ReadBehavior: Reading Probabilities Modeling of Tweets via the Users' Retweeting Behaviors

Jianguang Du, Dandan Song, Lejian Liao, Xin Li, Li Liu, Guoqiang Li, Guanguo Gao, and Guiying Wu

Beijing Engineering Research Center of High Volume Language Information Processing & Cloud Computing Applications, Beijing Key Laboratory of Intelligent Information Technology, School of Computer Science & Technology, Beijing Institute of Technology, Beijing, China, 100081
{dujianguang,sdd,liaolj,xinli,3120120376,lgqsj}@bit.edu.cn,
{guanguogao1005,wuguiying1989}@gmail.com

Abstract. Along with *twitter*'s tremendous growth, studying users' behaviors, such as retweeting behavior, have become an interesting research issue. In literature, researchers usually assumed that the *twitter* user could catch up with all the tweets posted by his/her friends. This is untrue most of the time. Intuitively, modeling the reading probability of each tweet is of practical importance in various applications, such as social influence analysis. In this paper, we propose a *ReadBehavior* model to measure the probability that a user reads a specific tweet. The model is based on the user's retweeting behaviors and the correlation between the tweets' posting time and retweeting time. To illustrate the effectiveness of our proposed model, we develop a *PageRank*-like algorithm to find influential users. The experimental results show that the algorithm based on *ReadBehavior* outperforms other related algorithms, which indicates the effectiveness of the proposed model.

1 Introduction

Micro-blog services, such as *Twitter*, have grown rapidly in recent years. It has more than 500 million registered users and 200 million active users. More than 400 million tweets are posted per day[1]. On *twitter*, the user whose tweets are followed is called *friend*, while the user who is following is called *follower*.

Followers may get lots of tweets from their *friends*[2]. One scenario is that the user has too many active friends, who post tweets very frequently. In this scenario, he/she is unable to read all the tweets. Previous works, such as on identifying influential users [1], assumed that *followers* could read all the tweets on their micro-blog spaces. However, it is actually untrue most of the time. Another work [2] systematically investigated the underlying mechanism of the retweeting

[1] http://expandedramblings.com/index.php/march-2013-by-the-numbers-a-few-amazing-twitter-stats/.

[2] Without causing misunderstanding, we will use "tweets" to denote "tweets from users' *friends*" in the rest of this paper.

V.S. Tseng et al. (Eds.): PAKDD 2014, Part I, LNAI 8443, pp. 114–125, 2014.

behaviors, and divided the tweets into three categories: retweet, ignore[3], and miss. Nonetheless, it is not obvious to detect whether the tweets are missed or ignored in some situations, even with the help of users' login time.

In this paper, we propose a *ReadBehavior* model to measure the reading probability of each tweet, where reading probability means the probability of a tweet that is read by a user. The model is based on the user's retweeting behavior, which is the correlation between the tweet's original posting time and the corresponding retweeting time. *ReadBehavior* is of practical importance and can be beneficial for various applications, such as social influence analysis.

To illustrate the effectiveness of *ReadBehavior*, we develop an **I**mproved **P**age**R**ank (*IPR*) algorithm, which is an extension of the *PageRank* [3] algorithm. Both *IPR* and *PageRank* find influential users on the whole network. *PageRank* assumes that all the tweets are read by the user, whereas *IPR* calculates the reading probability of tweets, and then estimates the number of tweets that are read.

Experiments are then conducted to evaluate the performance of our proposed model. The results show that *IPR* outperforms other related algorithms, such as *PageRank*, *FollowersNum*, and *TweetsNum*. Consequently, the results also verify the effectiveness of *ReadBehavior*.

The main contributions of this paper include:

1. We propose a *ReadBehavior* model to measure the reading probability of each tweet posted to a user according to his/her retweeting behaviors. To the best of our knowledge, it is the first model to measure the reading probability of tweets.
2. Based on our proposed model, we develop an *IPR* algorithm to find influential users. The experimental results show the advantages and effectiveness of the proposed model.

The rest of this paper is organized as follows. A review of related work is given in section 2. Section 3 presents the proposed *ReadBehavior* model in detail. Following that, in Section 4, we present *IPR*, which is based on *ReadBehavior*. Then, experimental results are presented in Section 5. Finally, Section 6 concludes this paper.

2 Related Work

There have been quite some studies on micro-blog services, especially on *Twitter*, e.g., identifying influential users. The Web Ecology project [4] measured influential users for a 10-day period. This work performed a comparison of three measures of influence - retweet, reply, and mention. Cha et al. [5] used mentions, retweets, and the number of followers as influence measurement. Kwak et al. [6] used *PageRank* of the network constructed by *followers*. Bakshy et al. [7] used the information cascades. Recently, researchers also studied social

[3] "ignore" means that *followers* read the tweets but do not choose to retweet them.

influence from the topics perspective [8,9,10,11,12]. Romero et al. [1] designed the influence-passivity algorithm to measure influential users. However, all these methods assumed that *followers* could read all the tweets. Whereas in reality, users may miss some of the tweets.

Other works studied the retweeting behaviors. Boyd et al. [13] treated retweeting behaviors as conversations inside Twitter, and studied the basic issues about retweet. Hong et al. [14] studied the problem of predicting popular tweets according to the future retweets. Petrovic et al. [15] explored tweets features to predict whether a tweet would be retweeted. Benevenuto et al. [16] analyzed the user workloads based on users' behaviors. Uysal and Feng [17,18] proposed a tweet ranking method to help users to catch up with valuable tweets based on the retweet history. Compared with our model, they always ignored the tweets time (tweets' posting time), which we treat as an important factor.

There were some works that mentioned the tweets time. Yang et al. [2] studied the underlying mechanism of the retweeting behaviors. They divided the tweets into three categories - retweet, ignore, and miss, and then classified the retweeting delay into short-term intervals and long-term intervals. However, they did not mention how to deal with the situation when it was not clear whether a tweet was missed or ignored. Dabeer et al. [19] proposed response probability that bear the similarity with susceptibility using the tweets time. However, as far as we know, very few researchers have studied the tweets time in the view of measuring the reading probability.

3 The Proposed Model

To measure the reading probabilities of the tweets on users' micro-blog spaces, we propose the *ReadBehavior* model according to users' retweeting behaviors. In this section, we will describe the model in detail.

3.1 *Terminology*, *Assumption*, and *Fact*

When users reading tweets on their micro-blog spaces, the timeline lists the tweets in reverse chronological order. Once they find an interesting tweet, they will retweet it. This is denoted as the retweeting behavior.

For the simplicity of the analysis, we have the following *terms*, *assumptions*, and *facts*. We should make it clear that our assumptions are based on users' general reading habits. Although we do not think about special cases, the experimental results in Section 5 still show the advantages of our model.

Term. *Check* means the user's reading behavior. *Check period* means a period when the user is continuously reading the tweets, and *check time* means the start time of the *check period*.

***Assumption 1*.** Users read the tweets from top to bottom, i.e., they read from the latest one to the earliest one. Although the mobile *Twitter* client starts from the last tweet that was shown to the user, he/she must refresh to get new tweets with the latest one on the top.

Assumption 2. If users encounter a tweet which they have already read, they will not read the tweets below it.

Assumption 3. Once users read an interesting tweet, they will retweet it, i.e., they will not read back to retweet a tweet.

Assumption 4. Assume that time $t_i^{'}$ and time $t_j^{'}$ are two adjacent retweets time during one *check period*, and t_i and t_j are the corresponding tweets time, respectively. If $t_i < t_j$, then $t_i^{'} > t_j^{'}$, and the user must read the tweets posted between t_i and t_j. Because we think he/she is continuously reading the tweets.

Fact 1. Given a tweet posted at time t_i, and it was retweeted at time $t_i^{'}$. The *check time* of this reading behavior must be between t_i and $t_i^{'}$.

Fact 2. If a user retweets a tweet, he/she must have read the tweet.

Fact 3. Tweets, which are posted after the *check time*, are not able to be read by the user in this *check period*. If users refresh to get new tweets, we treat it as a new *check period*.

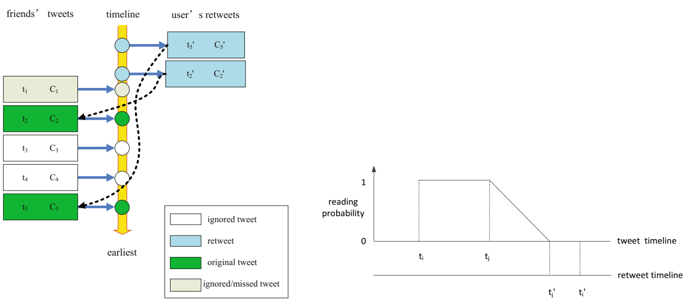
3.2 *ReadBehavior* Model

Two Retweeting Behaviors. In the proposed *ReadBehavior* model, we study the retweets time and the corresponding original tweets time. For example, assume that user A was reading the tweets. When he/she found an interesting tweet C_i which was posted at time t_i, he/she retweeted it at time $t_i^{'}$ ($t_i < t_i^{'}$). After that, he/she may continue reading or become idle. Assume that there also exists another retweeting behavior at time $t_j^{'}$, and the original tweet C_j was posted at time t_j ($t_j < t_j^{'}$). There are three intuitive scenarios considering the time sequence if $t_i < t_j$:

Scenario 1: $t_i < t_j < t_j^{'} < t_i^{'}$

Scenario 2: $t_i < t_i^{'} < t_j < t_j^{'}$

Scenario 3: $t_i < t_j < t_i^{'} < t_j^{'}$



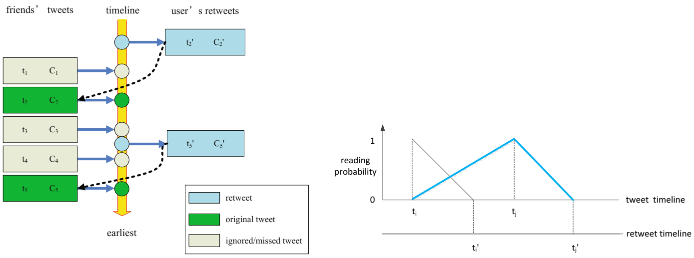
(a) Reading Behaviors (b) Reading Probability

Fig. 1. Reading Behaviors and Reading Probability of scenario 1

For scenario 1, we show an intuitive example in Fig. 1(a). When the user was reading the tweets, he/she retweeted C_2 at $t_2^{'}$. Then he/she continued reading the tweets, and retweeted C_5 at $t_5^{'}$. As the user was continuously reading the tweets, he/she read the tweets posted between t_2 and t_5. It was not sure whether C_1 was read or missed. Here t_i and t_j are corresponding to t_5 and t_2, respectively.

The reading probability is illustrated by Fig. 1(b). According to *Fact 2*, the two retweeting behaviors take place in one *check period*. The reading probability of every tweet posted between t_i and t_j (including t_i and t_j) is 1 according to *Assumption 4*. The *check time* is between t_j and $t_j^{'}$ according to *Fact 1*. Similarly, the reading probabilities of tweets newer than $C_j^{'}$ are 0 according to *Fact 3*. Because the reading probabilities of tweets posted between t_j and $t_j^{'}$ are descending, we assume that the reading probabilities are linearly descending. So given t_x which is between t_i and $t_i^{'}$, then the reading probability of C_x in this *check period* is measured by:

$$P(C_x) = \begin{cases} 1, & t_i \leqslant t_x \leqslant t_j \\ \frac{t_j' - t_x}{t_j' - t_j}, & t_j < t_x < t_j' \\ 0, & t_j' \leqslant t_x \leqslant t_i' \end{cases} \tag{1}$$



(a) Reading Behaviors (b) Reading Probability

Fig. 2. Reading Behaviors and Reading Probability of scenario 2

Fig. 2(a) is an example of scenario 2. The user retweeted C_5 at $t_5^{'}$ in a *check period*. Then he/she stopped reading the tweets. In the next *check period*, he/she retweeted C_2 at $t_2^{'}$. It was not sure whether C_1, C_3, and C_4 were read or missed. Here t_i and t_j are corresponding to t_5 and t_2, respectively.

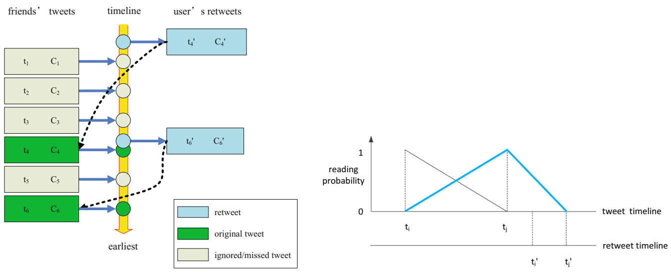
We can infer that the two retweeting behaviors take place in two *check periods*. Consequently, we calculate the reading probability for these two *check periods* respectively, as shown in Fig. 2(b). The black line and the blue line illustrate the reading probability of C_x measured by the former *check period* (where C_i was retweeted) and the latter *check period* (where C_j was retweeted), respectively. For the former *check period*, similar to the analysis of scenario 1, the reading probabilities of tweets posted at t_i and $t_i^{'}$ (if exist) are 1 and 0, respectively. If $t_i < t_x < t_i^{'}$, the reading probability of C_x is $\frac{t_i^{'} - t_x}{t_i^{'} - t_i}$. For the latter *check period*, the reading probabilities of tweets posted before $t_j^{'}$ are 0 according to *Fact 3*. If t_x

is between t_j and $t_j^{'}$, the reading probability of C_x is $\frac{t_j^{'}-t_x}{t_j^{'}-t_j}$. And if t_x is between t_i and t_j, the reading probability of C_x is $\frac{t_x-t_i}{t_j-t_i}$. The tweets posted earlier than C_i are not read according to *Assumption 2*. Now we take both the *check periods* into account. When the reading probability of C_x can be measured by two *check periods*, we will choose the higher one.

To sum up, the reading probability of C_x posted between t_i and $t_j^{'}$ is measured by:

$$P(C_x) = \begin{cases} \max(\frac{t_j^{'}-t_x}{t_i^{'}-t_i}, \frac{t_x-t_i}{t_j-t_i}), & t_i \leqslant t_x < t_j \\ \frac{t_j^{'}-t_x}{t_j^{'}-t_j}, & t_j \leqslant t_x \leqslant t_j^{'} \end{cases} \quad (2)$$

Fig. 3(a) is an example of scenario 3. Similar to scenario 2, C_6 was retweeted at $t_6^{'}$ in the first *check period*. C_4 was retweeted at $t_4^{'}$ in the second *check period*. Here t_i and t_j are corresponding to t_6 and t_4, respectively.



(a) Reading Behaviors (b) Reading Probability

Fig. 3. Reading Behaviors and Reading Probability of scenario 3

The main difference of this scenario from scenario 2 is that the retweet time $t_i^{'}$ is just after the tweet time t_j. It means that the first *check time* is between t_j and $t_i^{'}$ (just after t_j), as shown in Fig. 3(b). Then similar to scenario 2, the reading probability of C_x posted between t_i and $t_j^{'}$ is measured by:

$$P(C_x) = \begin{cases} \max(\frac{t_j-t_x}{t_j-t_i}, \frac{t_x-t_i}{t_j-t_i}), & t_i \leqslant t_x < t_j \\ \frac{t_j^{'}-t_x}{t_j^{'}-t_j}, & t_j \leqslant t_x \leqslant t_j^{'} \end{cases} \quad (3)$$

For tweets which are earlier than both of the original tweets, the reading probabilities of them can be measured by earlier *check periods*, which will be discussed in the following section.

Three and More Retweeting Behaviors. The above three scenarios are the essential situations. Now we expand them to three and more continuous retweeting behaviors, which are the combination of the three scenarios. We can summarize that if t_x is between t_i and t_j, the reading probability of C_x can be measured by one of the three scenarios, otherwise, the reading probability of C_x can be measured by additional boundary conditions.

Formally, assume the retweets time sequence is $(t'_1, t'_2, ..., t'_{n-1}, t'_n)$, and the corresponding tweets time sequence is $(t_1, t_2, ..., t_{n-1}, t_n)$[4]. $list_t$ is the time list of the tweets that are retweeted, t_{start} is the start time of the dataset and t_{end} is the end time of the dataset. If C_x is between two adjacent retweeting behaviors, t_i and t_j are the former tweet time and the latter tweet time, respectively, then the reading probability of C_x is measured by Algorithm 1.

Algorithm 1. Reading probability of tweet C_x posted at time t_x

Input:
The time of a tweet posted by user A's friends;
The set of user A's retweets time and the corresponding tweets time;
Output:
The reading probability of the tweet;

if $t_x > t_n$ AND $t_x < t'_n$ **then**
  $P(C_x) = \frac{t'_n - t_x}{t'_n - t_n}$; # The latest check period
else if t_x is in $list_t$ **then**
  $P(C_x) = 1$; # C_x was retweeted
else if $t_x > t_i$ AND $t_x < t'_j$ **then**
  if $t'_i < t_j$ **then**
    $P(C_x)$ is measured by Equation 2; # Scenario 2
  else if $t'_j < t'_i$ **then**
    $P(C_x) = 1$; # Scenario 1
  else if $t_j < t'_i$ AND $t'_i < t'_j$ **then**
    $P(C_x)$ is measured by Equation 3; # Scenario 3
  end if
else if $t_x \leqslant t_1$ AND $t_x \geqslant t_{start}$ **then**
  $P(C_x) = \frac{t_x - t_{start}}{t_1 - t_{start}}$; # The earliest check period
else
  $P(C_x) = 0$;
end if
return $P(C_x)$;

With the reading probability of each tweet, we estimate the number of tweets read by a user. Assume that user B is a friend of A, and the number of tweets B posted is n, then the number of tweets read by A from B is measured by:

$$Num_{BA} = \sum_{x=1}^{n} P(C_x) \tag{4}$$

[4] Though the tweets time sequence is in the order of time, the corresponding retweets time sequence may not be.

4 The Improved *PageRank* Algorithm

In this section, we will describe the *IPR* algorithm with our proposed *ReadBehavior* model. First of all, we have the following definition:

Definition 1. $\boldsymbol{G = (N, E, W)}$ is a directed graph. N is the set of nodes. E is the set of arcs. W is the set of arc weights. If user i has ever retweeted from user j, then there is a directed arc (i, j) between them.

Based on the proposed model, we develop an *IPR* algorithm, which extends from the well-known *PageRank* algorithm. *PageRank* [3] was used by researchers to find influential users in social media. It took both the pairwise influence and the link structure into account. The main difference between *IPR* and *PageRank* is the calculation of the arc weight in Definition 1. The arc weight of *PageRank* is measured by:

$$w_{ij} = \frac{S_{ij}}{Q_j}, \tag{5}$$

where S_{ij} is the number of tweets that i has retweeted from j, and Q_j is the number of tweets posted by j.

While the arc weight of *IPR* is measured by:

$$w_{ij} = \frac{S_{ij}}{Num_{ij}}, \tag{6}$$

where S_{ij} is the number of tweets posted by j and retweeted by i, and Num_{ij} is the estimated number of tweets read by i from j according to Equation 4.

With the directed graph G, both *PageRank* and *IPR* iteratively compute to find influential users. Since G is weighted, the random surfer probability from i to j is measured by $\frac{w_{ij}}{\sum_{k:(i,k)\in E} w_{ik}}$, where E is the set of arcs in G.

5 Experiments

To evaluate the effectiveness of our proposed model, we present the experiments on a large-scale *Twitter* dataset in this section.

5.1 *Twitter* Dataset

A set of *Twitter* data from [6] is prepared for this study. The dataset contains a continuous stream of about 132 million tweets. In order to compare with the work in [1], we also use the tweets with URLs. We get about 3.38 million users posting at least one URL. In the following descriptions, we will use "tweets" to represent "tweets with URLs" in short. Similar to [1], we choose to concern users who have at least 7 tweets with URLs, and exclude the invalid users whose user ID cannot be accessed with the usernames. After that, we get 497,782 users with 17,592,586 tweets. We then strike out the isolated users (users that never retweet or are retweeted by others). Finally, we get 74,813 users who post 3,150,334 tweets. Among them, 83,356 pairs of users have retweeting relationships, and the number of retweets is 103,774.

5.2 Comparison of *IPR* with Other Related Algorithms

In this section, we study the effectiveness of *IPR* in finding influential users. Several related comparison algorithms are conducted, which include:

- ***FollowerNum***, which measures the influence of users by the number of *followers*. This measurement is widely used in many *Twitter* services.
- ***TweetsNum***, which measures the influence of users by the number of tweets posted by them.
- ***PageRank***, which measures the influence of users taking both link structure and pairwise influence into account. Nevertheless, unlike *IPR*, the arc weight is measured by Equation 5.

Evaluation Method. All the four algorithms can find influential users. However, there is no existing method to directly compare their performances. We use a cross-validation method [20] to compare these algorithms. The method is described as below:

Given four algorithms A, B, C, and D, and the sets of Top-K influential users discovered by them are I_A, I_B, I_C, and I_D, respectively. The criterion set is denoted as:

$$I_2 = (I_A \cap I_B) \cup (I_A \cap I_C) \cup (I_A \cap I_D) \cup (I_B \cap I_C) \cup (I_B \cap I_D) \cup (I_C \cap I_D) \quad (7)$$

Then, the precision of algorithm X is:

$$P_X = \frac{|I_X \cap I_2|}{|I_X|} \quad (8)$$

The recall is:

$$R_X = \frac{|I_X \cap I_2|}{|I_2|} \quad (9)$$

A better algorithm will get a higher precision score and a higher recall score. Because the best algorithm should have the greatest contribution to the criterion set.

Performance and Analysis. We compare the precision, recall, and F measure of Top-K ($K = 200, 250, 300, 350, 400, 450, 500$) influential users of the above four algorithms. Fig. 4 shows the results.

We see from Fig. 4(a) that the precisions of all the algorithms increase when the parameter K increases. In addition, as K increases, the precisions of *IPR* and *PageRank* significantly increase, whereas the precisions of *FollowersNum* and *TweetsNum* only increase little. This is because both *IPR* and *PageRank* have greater contributions to the criterion set in Equation 7. The greater contributions cause the numerator of Equation 8 to grow at faster rate comparing with the denominator. We also find that the precisions of all the algorithms are low. The reason is that the Top-K influential users discovered by these algorithms are rarely correlated with each other. However, as K increases, the correlation

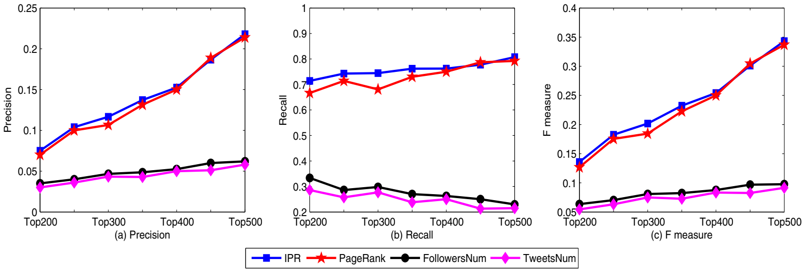


Fig. 4. Precision, Recall, and F measure of different algorithms in finding influential users

between *IPR* and *PageRank* becomes stronger. This also indicates the significant performance increasing of *IPR* and *PageRank*.

Fig. 4(b) shows the recalls of different algorithms. Surprisingly, as K increases, the recalls of *FollowersNum* and *TweetsNum* decrease. We note that the decrease in recall can be attributed to two main factors: first, both algorithms have little contribution to the criterion set, and second, both algorithms do not correlate with other algorithms. So these two factors cause the value of the numerator of Equation 9 to be small. We also observe that the recalls of *IPR* and *PageRank* are really high, which indicates that most elements of the criterion set are consist of the influential users discovered by the two algorithms.

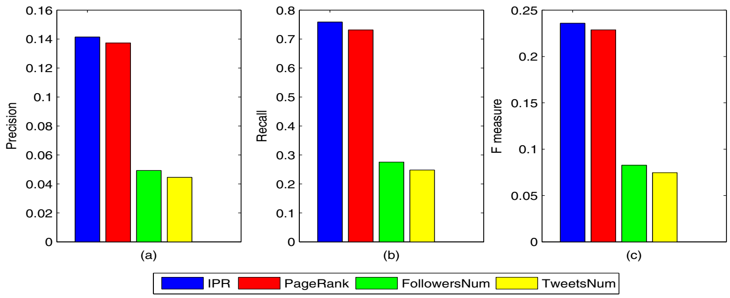


Fig. 5. The average of Precision, Recall, and F measure of different algorithms

By combining precision and recall, we get a comprehensive measurement-F measure-of those algorithms, as shown in Fig. 4(c). It is clear that *IPR* outperforms other algorithms in most cases except $K = 450$. When checking in detail, we find that the influential users (discovered by *IPR*), whose ranks are between 400 and 450, post and retweet small number of tweets containing URLs. Moreover, we find that *IPR* and *PageRank* significantly outperform *FollowersNum* and *TweetsNum* in all of the measurements, which suggests that both the number of followers and the number of tweets may not be good measures of influence.

We also compare the average precision, recall, and F measure of different algorithms. Fig. 5 clearly shows that *IPR* outperforms other algorithms.

All the experimental results verify the effectiveness of the proposed model. This is because the pairwise influence measured by *ReadBehavior* is more accurate than traditional methods, and this leads to more precision results of the global influence.

6 Conclusion and Future Work

The number of messages on the micro-blog space is large, and it is almost impossible for users to catch up with all the messages. Motivated by the fact, this paper focuses on proposing a *ReadBehavior* model to measure the reading probabilities of tweets posted by a user's friends according to his/her retweeting behaviors. To the best of our knowledge, this work is the first to measure the reading probabilities of tweets. To illustrate the effectiveness of our proposed model, an **I**mproved **P**age**R**ank (*IPR*) algorithm is proposed to find influential users on the whole network. We find from the experimental results that *IPR* outperforms its competitors, which indicate the effectiveness of our model. We should emphasize that the model not only can be used to measure influential users, but also can be used to a lot of applications which capture the reading behaviors of users.

Nevertheless, as the first attempt, it still has spaces for improvement. First, as a preliminary model, the linear approximation is used due to its simpleness. In the future, a more appropriate method should be used to better simulate users' reading behaviors. Second, the current model takes only the retweeting behaviors into account. So the reading probability cannot be measured by our model in the extreme situation when a user never retweets others. Although this does not impact the calculation in the current application, in our future work, we consider taking other user information into account. User behaviors, such as reply and mention, could also contribute to the reading probability estimation. Last but not least, *Twitter Lists* and *Groups* may be other factors that influence the reading probability. This would be an exciting direction for future work.

Acknowledgments. This work is funded by the National Program on Key Basic Research Project (973 Program, Grant No. 2013CB329605), National Natural Science Foundation of China (NSFC, Grant Nos. 60873237, 61300178 and 61003168), and Beijing Higher Education Young Elite Teacher Project (Grant No. YETP1198).

References

1. Romero, D.M., Galuba, W., Asur, S., Huberman, B.A.: Influence and passivity in social media. In: Gunopulos, D., Hofmann, T., Malerba, D., Vazirgiannis, M. (eds.) ECML PKDD 2011, Part III. LNCS, vol. 6913, pp. 18–33. Springer, Heidelberg (2011)

2. Yang, Z., Guo, J., Cai, K., Tang, J., Li, J., Zhang, L., Su, Z.: Understanding retweeting behaviors in social networks. In: CIKM, pp. 1633–1636. ACM (2010)
3. Page, L., Brin, S., Motwani, R., Winograd, T.: The pagerank citation ranking: bringing order to the web (1999)
4. Leavitt, A., Burchard, E., Fisher, D., Gilbert, S.: The Influentials: New Approaches for Analyzing Influence on Twitter. Webecology Project (September 2009)
5. Cha, M., Haddadi, H., Benevenuto, F., Gummadi, P.K.: Measuring user influence in twitter: The million follower fallacy. In: ICWSM, vol. 10, pp. 10–17 (2010)
6. Kwak, H., Lee, C., Park, H., Moon, S.: What is Twitter, a social network or a news media? In: WWW 2010: Proceedings of the 19th International Conference on World Wide Web, pp. 591–600. ACM, New York (2010)
7. Bakshy, E., Hofman, J.M., Mason, W.A., Watts, D.J.: Everyone's an influencer: quantifying influence on twitter. In: Proceedings of the Fourth ACM International Conference on Web Search and Data Mining, pp. 65–74. ACM (2011)
8. Liu, L., Tang, J., Han, J., Jiang, M., Yang, S.: Mining topic-level influence in heterogeneous networks. In: CIKM, pp. 199–208. ACM (2010)
9. Lin, C.X., Mei, Q., Han, J., Jiang, Y., Danilevsky, M.: The joint inference of topic diffusion and evolution in social communities. In: Proceedings of the 11th International Conference on Data Mining (ICDM), pp. 378–387. IEEE (2011)
10. Tang, J., Sun, J., Wang, C., Yang, Z.: Social influence analysis in large-scale networks. In: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 807–816. ACM (2009)
11. Weng, J., Lim, E.P., Jiang, J., He, Q.: Twitterrank: finding topic-sensitive influential twitterers. In: Proceedings of the Third ACM International Conference on Web Search and Data Mining, pp. 261–270. ACM (2010)
12. Macskassy, S.A., Michelson, M.: Why do people retweet? anti-homophily wins the day. In: Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media, pp. 209–216 (2011)
13. Boyd, D., Golder, S., Lotan, G.: Tweet, tweet, retweet: Conversational aspects of retweeting on twitter. In: 2010 43rd Hawaii International Conference on System Sciences (HICSS), pp. 1–10. IEEE (2010)
14. Hong, L., Dan, O., Davison, B.D.: Predicting popular messages in twitter. In: Proceedings of the 20th International Conference Companion on World Wide Web, pp. 57–58. ACM (2011)
15. Petrovic, S., Osborne, M., Lavrenko, V.: Rt to win! predicting message propagation in twitter. In: Prof. of AAAI on Weblogs and Social Media (2011)
16. Benevenuto, F., Rodrigues, T., Cha, M., Almeida, V.: Characterizing user behavior in online social networks. In: Proceedings of the 9th ACM SIGCOMM Conference on Internet Measurement Conference, pp. 49–62. ACM (2009)
17. Uysal, I., Croft, W.B.: User oriented tweet ranking: a filtering approach to microblogs. In: Proceedings of the 20th ACM International Conference on Information and Knowledge Management, pp. 2261–2264. ACM (2011)
18. Feng, W., Wang, J.: Retweet or not?: personalized tweet re-ranking. In: Proceedings of the Sixth ACM International Conference on Web Search and Data Mining, pp. 577–586. ACM (2013)
19. Dabeer, O., Mehendale, P., Karnik, A., Saroop, A.: Timing tweets to increase effectiveness of information campaigns. In: Proc. ICWSM (2011)
20. Zhaoyun, D., Yan, J., Bin, Z., Yi, H.: Mining topical influencers based on the multi-relational network in micro-blogging sites. Communications, China 10(1), 93–104 (2013)