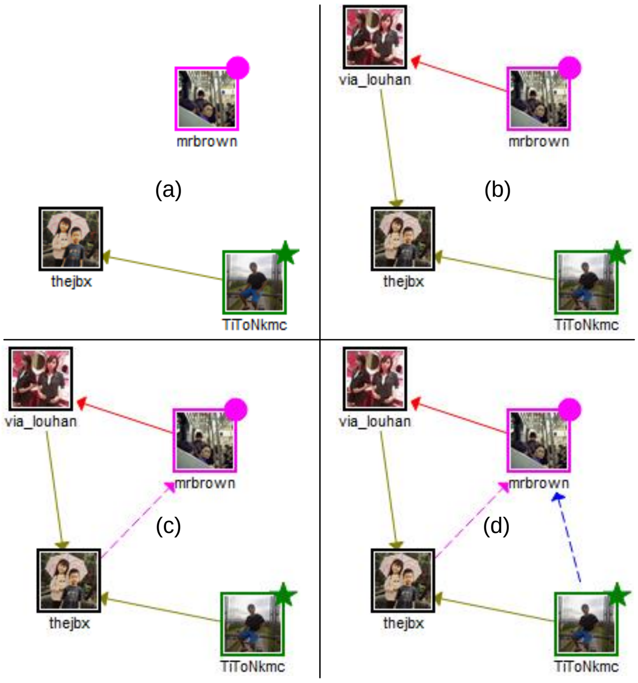


Figure 9: Sequence of follow actions - Multi-action strategy

to increase the chance of a target user following him. Our method utilizes well known follow link patterns that summarize the behaviors of users following other users. Having analyzed the target user's past behavior in adopting these patterns, our method generate a series of user actions to the source user each assigned with a likelihood value. We have conducted experiments to show that our method performs better than baseline methods. We further developed a visualization tool called FRIENDER to guide users visually using our method.

There are several possible future research works to pursue. Our method can be easily extended to consider tweet content, topics and user profile information. We also plan to study other type of user actions such as retweet and mention. We believe that the richer set of user actions will help to further improve the accuracy of the method. We also plan to study the effect of using regression and smoothing in computing confidence score. Finally, a user study involving a small set of Twitter users could be carried out to further validate the method.

8. ACKNOWLEDGEMENT

This work is supported by Singapore's National Research Foundation's research grant, NRF2008IDM-IDM004-036.

9. REFERENCES

[1] Aeration bin laden death faster than traditional media first twitter 20 minutes. `http://att-smartphone.blogspot.com/2011/05/aeration-bin-laden-death-faster-than.html`. [Online; accessed March 30, 2012].

[2] Overview of browser concurrent connection limits. `http://meronymy.blogspot.com/2011/09/overview-of-browser-concurrent.html`. [Online; accessed April 9, 2012].

[3] L. A. Adamic and E. Adar. Friends and neighbors on the web. *Social Networks*, pages 211–230, 2003.

[4] D. Boyd, S. Golder, and G. Lotan. Tweet, tweet, retweet: Conversational aspects of retweeting on twitter. In *HICSS*, pages 1–10. IEEE Computer Society, 2010.

[5] M. Brzozowski and D. Romero. Who should i follow? recommending people in directed social networks. 2011.

[6] S. A. Golder and S. Yardi. Structural predictors of tie formation in twitter: Transitivity and mutuality. In *Proceedings of the 2010 IEEE Second International Conference on Social Computing*, pages 88–95, Washington, DC, USA, 2010. IEEE Computer Society.

[7] L. Katz. A new status index derived from sociometric analysis. *Psychometrika*, 18:39–43, 1953.

[8] R. N. Lichtenwalter, J. T. Lussier, and N. V. Chawla. New perspectives and methods in link prediction. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '10, pages 243–252, New York, NY, USA, 2010. ACM.

[9] V.-A. Nguyen, C. W.-K. Leung, and E.-P. Lim. Modeling link formation behaviors in dynamic social networks. In *Proceedings of the 4th international conference on Social computing, behavioral-cultural modeling and prediction*, SBP'11, pages 349–357, Berlin, Heidelberg, 2011. Springer-Verlag.

[10] D. M. Romero and J. Kleinberg. The directed closure process in hybrid social-information networks, with an analysis of link formation on twitter. In *ICWSM*. The AAAI Press, 2010.

[11] D. M. Romero and J. M. Kleinberg. The directed closure process in hybrid social-information networks, with an analysis of link formation on twitter. In *ICWSM*, 2010.

[12] D. Yin, L. Hong, and B. D. Davison. Structural link analysis and prediction in microblogs. In *CIKM*, pages 1163–1168. ACM, 2011.

[13] D. Yin, L. Hong, X. Xiong, and B. D. Davison. Link formation analysis in microblogs. In *SIGIR*, pages 1235–1236. ACM, 2011.