Prediction in a Microblog Hybrid Network Using Bonacich Potential

Shanchan Wu
HP Labs Palo Alto
1501 Page Mill Road, Palo Alto, CA, 94304
shanchan.wu@hp.com

Louiqa Raschid
University of Maryland Institute for Advanced Computer Studies College Park, MD 20742
louiqa@umiacs.umd.edu

ABSTRACT

Microblogs such as Twitter support a rich variety of user interactions using hashtags, urls, retweets and mentions. Microblogs are an exemplar of a hybrid network; there is an explicit network of followers, as well as an implicit network of users who retweet other users, and users who mention other users. These networks are important proxies for influence. In this paper, we develop a comprehensive behavioral model of an individual user and her interactions with other users, and users who mention other users. These networks are important proxies for influence. In this paper, we develop a comprehensive behavioral model of an individual user and her interactions in the hybrid network. We choose a focal user and predict those users who will be influenced by her, and will retweet and/or mention the focal user, in the near future. We define a potential function, based on a hybrid network, which reflects the likelihood of a candidate user being influenced by, and having a specific type of link to, a focal user, in the future. We show that the potential function based prediction model converges to the Bonacich centrality metric. We develop a fast unsupervised solution which approximates the future hybrid network and the future Bonacich potential. We perform an extensive evaluation over a microblog network and a stream of tweets from Twitter. Our solution outperforms several baseline methods including ones based on singular value decomposition (SVD) and a supervised Ranking SVM.

1. INTRODUCTION

Microblogs such as Twitter support a rich variety of user interactions. One can follow a user and read her tweets. One can search for keywords or hashtags or follow trending tweets. A user can initiate a new topic by creating a new tweet or hashtag, often including a url in the tweet to refer to more detailed articles. One can interact with another user by mentioning them. One can also participate in the diffusion of a topic by retweeting. All of these interactions create a dynamic and rich social network for diffusion of information and to establish the influence of a user. In addition to the rich communication platform, microblogs have the advantage that it is simple to monitor the media stream due to the brevity of microblogs. Hashtags and urls enhance the stream with richer content and links. More important, diffusion can be easily monitored through retweets and mentions and both retweets and mentions have been identified as an important proxy for influence [4].

All the above factors increase the importance of monitoring social media. Our motivation for this research is to develop models and tools to exploit the power of this medium, as follows: A first step is monitoring to build behavioral models at the level of the individual user and her interactions. The next step is developing accurate prediction models of future actions and interactions in social media streams. The final step is to make personalized recommendations that combine the individual level behavioral model with predictions of future social media interactions. Our objective is thus to develop a behavioral model that can capture the richness and complexity of microblogs. We use this to make predictions about the future at the level of an individual focal user, i.e., given a focal user, we want to predict the other users who will be influenced by her and will interact with her.

An example application for this research is monitoring for personalized and interactive brand management. A brand manager has the objective of monitoring conversations about a brand, to track relevant topics and sentiment, and to identify potentially negative conversations. While aggregate statistics, e.g., a trending topic about the brand, or an increase in negative sentiment, is important, social media also provides a platform for personalized and interactive brand management.

Suppose a brand manager knows which user $u$ is likely to talk about her brand, either in a positive or negative way. It will be useful if she could determine if $u$ is influential, as well as other users $v$ who will be influenced by $u$. With this knowledge, the brand manager could perhaps take a proactive action such as engaging in a conversation with $u$.

However, a user $u$ who is an expert on a brand (topic) and is influential may not be willing or interested in engaging directly with a brand manager. The brand manager may then decide to engage with other users $v$ who are in turn influenced by $u$. The richness of social media also allows one to make diverse recommendations, e.g., to target yet another user $w$ who has not previously tweeted about the brand but who has several friends with an interest in the brand and who have engaged in creating or diffusing relevant content.

There are many diffusion and influence models for social networks. For example, the Linear Threshold Model and the Independent Cascade Model have been widely studied [8, 15]. For the Linear Threshold Model, in each step, a user will be activated (influenced) if the total weight of her active neighbors is greater than a threshold. For the Independent Cascade Model, each active user has a single opportunity with some probability to activate each of her inactive neighbors.

These prior models have limitations when applied to microblogs. One limitation is that these prior models capture behavior at the aggregate level, e.g., at the level of a topic [24]. One popular aggregate level influence challenge is the influence maximization problem. It
was first formulated as a discrete optimization problem in [15] and has been studied by others [5]. The target of influence maximization is to select an initial set of users who eventually influence the largest number of people in the network. Predicting the degree of influence has also been studied; for example, a regression model was used to predict the influence of a user [1].

Another limitation is that both the Linear Threshold Model and the Independent Cascade Model typically assume that one can only influence her immediate neighbors. The definition of neighbors is that there are edges in the network between these users. The edges are concrete and observable. For example, in a friendship network, one can only influence her friends. In a disease spread network, some diseases can only spread through direct contact with a user. Our observation for microblogs is that influence is not limited to the immediate neighborhood, i.e., influence can spread outside the friendship network of microblogs. One can retweet or mention users who are not one’s friends. For example, in our experimental dataset (described in Section 5), more than 40% of the mentions are from outside the Follower network.

Most other research consider diffusion or prediction for a single homogeneous network. However, Twitter is an example of a hybrid network composed of multiple networks. There is an explicit Follower network. Retweet actions and mention actions also reflect key interactions between users; hence, Retweet and Mention networks can be constructed. Prior research typically considered the evolution of a single network, typically the equivalent of the Follower network of Twitter. However, given a hybrid network of microblogs, we expect all three networks to evolve as a result of the influence of the user. For example, if $u$ is retweeted a lot, she may attract additional followers $f$, and that in turn may lead to even more retweets and mentions from the followers of each $f$ who recently joined the Follower network of $u$. This is an example of the Retweet network causing an evolution of the Follower network, which in turn results in an evolution of the Retweet and Mention networks, respectively. We further note that our prediction method relies primarily on network features. The evolution of the three networks makes predicting the future network, while using network based features, more challenging.

To summarize, microblogs exhibit complex user interactions over a hybrid network. Influence is not limited to the immediate followers and it can be measured through the characteristics of the three networks. Further, the impact of influence may result in the evolution of all three networks. We wish to develop a prediction model at the level of the individual user, to predict those users who are most influenced by the focal user, and will retweet or mention the focal user, in the future.

Our contributions approach can be summarized as follows:

- We define a hybrid network made up of a Follower Retweet and Mention network. We define two prediction problems at the level of the individual focal user: the future retweet prediction and the future mention prediction.
- We define a potential function over the hybrid network that reflects the likelihood of a candidate user being influenced by, and having a specific type of link in the future to a focal user.
- We show that the potential function converges to the Bonacich centrality metric in the hybrid network. This is consistent with prior research that showed this metric was a good proxy for influence.
- We develop a fast unsupervised solution that approximates the future hybrid network and the future Bonacich potential.

We perform an extensive evaluation over a microblog network and a stream of tweets from Twitter. Our model can provide high quality personalized recommendations of future interactions, at the level of individual behavioral interactions, with a surprisingly high degree of accuracy.

- We outperform several baseline methods including one based on singular value decomposition (SVD) and a supervised Ranking SVM based method.

The rest of the paper is organized as follows: Section 2 discusses related work; Section 3 formally formulates the problem and proposes the prediction model and solution; Section 4 explains how to choosing parameters; 5 describes the evaluation dataset and metrics and data properties; Section 6 presents evaluation results on our methods. Finally, Section 7 concludes our work.

2. RELATED WORK

Cha et al. in [4] have studied measuring user influence in a microblog, i.e., Twitter. They found that retweets and mentions are more important than the Follower network to determine influence. Predicting the most influential users in a microblog has been addressed in [23]. Beyond microblogs, future link prediction in the blogosphere has been addressed in [27]. Gomez-Rodriguez et al. in [9] proposed an approach to infer networks of diffusion and influence. They tried to reconstruct the network over which contagions propagate. So their target is to build the historical network where there is rich information in the history when the network was built. Our target is to predict future relationship where the information in the future is unknown when the relationship will be built.

Hong et al. in [11] proposed Matrix Co-Factorization to model user interests and to predict individual decisions in Twitter. Given a specific tweet, they predict if a focal user will retweet it. Peng et al. in [22] used Conditional Random Fields and Yang et al. in [28] proposed a semi-supervised framework on a factor graph model to solve a similar problem. While this problem is related to our problem of retweet prediction, their problem targets a specific tweet, where the content of the tweet is known a priori. Thus, content based features plan a very important in their prediction. Our prediction task is targeted on a specific user, where the contents of his/her future tweets is not known. Our research has to use both content and network based features. We note that the supervised learning approach of Hong et al. [11] has good accuracy and we compare the datasets and evaluation results in a later section.

Retweet and mention prediction can be formalized as a link prediction problem. Link prediction has been studied in various applications including social networks, relational datasets, labeled entity-relationship graphs, etc. An array of topological methods for link prediction were studied by Liben-Nowell and Kleinberg [19] who evaluated them on co-authorship networks. Machine learning approaches have also been applied to link prediction. Solution approaches include spectral transformation [17], the heat diffusion kernel [12], Markov Random Field Model [26], collective classification [25] and ranking SVM [27] etc. We note that previous link prediction research has not considered a hybrid network, where there may be different types of links between two nodes.

The most similar work on a prediction model based on a composite network [21] has been applied to predict the adoption of mobile apps. The authors collected different social networks using built-in sensors, including Bluetooth proximity network, call log network, etc. Their model is based on the assumption that the adoption decision is influenced by neighbors in the composite network. They solved an optimization problem to create the composite network; however, there are scalability challenges in applying their work.
3. PROBLEM FORMULATION AND SOLUTION

3.1 Problem Definition

DEFINITION 1. Future Retweet Prediction: Given a focal microblog user \( u \) at a specific time point \( T \), identify \( K \) microblog users \( S^m \) who will retweet one (or more) future tweets(s) of user \( u \) in the near future.

DEFINITION 2. Future Mention Prediction: Given a microblog user \( u \) at a specific time point \( T \), identify \( K \) microblog users \( S^m \) who will mention microblog user \( u \) one (or more) times in the near future.

3.2 Prediction Model

Our objective is to exploit historical knowledge and the corresponding hybrid network to accurately predict future links. Let \( G^1, \ldots, G^M \) represent the \( M \) relationship networks constructed using history; for Twitter \( M = 3 \) corresponding to the three networks: Follower, Mention and Retweet. The corresponding relationship networks in the future time period are denoted \( Y^1, \ldots, Y^M \). Our objective is to infer an optimal composite network \( H^c \) from \( G^1, \ldots, G^M \) to predict each \( Y^c \), \( 1 \leq c \leq M \). To be optimal, the hybrid network \( H^c \) should be customized to each network \( Y^c \).

Let \( G^m_{i,j} \) represent the weight associated with the edge from node \( j \) to node \( i \) in some network \( G^m \). Let \( H^c_{i,j} \) represent the weight associated with the edge from \( j \) to \( i \) in the hybrid network \( H^c \).

We define the hybrid network \( H^c \) for each \( Y^c \) as follows:

\[
H^c_{i,j} = \sum_m \omega^c_m G^m_{i,j} \quad \text{where} \forall m, \omega^c_m \geq 0. \tag{1}
\]

A potential function \( P^c_{x,i} \), defined over each hybrid network \( H^c \) reflects the likelihood of a candidate node \( i \) having a link of type \( c \) in the future to a focal node \( x \). It reflects if \( i \) is influenced by \( x \). We define this potential function \( P^c_{x,i} \) as follows:

\[
P^c_{x,i} = \alpha H^c_{x,i} + \beta \sum_j P^c_{x,j} H^c_{j,i} \quad \text{where} \alpha \geq 0, \beta \geq 0. \tag{2}
\]

For a focal node \( x \), we consider several factors contributing to the potential from node \( i \). The first factor is the weight of the hybrid network edge from node \( i \) to node \( x \). The second factor is the potential of the neighbors \( j \) of node \( i \) to focal node \( x \). The third factor is the weight of the hybrid network edges from node \( i \) to its neighbors \( j \). Figure 1 is the visualization of these factors towards the potential function.

We can finally define a conditional probability to determine whether node \( i \) will have a type \( c \) link to node \( x \) in the future, based on the potential of node \( i \). Similar to [21], we adopt an exponential probability distribution to determine this function. We define our conditional probability as follows:

\[
\text{Prob}(Y^c_{x,i} > 0 \mid G^m, 1 \leq m \leq M) = 1 - \exp(-s - P^c_{x,i}) \tag{3}
\]

\( Y^c_{x,i} \) is the weight of a future type \( c \) edge from node \( i \) to node \( x \). If the value is greater than zero, it means that such a future edge exists. In the right side of the equation, \( s \) is a parameter. Parameter \( s \) will not change the ranking of the probabilities, but it will help to adjust the probability values and make the values more meaningful, especially when a probability threshold needs to be picked up.

The exponential function of \( f(x) = \exp(-x) \) has the monotonic and concave properties and matches the recent research [3] which suggests that the probability of adoption increases at a decreasing rate with increasing external network signals [21].

We note that this is a linear mixture model for creating the hybrid network. There are potentially other models for creating the hybrid network. For example, we could add more terms such as \( \sum_m \omega^c_m G^m_{i,j} G^m_{j,i} \) to equation (1) to consider a quadratic model. Different models for the hybrid network will not affect the way we define a potential function. However, different models would have different complexity to determine an optimal set of parameter values. We choose the linear model since it straightforward to obtain the optimal set of parameter values. Further, the linear model captures the essence of the hybrid network.

**Convergence of our solution:**

In equation (2), \( H^c_{x,i} \) is the weight of the hybrid network edge from node \( i \) to node \( x \), \( H^c_{j,i} \) is the weight of the hybrid network edge from node \( i \) to node \( j \), and \( \alpha \) and \( \beta \) are two parameters. Here we have potential values on both sides of the equation. By solving a group of these equations, we can represent the potential by other factors and parameters rather than having potential variables on the right side. By applying equation (1) to equation (2), we have the following formula:

\[
P^c_{x,i} = \beta \sum_m \omega^c_m G^c_{x,i} + \alpha \sum_j P^c_{x,j} \sum_m \omega^c_m G^c_{j,i} \tag{4}
\]

By recursively replacing \( P^c_{x,j} \) with its expression in equation (4), we have the following expression of \( P^c_{x,i} \).
We define $F^*$ to be an adjusted follower network adjacency matrix reflecting the popularity factor of each follower. $F^*_{i,j}$ is a weighted value if user $j$ is a follower of user $i$; otherwise $F^*_{i,j}$ is equal to 0. If user $j$ is a follower of user $i$, the weighted value of $F^*_{i,j}$ is affected by the number of friends of that user $j$. We calculate $F^*_{i,j}$ as follows:

$$F^*_{i,j} = \frac{\mathcal{T}}{D_j}$$

$\mathcal{T}$ is the average of number of friends over all users; $D_j$ is the number of friends of user $j$. The intuition is that if a user $j$ has a lot of friends, then her attention will be divided among those friends, and she will pay less attention to user $u$. Consequently, user $u$ has a lower influence on user $j$ if $j$ has a lot of friends.

### 3.4 Solution Using the Hybrid Network: hybridBON

For this approximate solution, we first estimate the weights of different networks to create the hybrid (composite) network. We discuss choosing parameters for estimating weights in Section 4. After we create the composite network, for the focal user $x$, we can calculate the potential values of all candidate users according to equation (5), with respect to the likelihood of having type $c$ link to $x$ in the future. Since we already know the weights of different networks, we can simplify the equation (5) by using equation (1) as:

$$P'_{x,i} = \beta H'_{x,i} + \beta \alpha (H' \cdot H'')_{x,i} + \beta \alpha^2 (H' \cdot H' \cdot H'')_{x,i} + \cdots$$

The formula (6) has the same format as the Bonacich centrality [2].

Let $A$ be the adjacency matrix. Recall that each node in the link graph is a user. Suppose $T$ is the time when the prediction is to be made. Thus, we consider all historical links prior to $T$. We set the value of the element $A_{i,j}$ in the adjacency matrix $A$ is the number of direct links of some type pointing from node $j$ to node $i$ that exist in history before time $T$. Then, the value of $(A^n)_{i,j}$ is equal to the number of paths of length $n$ from node $j$ to node $i$ in history. Bonacich centrality is computed as follows:

$$C(\alpha, \beta) = (\beta A + \beta \alpha A \cdot A + \cdots + \beta \alpha^n A^{(n+1)} \cdots)$$

$$= \beta A (1 - \alpha A)^{(-1)}$$

This equation holds while $\alpha < 1/\mu$, where $\mu$ is the largest characteristic root of $A$ [6]. For $\alpha = \beta$, this measure reduces to the Katz score.

Unlike $A$ in the above equation, where the value of each element is the number of links from a node to another node, for matrix $H'$ of a composite network, the value of each element is a real number which represents the weight of the edge from a node to another node. Then, rather than being equal to the number of paths of length $n$ from node $j$ to node $i$, the value of $(H')_{i,j}$ can represent the weight of the path of length $n$ from node $j$ to node $i$. Let $P$ be the matrix of all potential values $\{P'_{x,i}\}$. According to formula (6), we can calculate the score matrix $P$ as following:

$$P = (\beta H^1 + \beta \alpha H^2 + \cdots + \beta \alpha^n H^{(n+1)} \cdots)$$

$$= \beta H (1 - \alpha H)^{(-1)}$$

In this equation, we refer to $H^* = H$ since we use the superscript $n$ in $H^n$ to refer to the power matrix expression for the matrix representation of each $H^c$. Then $P$ can be used for prediction, and we label this method as hybridBON.
4. CHOOSING PARAMETERS

For the approximation approach hybridBON, where we want to first create a composite network, \( H^* = r \cdot R + m \cdot M + f \cdot F^* \), we consider two factors for the weights \( r, m, f \). We first scale the matrices so that no matrix can dominate the others. We then use the ground truth from the training data to calibrate the influence of each network \( R, M, F^* \) with respect to retweet prediction and mention prediction.

4.1 Scale Factor

The scale of each of the matrices may be different, e.g., the distribution of the values. This can result in one matrix dominate another.

We define a scale factor \( \gamma \). Given a matrix \( B \), then \( \gamma(A, B) \) will scale matrix \( A \) with respect to \( B \). For retweet prediction, \( R \) is the matrix that has the greatest influence and is used as a standard. We use the summation of the weights (values) of \( R \), \( \sum R_{i,j} \), to determine \( \gamma \) for retweet prediction. The scale factors for the three different networks with respect to retweet prediction are formally calculated as follows:

\[
\gamma(R, R) = 1, \quad \gamma(M, R) = \frac{\sum R_{i,j}}{\sum M_{i,j}}, \quad \gamma(F^*, R) = \frac{\sum R_{i,j}}{\sum F_{i,j}^*}
\]

The meaning of using retweet network as standard is that there are some total number of retweets among the users in the training period. Scaling other network to this standard is to mimic the retweet relationship but with somewhat different distribution.

Similarly, the scale factors for the three different networks with respect to mention prediction are formally calculated as follows:

\[
\gamma(R, M) = \frac{\sum M_{i,j}}{\sum R_{i,j}}, \quad \gamma(M, M) = 1, \quad \gamma(F^*, M) = \frac{\sum M_{i,j}}{\sum F_{i,j}^*}
\]

4.2 Penalty Factor

We define penalty factors to lessen the weight of the networks which have lower prediction ability. Intuitively, for retweet prediction, the matrix \( R \) (which matches the retweet ground truth from training data) is the most important. If another network \( M \) or \( F^* \) deviates from \( R \) then a penalty factor must be imposed. We assign the penalty factors to other networks according to their correlation with the ground truth network in the training data, i.e., the network in the history with the same type of link to predict in the future.

We use average Spearman’s rank correlation coefficient as the metric of how two adjacency matrices are correlated. A high correlation means a low deviation. For two rank sets \( X \) and \( Y \), suppose \( x_i \) and \( y_i \) are the ranks of the values of \( X_i \) and \( Y_i \) in \( X \) and \( Y \) respectively, and the number of elements in \( X \) and \( Y \) are both \( n \), Spearman’s rank correlation coefficient is calculated as:

\[
\rho(X, Y) = 1 - \frac{6 \sum (x_i - y_i)^2}{n(n^2 - 1)}
\]

The closer \( \rho \) is to +1 or -1, the stronger the correlation is. A perfect positive correlation will have a \( \rho \) value +1 and a perfect negative correlation will have a \( \rho \) value -1. We only consider the positive correlation. The penalty factor should be related to the correlation. The lower the correlation, the penalty will be stronger, i.e., the penalty factor should be lower. We define a penalty factor to be a function with respect to \( \rho \) as \( \rho^\epsilon \). Figure 3 shows the curves of \( \rho^\epsilon \) with respect to different \( \epsilon \) values. When \( \rho \) is in the range of \( (0, +1] \), \( \rho^\epsilon \) will also be in the range of \( (0, +1] \). When \( \epsilon \) is greater than 1, \( \rho^\epsilon \) has lower value than \( \rho \). The bigger value of \( \epsilon \), the stronger penalty is applied with the same correlation value.

The penalty factors for the three different networks with respect to retweet prediction are formally calculated as follows:

\[
\varphi(R, R) = 1, \quad \varphi(M, R) = \left\{ \frac{1}{N} \sum \rho(M_i, R_i) \right\}^\epsilon, \quad \varphi(F^*, R) = \left\{ \frac{1}{N} \sum \rho(F^*_i, R_i) \right\}^\epsilon
\]

where \( \rho(M_i, R_i) \) is the Spearman’s rank correlation of the \( i \)th row of \( M \) and the \( i \)th row of \( R \). \( \rho(F^*_i, R_i) \) is the Spearman’s rank correlation of the \( i \)th row of \( F^* \) and the \( i \)th row of \( R \), and \( N \) is the number of rows.

Similarly, the penalty factors for the three different networks with respect to mention prediction are as follows:

\[
\varphi(R, M) = \left\{ \frac{1}{N} \sum \rho(R_i, M_i) \right\}^\epsilon, \quad \varphi(M, M) = 1, \quad \varphi(F^*, M) = \left\{ \frac{1}{N} \sum \rho(F^*_i, M_i) \right\}^\epsilon
\]

For two variable sets, to calculate the Spearman’s rank correlation coefficient, identical values (value duplicates) are assigned a rank equal to the average of their positions in the ascending order. By this way, even if most of the entries of two matrices are 0, the correlation of the two matrices will also be in the range of \( (0, +1] \) when \( \epsilon \) is greater than 1. Table 1 is an example to show the different rank values for the zero values for two variable sets.

4.3 Merging Parameters

Next we consider the merging parameters to create the composite network; they combine the scale factors and penalty factors. We
Table 1: Example of the rank value of tie values

<table>
<thead>
<tr>
<th>Variable $X_i$</th>
<th>Position in the ascending order</th>
<th>Rank $x_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>0</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>5</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Variable $Y_i$</th>
<th>Position in the ascending order</th>
<th>Rank $y_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>1.5</td>
</tr>
<tr>
<td>0</td>
<td>2</td>
<td>1.5</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>8</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>12</td>
<td>5</td>
<td>5</td>
</tr>
</tbody>
</table>

5. Evaluation Dataset and Metrics

5.1 Data Collection

There have been several successful efforts to construct a proxy graph that characterizes the structure of a real network [7, 18]. For this experiment, our objective was different. It was to construct a dataset that reflected a comprehensive history of user interaction and tweet content, over an extended period, for a significant number of active users, given the strict limitations imposed by the Twitter API. We constructed a network of 15,000 users, as well as all their follower (friend) associations within this subnetwork. In choosing this experiment, our objective was different. It was to construct a network that characterizes the structure of a real network [7, 18]. For mention prediction, the composite network $H^*$ is created by merging the hybrid networks as

$H^* = r \cdot R + m \cdot M + f \cdot F^*$

We used the Twitter streaming API to collect all tweets published by the 15K active users between April 15, 2011, and June 12, 2011. Retweeting is identified by the use of @username in tweets. Mentions are identified by @username in the tweet content, after excluding RT @username. We built up the retweet and mention network by extracting users being retweeted or mentioned in each tweet. Hashtags identified by #hashtags and URLs were also extracted. Since username and hashtag are case insensitive, we transformed all usernames and hashtags to lowercase.

Test Dataset and Training Dataset:

We used the first month data (from April 25th to May 25th) as the training dataset and we obtained the ground truth from the second month data (from May 26 to June 25) and used it as the test dataset. We picked the sets of microblog users who had ground truth in the test dataset for evaluation. 4257 users had retweet ground truth and 7296 users had mention ground truth. The average number of ground truth (retweeters) for the 4257 users is 4.56, and the average number of ground truth (mentions) for the 7296 users is 8.12.

For tuning the parameters, we selected those microblog users that contain ground truth from May 15 to May 25 as focal training users. The networks for training were collected in the preceding time interval from April 25 to May 15. There was no overlap between the training and testing time interval. Similarly there was no overlap in the time interval for the collection of training networks and to obtain the training ground truth.

5.2 Metrics and Parameters

Mean Average Precision (MAP) is widely used for evaluating ranking methods; it provides a single-figure measure of quality across recall levels. MAP has been shown to have especially good discrimination and stability [20] and we report on the values of MAP. We also use Normalized Discounted Cumulative Gain (NDCG) [10] for evaluation. NDCG is also a measure commonly used for evaluating the results of ranking methods. The NDCG value of a ranking list at position $i$ is calculated as:

$NDCG@i = Z_i \sum_{j=1}^{i} 2^r(j) - 1 \log(1 + j)$

where $r(j)$ is the rating for the $j$th item and $Z_i$ is a normalization constant. $Z_i$ is chosen so that the NDCG score for a perfect ranking is 1. In our experiments we measured NDCG at the positions of 5 and 10.

We report on a sensitivity analysis for the choice of parameters in the next section. Here we report on specific parameter choices as follows:
of networks. As discussed in section 4, we used the metric of average Spearman’s rank correlation coefficient to evaluate their correlation. For two network adjacency matrices $A$ and $B$, we calculated the Spearman’s rank correlation coefficients of the corresponding rows of the two matrices, and then used the average value to represent the correlation of the two matrices.

Table 3 reports on the average Spearman’s rank correlation of different types of networks to the retweet network constructed from the first 30-day training data. Table 4 reports on the average Spearman’s rank correlation of different types of networks to the mention network constructed from the first 30-day training data. From the two tables, we can see that a network adjacency matrix has the highest correlation to itself and the matrices of different paths (or networks) show different correlations to the retweet adjacency matrix $R$ and mention adjacency matrix $M$. This indicates that different paths (or networks) should be given different credits for the prediction task. Furthermore, most of the correlation values are not shown to be close to 1, which demonstrates that this correlation metric would not be dominated by the sparseness of the network adjacency matrices. If they are dominated by the sparseness of the matrices, the correlation values should be very close to 1. We also have the following observations and indications:

- The correlation between $R$ and $M$ is stronger than the correlation between $R$ and $F^*$, and stronger than correlation between $M$ and $F^*$. This indicates that to predict retweeters, the feature of mentioners are more important than the feature of followers; similarly to predict mentioners, the feature of retweeters is more important than the feature of followers.

- The correlation between $R$ and $RF^*$ is stronger than the correlation between $R$ and $F^*R$. This indicates that for a user, the followers of his retweeters are more likely to retweet him than the retweeters of his followers.

- The correlation between $R$ and $MF^*$ is stronger than the correlation between $R$ and $F^*M$. This indicates that for a user, the followers of his mentioners are more likely to retweet him than the mentioners of his followers.

- The correlation between $R$ and $MF^*$ is stronger than the correlation between $M$ and $F^*$. This indicates that for a user, the followers of his mentioners are more likely to mention him than the retweeters of his followers.

- The correlation between $M$ and $MF^*$ is stronger than the correlation between $M$ and $F^*M$. This indicates that for a user, the followers of his mentioners are more likely to mention him than the mentioners of his followers.

### 5.3 Network Properties

Figure 4 reports on the average numbers of followers, retweeters, mentioners from the 15K users and the 2-month dataset for each twitterer (excluding to-self).

![Figure 4: Average numbers of followers, retweeters, mentioners from the 15K users and the 2-month dataset for each twitterer (excluding to-self).](image)

Table 3: Correlation to retweet network adjacency matrix

<table>
<thead>
<tr>
<th></th>
<th>$R$</th>
<th>$M$</th>
<th>$F^*$</th>
<th>$RR$</th>
<th>$MM$</th>
<th>$F^<em>F^</em>$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$M$</td>
<td>0.653</td>
<td>1.000</td>
<td>0.222</td>
<td>0.638</td>
<td>0.495</td>
<td>0.073</td>
</tr>
<tr>
<td>$RM$</td>
<td>0.326</td>
<td>0.601</td>
<td>0.130</td>
<td>0.443</td>
<td>0.116</td>
<td></td>
</tr>
</tbody>
</table>

$K = 20$, where $K$ is used in the definition of Retweet and Mention Prediction in Section 3.1.

$\alpha = 0.00005$ and $\beta = 1.0$, where $\alpha$ and $\beta$ are parameters in equations (2), (4), (5), (6) and (7). $\epsilon = 3.5$, where $\epsilon$ is a parameter in equations (8) and (9).

For methods SVD and SVD+BON, in equation (10), $s$ is set to be 5227 for retweet prediction and 6997 for mention prediction; again these are the best values for the dataset.

### 5.4 Hybrid Network Correlation

We know that there are some correlation between different types of networks. As discussed in section 4, we used the metric of average Spearman’s rank correlation coefficient to evaluate their correlation. For two network adjacency matrices $A$ and $B$, we calculated the Spearman’s rank correlation coefficients of the corresponding rows of the two matrices, and then used the average value to represent the correlation of the two matrices.

Table 4: Correlation to mention network adjacency matrix

<table>
<thead>
<tr>
<th></th>
<th>$R$</th>
<th>$M$</th>
<th>$F^*$</th>
<th>$RR$</th>
<th>$MM$</th>
<th>$F^<em>F^</em>$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$M$</td>
<td>0.653</td>
<td>1.000</td>
<td>0.222</td>
<td>0.578</td>
<td>0.475</td>
<td>0.076</td>
</tr>
</tbody>
</table>

### 6. EVALUATION RESULTS

#### 6.1 Baseline Methods

**NaiveHybrid**: This baseline method is to combine $R$, $M$, $F^*$ unweighted and calculate Bonacich metric at this combined network.
where $\delta$ is a diagonal matrix of singular values of $\Sigma$. We provide several network based features to the SVM in -

SVM [13] learns a ranking function from partial training ranking.

Finally, we consider a supervised learner. Ranking SVM [13] learns a ranking function from partial ranking training data. We provide several network based features to the SVM including the adjacency matrices of retweet counts, mention counts, and the edge weights of the adjusted follower network $F^\ast$.

**6.2 Comparison wth Baseline Methods**

Figure 8 reports on the MAP, NDCG@5 and NDCG@10 for retweet prediction. Figure 9 reports on the results for mention prediction. Both Figures 8 and 9 show that hybridBON has the best prediction accuracy. It dominates NaiveHybrid and BON. It also outperforms SVD and SVD+BON. SVD+BON does not show improvement over BON, which means that applying Bonacich metric to the score matrix from the output of SVD rather than the original matrix does not help.

Finally, hybridBON also outperforms a supervised learner rank-
ingsVM using network based features. This demonstrates that our proposed solution hybridBON, inspired by the Bonacich metric based potential function and the hybrid network, can improve prediction accuracy. Further, while hybridBON is an unsupervised method that only estimates the future Bonacich potential and future hybrid network, it still has the best accuracy.

The performance accuracy appears to be higher for retweet prediction compared to mention prediction. We note that retweets reflect the influence of both a topic and a focal user, and this may explain the improved prediction accuracy.

To evaluate the running time of the solutions, we ran the Matlab code on Red Hat Enterprise Linux with Intel Core2 Quad Q8400,
Table 5: Running time of the methods SVD, BON, SVD+BON and hybridBON

<table>
<thead>
<tr>
<th>Method</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>NaiveHybrid</td>
<td>0.2 hour</td>
</tr>
<tr>
<td>SVD</td>
<td>11 hours</td>
</tr>
<tr>
<td>BON</td>
<td>0.2 hour</td>
</tr>
<tr>
<td>SVD+BON</td>
<td>11.2 hours</td>
</tr>
<tr>
<td>hybridBON</td>
<td>0.5 hour</td>
</tr>
</tbody>
</table>

4 cores and 12GB memory. Table 5 shows the running time for the methods except RankingSVM. SVD took around 11 hours; BON and NaiveHybrid took around 10 minutes; SVD+BON took more than 11 hours; hybridBON took around 30 minutes. To summarize, hybridBON and BON and NaiveHybrid are much more efficient than SVD and SVD+BON. We did not include the running time for RankingSVM in the table since it was not implemented in Matlab and hence it is not a good way to put them together. For RankingSVM it took us less than half an hour to train a model with a fixed trade-off parameter $\mathcal{C}$. To tune the trade-off parameter $\mathcal{C}$ for RankingsVM, it took pretty many rounds and pretty much time.

To explain, SVD has complexity $O(C \cdot N^3)$, where $C$ is a constant and $N$ is the matrix size. The complexity of BON is the complexity of matrix inversion. Matrix inversion has complexity $O(C \cdot N^3)$. However, Strassen’s method for inversion has complexity $O(C \cdot N^\log_2(7)) = O(C \cdot N^{2.807...})$ to calculate the inversion of a completely general matrix. Further, the constant $C$ for SVD is bigger than the constant for matrix inversion and there are additional optimized solutions for matrix inversion on sparse matrices. The complexity of hybridBON is the complexity of the parameters which is $O(C \cdot N^2)$ plus the complexity of BON. Thus, BON and hybridBON is more efficient than SVD.

We note that the research in [11] addressed a similar problem: given a tweet, does a focal user retweet the tweet. They used a matrix co-factorization (coFM) based solution for the objective function (http://www.libfm.org/), and they used stochastic gradient descent (SGD) for learning. While they do not provide details of execution time, we note that the computation of the many (content and network based) features that they use is likely to be quite expensive. They report on high MAP scores for their prediction which is appropriate since they use supervised learning.

To summarize, hybridBON is a fast approximate solution with good accuracy. Despite hybridBON only estimating the future Bonacich potential and future hybrid network, it shows the superiority of the Bonacich metric and hybrid network as features for link prediction. It outperforms a state of the art unsupervised approach based on singular value decomposition (SVD) in both accuracy and efficiency. hybridBON is also more accurate than a supervised approach based on ranking SVM that uses three network based features. While a supervised learning approach coFM+SGD in [11] may have higher accuracy, it used a large number of features and is likely to be computationally more expensive; the higher accuracy is expected from a supervised approach. In future work, we plan to explore supervised solutions using the Bonacich based feature.

### 6.3 Parameter Analysis

For method hybridBON, there are three parameters: $\alpha$, $\beta$, and $\varepsilon$. If we look at Equations (5), (6) and (7), we can see that $\beta$ will only have impact on the absolute score values of the nodes, but will not have impact on their rankings and prediction output. For simplicity, we set $\beta = 1.0$. We conducted experiments to test the impact of the parameters $\alpha$ and $\varepsilon$ to the method hybridBON.

Figure 10 reports on the MAP values of hybridBON with respect to different $\varepsilon$ values, while keeping $\alpha$ and $\beta$ fixed. The figure shows that when $\varepsilon$ is greater than 0, the MAP values are pretty stable. When $\varepsilon$ value is around 3.5, the MAP value shows highest, and when $\varepsilon$ is bigger than 3.5 it decreases slowly. This tells us that method hybridBON is not sensitive to the values of parameter $\varepsilon$. Figure 11 reports on the MAP values of hybridBON with respect to different $\alpha$ values, while keeping $\varepsilon$ and $\beta$ fixed. The figure shows that when $\alpha$ is less than 0.0005 and greater than 0, the MAP has pretty stable high values. When $\alpha$ value is is greater than 0.0005, the MAP values show significant decrease. So as long as we keeps $\alpha$ values less than some small value, the method hybridBON is not sensitive to the values of parameter $\alpha$.

### 7. CONCLUSIONS

In this paper, we address the microblog prediction problem by utilizing the hybrid networks. Given a focal user, we predict who will retweet her tweets and who will mention her in the near future. We propose a general prediction model to utilize the hybrid...
networks for the prediction and propose an approximate approach hybridBON based on the prediction model. Our solution hybridBON is based on a potential function over a hybrid network. The potential function converges to the Bonacich metric.

We found that for a focal user, the followers of his retweeter are more likely to retweet him than the retweeters of his followers, and the followers of his mentioners are more likely to mention him than the mentioners of his followers, and other path patterns.

We demonstrate that hybridBON outperforms a state of the art baseline solution based on singular value decomposition SVD, and also outperforms RankingSVM with edge weight features of the networks. We note that coFM-SGD is a supervised approach using matrix co-factorization and stochastic gradient based learning and has good accuracy on a similar problem.

Further, we note that there is little research on learning any centrality based objective function and in future work we will explore a supervised approach to hybridBON.

8. REFERENCES