TURank: Twitter User Ranking Based on User-Tweet Graph Analysis

Yuto Yamaguchi[1], Tsubasa Takahashi[1,⋆],
Toshiyuki Amagasa[1,2], and Hiroyuki Kitagawa[1,2]

[1] Graduate School of Systems and Information
Engineering, University of Tsukuba, Japan
{yuto_ymgc,tsubasa}@kde.cs.tsukuba.ac.jp
[2] Center for Computational Sciences, University of Tsukuba, Japan
{amagasa,kitagawa}@cs.tsukuba.ac.jp

Abstract. In this paper, we address the problem of finding authoritative users in a micro-blogging service, Twitter, which is one of the most popular micro-blogging services [1]. Twitter has been gaining a public attention as a new type of information resource, because an enormous number of users transmit diverse information in real time. In particular, authoritative users who frequently submit useful information are considered to play an important role, because useful information is disseminated quickly and widely. To identify authoritative users, it is important to consider actual information flow in Twitter. However, existing approaches only deal with relationships among users. In this paper, we propose TURank (Twitter User Rank), which is an algorithm for evaluating users' authority scores in Twitter based on link analysis. In TURank, users and tweets are represented in a *user-tweet graph* which models information flow, and ObjectRank is applied to evaluate users' authority scores. Experimental results show that the proposed algorithm outperforms existing algorithms.

1 Introduction

In recent years, micro-blogging services, where users exchange short messages, have attracted considerable attention as a new type of web services. Micro-blogging services are interesting in that they offer features similar to both blogs and SNS (Social Network Services), while they limit the length of a message that a user can send. Due to the limitation, a user can casually send messages even if messages being exchanged are not so meaningful or informative for other users. As a consequence, messages exchanged in those services are mixtures of useful and unuseful information, such as users' current status, news stories, reviews of a product or a service, and other interests. Another important feature is that a micro-blogging service allows users to communicate with each other.

Twitter [1] is one of the most famous micro-blogging services, and has shown an explosive growth in the past several years. A message posted by a user is called

⋆ The current affiliation is Service Platforms Research Laboratories, NEC Corporation.

L. Chen, P. Triantafillou, and T. Suel (Eds.): WISE 2010, LNCS 6488, pp. 240–253, 2010.

a *tweet*, and a user can *follow* any user accounts if he/she finds those accounts interesting and/or useful. Conversely, a user account may have some *followers* depending on his/her popularity or usefulness. The approximate number of user accounts in Twitter is estimated to be 75 million as of the end of 2009.

User accounts in Twitter are of various types, such as ordinary users, companies, politicians, and news sites. Java et al. [7] classified Twitter accounts into three categories. *Information source* is the category of user accounts who posts useful information. Consequently, accounts in this category tends to collect many followers; *Friends* is the category of user accounts who are friends, families, and co-workers in the real world. *Information seeker* is the category where users who rarely post messages, but follow an enormous number of user accounts to obtain information from other users' tweets. Therefore, noticing the difference among the user categories is very important, because the way information is exchanged in Twitter inherently depends on users and their behaviors.

For this reason, identifying authoritative user accounts in Twitter is a challenging and an important task for obtaining useful information. To this end, some researchers have proposed algorithms for measuring users' authority scores by analyzing the link structure consisting of *follow* relationships [7] [12]. However, in Twitter, most of users follow back their followers in accordance with mere formal courtesy. Besides, only a few percent of users in *follow* relationships communicate with each other [6]. Therefore, the algorithm using only a Twitter social graph which consists of only *follow* relationships is not sufficient.

In this paper, we propose TURank (Twitter User Rank), which is an algorithm that measures the Twitter users' authority scores considering both a Twitter social graph and how tweets actually flow among users. To address this problem, we give our focus on *retweet* (RT hereafter). RT is originated from one of the user conventions, and it allows users to resend other user's tweets to his/her followers. Generally, a user retweets a tweet if it appears to contain useful information, because he/she wants to share it with his/her followers. As a consequence, a user is considered to be authoritative if his/her tweets contain useful information, and if so, those tweets tend to be retweeted by other users. To model this, we introduce the *user-tweet graph*. A user-tweet graph consists of nodes, corresponding to user accounts and tweets, and edges, corresponding to *follow* and retweet relationships. Unlike the Twitter social graph which is relatively static, the user-tweet graph is dynamic and reconstructed whenever a new retweet is observed. We perform link structure analysis on this graph based on ObjectRank [3] to evaluate the users' authority scores reflecting the actual information flow and the dynamic property of Twitter. We show the feasibility of our approach by some experiments.

The rest of this paper is organized as follows. Section 2 describes preliminaries including an overview of Twitter and the concept of ObjectRank. Our algorithm is presented in Section 3, and is experimentally evaluated and compared to other conventional algorithms in Section 4. Related works are discussed in Section 5. Finally, Section 6 concludes this paper and discuss future works.

2 Preliminaries

In this section, we overview Twitter and its related concepts, followed by an overview of ObjectRank [3], which is a link analysis scheme for linked objects.

2.1 An Overview of Twitter

Basics of Twitter. Twitter [1] is one of the most notable micro-blogging services founded in 2006. Messages exchanged in Twitter are called *tweets*, and the maximum length of them is limited at most 140 characters. For each user account, there is a profile page with its permanent link (permalink) for displaying posted tweets. Moreover, even a tweet has its own permalink, thereby making it possible to browse or link tweets regardless of Twitter account ownership. Note that protected user accounts do not allow non-approved users to browse their tweets; only approved users are allowed to refer to the tweets.

A user may *follow* other user accounts to subscribe their tweets. Once user accounts are followed by a user, called *follower*, all tweets posted by the accounts are then displayed in the follower's *timeline* instantly. Unlike most of the other SNS services, where mutual social networking model is applied, Twitter allows users to follow others without any permission. This feature is one of the most crucial reasons why Twitter won an enormous number of users.

An RT (retweet) is a tweet that quotes a past tweet in order to disseminate the past tweet to the followers. Because RTs are originated in user convention, there are multiple formats to represent RTs. However, most of them follow the following format:

```
[additional text] RT @[account]: [original tweet]
```

It represents that *original tweet* by *account* is retweeted with some *additional text*.

Recently, Twitter has adopted this user convention as a part of official functionality, because RT is commonly used as a de facto standard. The official version of RT is called *official RTs*, while conventional RTs are called *unofficial RTs*. The official RT differs from the unofficial one in several ways: 1) for official RTs, users are not allowed to add additional text, and 2) sending an official RT is easy, because it just requires pressing a dedicated button, whereas users must copy the original text to their tweets manually in unofficial RTs.

Semantics of RT. In Twitter, information contained in tweets spreads across users via RTs. It is important to understand this dissemination process, when we try to measure a user's authority score. Boyd et al. reported that RTs are used for different purposes, such as circulating of tweets to the followers and commenting on a tweet for initiating exchange of opinions [4]. Specifically, they made analysis on RTs used for conversations. In addition, they investigated how users retweet, why users retweet, and what users retweet, and reported that there are multiple formats to represent RTs. However, recently mots RTs are of the

format described above due to the emergence of official RTs and popularization of Twitter client that support RTs.

A tweet may be retweeted by not only one user, but also two or more users, which result in a chain of RTs. For an RT chain, it can be regarded as a flow of information originated from a user to his/her followers and their descendant followers. When we look into the types of retweets, there is a tendency that a tweet is widely disseminated if the objective of the RT is to circulate useful information. On the other hand, it may not be widely disseminated if the objective is the conversation among a small number of users.

There have been several proposals that rank users based on the number of RTs. However, they have some problems. First, as mentioned above, RTs have different characteristics depending on their objectives, such as conversation and information circulation. For this reason, it is important to consider how RTs spread, that is, conversational RT spreads only for participants involved in the conversation. In contrast, recommending RTs spread widely. Hence, only counting the number of RTs will not be enough to measure the user's authority score, since RTs for conversation are less relevant to the user's authority score. Second, these methods do not consider the authority score of the user who retweets a tweet. It is natural that tweets retweeted by authoritative users are likely to be more useful than those that retweeted by unauthoritative users.

To address these problems, our algorithm considers the link structure of the tweet flow by RTs. In fact, an information dissemination process in Twitter can be modeled using a graph structure called *user-tweet graph*, which we shall introduce later. We then analyze the link structure of such graphs to evaluate users' authority scores.

2.2 ObjectRank

ObjectRank [3] is an extention of PageRank [11]. It performs link structure analysis over linked objects for measuring the importance of objects in the database. Unlike PageRank, ObjectRank takes account of edge types and node types in order to deal with multiple kinds of edges and nodes. Specifically, in ObjectRank, we differentiate each edge by setting an appropriate weight to control the flow of scores going through the edge.

At first, we construct a graph called an *authority transfer schema graph*, which is an intentional graph that models the domain of discourse. Figure 1 shows an example of an authority transfer schema graph. It illustrates the structure of the graph as well as the weight of each edge. The graph consists of the node set V containing all types of target objects, and the edge set E containing all types of edges existing between nodes in V. Notice that, for a pair of linked nodes, there always exist two edges with different directions, i.e., forward and backward edges. This means that the scores should flow both forward and backward directions. For example, the score of a paper should flow to its authors, and from the authors to the paper as well. Notice that the edge weights can sometimes be zero depending on the domain of discourse. For example, a paper which is cited by an important paper is also important, but a paper which cites an important

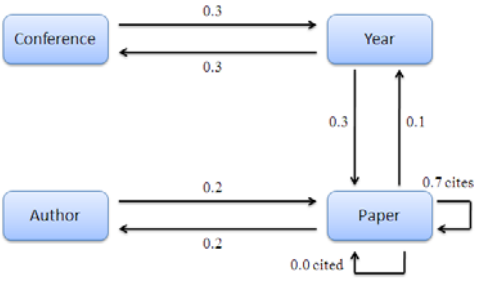

Fig. 1. An Authority Transfer Schema Graph

paper is not always important. Weights of all edges can be set by hand so that they reflect the semantics of relations between all objects. Note that the sum of weights of all edges starting from certain node must be less than or equal to 1.

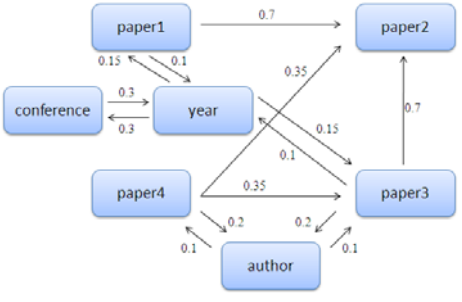

Fig. 2. An Authority Transfer Data Graph

Based on the authority transfer schema graph, we can create a *authority transfer data graph*. Figure 2 shows an example. We can see that various kinds of relationships among objects are represented. Link structure analysis is applied to this graph in such a way that, for each node, the score is propagated along the outgoing edges, and the propagated scores are computed by the weighted average based on the edge weights defined by the authority transfer schema graph. Specifically, scores are calculated by applying the Equation 1 to the authority transfer data graph constructed as above. Equation 1 is the same as the one employed by PageRank, which uses the random surfer model

$$\mathbf{r} = d\mathbf{A}\mathbf{r} + \frac{(1-d)}{|V|}\mathbf{e} \tag{1}$$

where $\mathbf{r}$ is the ObjectRank score vector, d is the probability of random jump, and $\mathbf{A}$ is the transition matrix. Note that the element a_{ij} of the transition matrix $\mathbf{A}$ is the weight of the edge from node i to node j if it exists, otherwise, a_{ij} is 0.

3 Twitter User Rank (TURank)

In this section, we describe a scheme to evaluate Twitter users' authority scores based on the link structure analysis on the *user-tweet graph*. It reflects both Twitter social graph and actual information flow among users. We call this scheme Twitter User Rank (TURank).

3.1 Basic Idea

In Twitter, there are several relationships between users and tweets, such as post, follow, and RT. We take these relationships into account for measuring the authority score. Our scheme is based on the following observations:

- A user followed by many authoritative users is likely to be an authoritative user.
- A tweet retweeted by many authoritative users is likely to be a useful tweet.
- A user who posts many useful tweets is likely to be an authoritative user.

Based on these observations, we construct the user-tweet graph in which user nodes and tweet nodes are interconnected by edges corresponding to the relationships among users and tweets. A user-tweet graph allows us to understand how information spreads among users by RTs. Then, the link structure analysis is applied to this graph to calculate authority scores based on ObjectRank. The remainder of this section describes these steps in detail.

3.2 User-Tweet Graph

To construct a user-tweet graph, we define a *user-tweet schema graph*, as illustrated in Figure 3. It corresponds to the authority transfer schema graph in ObjectRank. A user-tweet schema graph $UTG_S = (V_S, E_S)$ defines the structure and edge weights of a user-tweet graph. Here, V_S is the node set consisting of user nodes and tweet nodes, and E_S is the edge set consisting of post, posted, follow, followed, RT, and RTed edges. A post edge is from a user u to a tweet posted by u. A follow edge is from a user u to a user followed by u. An RT edge is from a tweet t to a tweet retweeted by t. Posted, followed, and RTed edges are the reverse edges corresponding to post, follow, and RT edges. The weight $w(e_S)$

Fig. 3. A User-Tweet Schema Graph

is set to edge $e_S \in E_S$ by hand in order to reflect the semantics of each edge. Scores flow in accordance with the amount of weights from a node to another. The weight is shown beside the edge type in Figure 3.

A user-tweet graph $UTG = (V, E)$ is a graph corresponding to authority transfer data grpah in ObjectRank, and is derived from its corresponding user-tweet schema graph using actual data obtained from Twitter (Figure 4). Here, V is the node set which contains all tweet nodes and user nodes in obtained data. E is the edge set which contains all existing edges in obtained data, namely, post, posted, follow, followed, RT, and RTed edges. The weight $w(e)$ set to the edge $e \in E$ from node $u \in V$ is calculated by Equation 2.

$$w(e) = \frac{w(e_S)}{OutDeg(u, e_S)} \tag{2}$$

where $e_S \in E_S$ is the edge of the same type as e, and $OutDeg(u, e_S)$ is the number of outgoing edges of type e_S from node u.

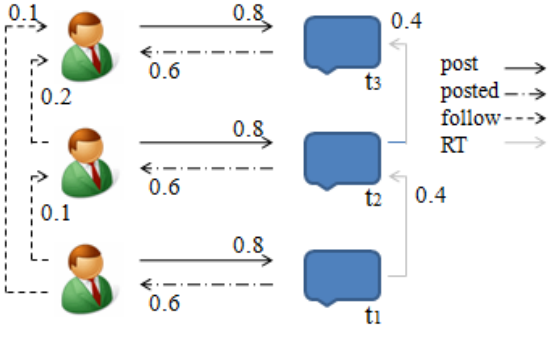

Fig. 4. A User-Tweet Graph

Although it looks quite simple, a user-tweet graph successfully represents important relationships in Twitter. If the graph consists of user nodes and follow edges, or alternatively tweet nodes and RT edges, it is almost the same as the graph for PageRank. However, in the case of a user-tweet graph, it has the edges between users and tweets, thereby making it possible to reflect the score flow from a user to its related tweets and vice versa. Specifically, scores of tweet nodes aggregated from RT edges are delivered to their authors through posted edges. Likewise, scores of user nodes aggregated from follow edges are delivered to their tweets through post edges. Therefore, we can calculate scores of both users and tweets concurrently. Moreover, a user-tweet graph is successful in capturing the RT chains appropriately. For example, as illustrated in Figure 4, when tweet t_1 retweets t_2, and t_2 retweets t_3, the score of t_1 affects t_3 through t_2. Of course the score of t_2 also affects t_3. In this way, scores of all tweets participating the RT chain affect the original tweet of the chain.

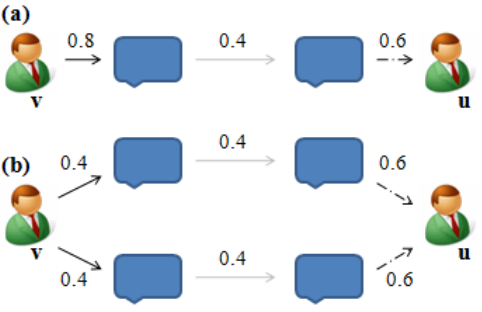

Fig. 5. Multiple RTs by the Same User

Even if user v retweets user u's tweets multiple times, u does not get severalfold scores from v. Because scores travel in accordance with the normalized weight as illustrated in Figure 5. In the case of (b), user u is retweeted by v twice. However, the same amount of score from v transfers to u as the case of (a), because the weights of the RT edges are normalized. This means that the user whose tweets are retweeted multiple times by only one user does not get a high score, while the user whose tweets are retweeted multiple times by different users gets a high score.

3.3 Calculating Scores

Having obtained a user-tweet graph, we then try to compute the scores of each user for subsequent ranking. The score calcuration is based on the Equation 1. Here, if edge e from node i to node j exists, element a_{ij} of $\mathbf{A}$ is $w(e)$, otherwise, a_{ij} is 0.

TURank
  $r^0 \leftarrow [1, \cdots, 1]$
  $\alpha \leftarrow 0$
  Repeat
    $\alpha \leftarrow \alpha + 1$
    foreach $r_i^\alpha \in r^\alpha$
      $r_i^\alpha \leftarrow \sum_{e=(j,i)\in E} w(e) r_j^{\alpha-1} + (1-d)\,/\,|V|$
    end
    $r^\alpha \leftarrow r^\alpha \,/\, ||r^\alpha||_1$
  until $||r^\alpha - r^{\alpha-1}||_1 < \epsilon$
  return r^α
end

Fig. 6. TURank Algorithm

Figure 6 shows the algorithm to calculate TURank scores in detail. The score of node i in step α is calculated by summing up scores of all nodes which have the edge to i in step $\alpha - 1$ and scores by random jump. This calculation is iterated until all scores are converged, where convergence threshold ϵ is set to be sufficiently small. Using this algorithm, we can measure the authority scores based on the link structure of the relationships among users and tweets.

4 Experimental Evaluation

This section shows experimental evaluation of the proposed TURank to show that the proposed algorithm outperforms other conventional algorithms. The dataset used in this experiment is explained in Section 4.1, and Section 4.2 presents the methodology of this experiment. Lastly, we discuss experimental results in Section 4.3.

4.1 Dataset

The dataset used in our experiment is crawled from Twitter using Twitter API [2] from January 26 to 28 in 2010. From the raw data, we extracted the following data: $D = (T, U, P, F, R)$, where T is the tweet set consisting of Japanese tweets which retweet other tweets or are retweeted by other tweets; U is the user set containing users who post tweet $t \in T$; P is the post edge set containing post edges from user $u \in U$ to a tweet posted by u; F is the follow edge set containing follow edges from user $u \in U$ to a user followed by u; and R is the RT edge set containing RT edges from tweet $t \in T$ to a tweet retweeted by t. The details of the dataset is shown in Table 1.

Table 1. Dataset details

	size
# of tweet nodes $\|T\|$	605,968
# of user nodes $\|U\|$	112,035
# of post edges $\|P\|$	605,968
# of RT edges $\|R\|$	369,383
# of follow edges $\|F\|$	14,631,014

Unfortunately, Twitter API does not provide information about RT edges between a retweet and its original tweet. For this reason, we extracted the correspondence information in the following way. At first, we extract the original user id which corresponds to the user who posted the original tweet, and the original text from a retweet, which is formed as shown in Section 2.1. Next, to identify the original tweet, we look into the latest 50 tweets posted by the original user. Actually, we compute the Levenshtein (edit) distance [10] between the original text and the latest 50 tweets, and find the tweet with the least distance which is less than the predefined threshold.

Here, the Levenshtein distance [10] is a metric that measures the similarity between two strings. It is defined as the minimum number of edit operations needed to transform one string into the other. Edits include deletion, insertion, and substitution of a character. For example, the Levenshtein distance between *apple* and *play* is 4:

1. *pple* (delete *a*)
2. *ple* (delete *p*)
3. *pla* (substitution of *a* for *e*)
4. *play* (insert *y* at the end)

It is possible to set the different cost to each edit. In the example above, the costs of all edits are 1. When cost 1 is set to insertion and deletion and cost 2 is set to substitution, the Levenshtein distance between *apple* and *play* is 5.

In the procedure of identifying the correspondence between retweets and their original tweets, we assign a larger cost for insertion than that of deletion, because users tend to retweet shortened messages by deleting some of the original texts due to the strict limitation of message length (140 characters).

4.2 Methodology

We compare 8 ranking schemes, including FollowNum, RTNum, PageRank, HITS [8], and 4 variants of TURank. FollowNum ranks users according to the number of followers. RTNum ranks users by the number of RTs. For PageRank and HITS, we apply each algorithm to the Twitter social graph, and rank the users according to the scores. For TURanks, we use the user-tweet graph weighted as shown in Table 2. The weights of followed edges in all TURanks are set to 0, because the authority score should not flow from users to their followers. Weights of other types of edges are varied in order to compare these TURanks and analyze how the weights work.

Table 2. TURank Weights

	follow	followed	post	posted	RT	RTed
TURank1	0.4	0.0	0.6	0.6	0.4	0.0
TURank2	0.2	0.0	0.8	0.6	0.4	0.0
TURank3	0.2	0.0	0.8	0.4	0.6	0.0
TURank4	0.2	0.0	0.8	0.6	0.2	0.2

The obtained rankings are evaluated by 34 examinees. We show top 25 authoritative users, and examinees evaluate the adequacy of the authority score of each user by scoring 1 to 5 by browsing his/her latest 100 tweets. In this experiment, an authoritative user is defined as a user who posts either some breaking news stories attracting many users, interesting or humorous tweets, or ideas about some topics which are regarded as useful by many users.

Average adequacy of evaluated the top k authoritative users is shown in Figures 7 and 8. Figure 7 compares all algorithms, and Figure 8 compares 4 TURanks using the graph weighted as shown in Table 2.

4.3 Discussions

Figure 7 shows that our algorithm successfully suggests relatively high adequacy for the entire range. TURank outperforms both PageRank and HITS in that only *follow* relations are considered. The result suggests the importance of considering tweet/retweet relations, thereby making it possible to model information flow in Twitter. On the other hand, FollowNum and RTNum indicate low adequacy, probably because the number of followers or RTs alone is not adequate.

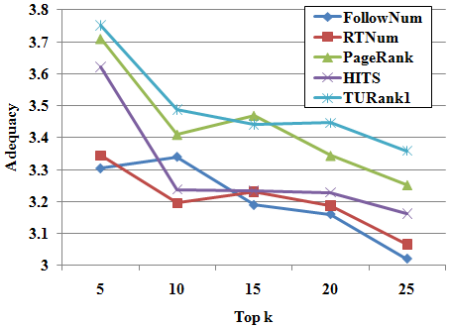

Fig. 7. Average Adequacy of Top k Users of All Algorithms

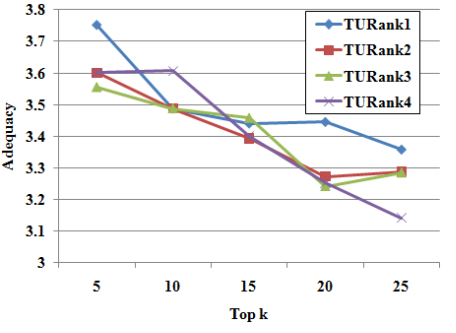

Fig. 8. Average Adequacy of Top k Users of TURanks

TURank shows more effectiveness than RTNum which only uses the number of RTs. Specifically, RTNum tends to extract users who use RTs for conversation in higher ranks. These users often use replies of the form @[*user id*] in order to specify the user to which he/she replies. In these cases, tweets do not convey useful information, and actually examinees evaluated most of them as useless.

The total number of replies obtained by the top 100 users of RTNum is 1753, while the one of TURank is 1241. For this reason, we observe that only counting the number of RTs is not sufficient to capture the semantics of RTs; RTNum do not discriminate the difference between conversational RTs and recommending RTs. In contrast, TURank is successful in discriminating the difference. TURank uses the user-tweet graph which represents how information is disseminated via RT chains. In this case, recommending RTs, which spread widely, gain larger scores, while conversational RTs, which spread across limited users, gain smaller scores.

The reason why TURank outperforms PageRank and HITS is as follows. TURank ranks users, whose tweets are retweeted many times, in higher ranks, and such users are regarded as authoritative users. If we only use *follow* relations, we may fail to rank these users higher. In contrast, assume a user who has many followers, but his/her tweets are not retweeted many times. Such users do not get higher ranks by TURank, and categorized as unauthoritative users. In Twitter, one wants to follow a user, if the user is likely to transmit useful information even without guarantee. In many cases followers do not stop following even if he/she turns out not to transmit useful information any more. This is partly because one does not completely remember all users whom he/she is following, and because the number of following user accounts can be very large (100 to more than 1,000) in many cases. Even if a user has a lot of followers in this way, his/her tweets may be rarely retweeted, which result in lower rankings.

The results also show the problem of TURank. In fact, there are some authoritative users, who have a lot of followers, but their tweets are rarely retweeted. Even in that case, TURank lowers such users' scores. For example, although the weather forecasting bot[1] is evaluated as very useful by examinees, TURank ranked it in lower ranks. This bot acquires a lot of followers, but the followers do not want to retweet its tweets. Actually, the followers of this bot is satisfied by receiving weather forecasts, but do not have a strong motivation to disseminate the forecasts to their followers. Therefore, proper evaluation of such user accounts is an important future issue.

Figure 8 shows that TURank1 is the best among all TURanks. In Twitter, as shown in Table 1, the number of RT edges is much less than follow edges. For this reason, if the weight of RT edges is larger than necessary, the sparse part of the user-tweet graph is weighed heavily, and consequently, TURank is unsuccessful in evaluating the users.

TURank1 shows a better results, because it is based on a proper setting of edge weights. Comparing TURank2 with TURank3, there are some differences between them. Setting larger weights to RT edges, a tweet which is retweeted many times obtains a higher score. For this reason, a user who posts such a tweet ranks higher position of TURank2 than its position of TURank3. Moreover, setting smaller weights to posted edges, smaller scores transfer from tweets to users through posted edges. Hence, scores of users totally decrease, and as a consequence, follow edges convey smaller amount of scores. However, evaluated effectiveness of these two rankings are almost the same in this experiment, because the difference of weights between TURank2 and TURank3 is not large enough. The evaluated result of TURank4 shows a little lower effectiveness than others. Because the weight of the RTed edge is not 0, it is expected that the user who often retweets useful tweets moves up on the list of TURank4. However, as a result, only two useless bots[2] entered the upper level in TURank4.

[1] A *bot* is a user who posts automatically by a program.

[2] Two bots are @yumemitter and @soysaucebot. The former is the bot which randomly retweets the tweet about dreams posted by other users, and the latter is the bot which randomly retweets the tweet about the soy sauce posted by other users.

5 Related Work

In recent years, many researchers have investigated Twitter. Java et al. [7] studied the topological and geographical properties of Twitter analyzing the social graph composed of *follow* relationships. In addition, they investigated the community formed by Twitter users at large, and then divided users into three categories. Honeycutt et al. [5] analyzed the conversational practice in Twitter, such as replies and RTs. Huberman et al. [6] reported that the social graph of Twitter does not describe relationships among users well. Besides, they indicated that there is the non-dense social graph composed of *friends* relationships, which cannot be illustrated by *follow* relationships. Boyd et al. [4] conducted the usage survey of RTs. They interviewed Twitter users in order to analyze the syntaxes of RTs, the purposes of RTs, and the targets of RTs.

Weng et al. [12] and Leavitt et al. [9] proposed the algorithm which ranks Twitter users. TwitterRank [12] measures the users' influence considering the link structure of *follow* relationships, the similarity between users, and the number of posts. TURank differs from TwitterRank in two ways: TURank takes RTs into accounts which is not considered by TwitterRank, and is the graphically-based algorithm which does not use the content of tweets. Leavitt et al. insisted that measuring the influence by only *follow* relationships is inadequate and take the communication such as RTs and replies into consideration. Although this research is close to our research, this research does not consider the link structure of the graph composed in Twitter.

6 Conclusions

This paper presented a scheme for identifying authoritative users in Twitter. Considering the actual information flow in Twitter, we proposed the TURank which is an algorithm to measure the authority scores and rank users applying the link structure analysis to the *user-tweet graph*. The user-tweet graph is constructed from user nodes and tweet nodes linked by post, follow, and RT edges in order to model how information flows and spreads among users. In spite of its simple structure, this graph describes these relationships well enough. Moreover, this graph appropriately represents RT chains and multiple RTs by the same user.

In this paper, we demonstrated the effectiveness of the proposed TURank. As a result, we showed that TURank can extract users, who are not followed by many users, but his/her tweets are retweeted for many times, with higher position in the ranking, when all edge weights in the user-tweet schema graph are appropriately defined. Second, users whose tweets are not retweeted many times tend to be regarded relatively useless by examinees, even if they have a large number of followers. Our algorithm can lower such users' authority scores. Lastly, users, most of their tweets are for conversation, are evaluated as completely useless by examinees. Our algorithm can successfully lower their authority scores.

Authoritativeness is differentiated by topics. For instance, a user who knows a lot about tennis and less about baseball is authoritative in the area of tennis but

not in the area of baseball. We did not consider this topic-wise authoritativeness. However, the proposed algorithm can incorporate this type of authoritativeness easily, because the user-tweet graph includes tweet nodes. Each tweet has its topic, so we can measure each user's topic-wise authoritativeness analyzing the topic of each tweet and delivering it to its authors through the posted edges.

Tweets are posted at a furious speed as a stream of data, and hence the user-tweet graph grows dynamically. To evaluate the authority scores in such environments, we should consider this dynamic property of Twitter in the future.

Acknowledgement

This research has been supported in part by the Grant-in-Aid for Scientific Research from MEXT (#21013004).

References

1. Twitter, `http://twitter.com`
2. Twitter API, `http://apiwiki.twitter.com/Twitter-API-Documentation`
3. Balmin, A., Hristidis, V., Papakonstantinou, Y.: Objectrank: Authority-based keyword search in databases. In: VLDB (2004)
4. Boyd, D., Golder, S., Lotan, G.: Tweet, tweet, retweet: Conversational aspects of retweeting on twitter. In: HICSS-43, January 6. IEEE, Kauai (2010)
5. Honeycutt, C., Herring, S.C.: Beyond microblogging: Conversation and collaboration in twitter. In: Proc. 42nd HICSS. IEEE Press, Los Alamitos (2009)
6. Huberman, B.A., Romero, D.M., Wu, F.: Social networks that matter: Twitter under the microscope. First Monday 14(1) (January 5, 2009)
7. Java, A., Song, X., Finn, T., Tseng, B.: Why we twitter: Understanding microblogging usage and communities. In: Joint 9th WEBKDD and 1st SNA-KDD Workshop, San Jose, CA (2007)
8. Kleinberg, J.: Authoritative Sources in a Hyperlinked Environment. In: Proc. of the 9th ACM SIAM Symposium on Discrete Algorithms (SODA 1998), pp. 668–677 (1998)
9. Leavitt, A., Burchard, E., Fisher, D., Gilbert, S.: The influentials: New approaches for analyzing influence on twitter. A Publication of the Web Ecology Project (2009)
10. Levenshtein, I.V.: Binary codes capable of correcting deletions, insertions, and reversals. Cybernetics and Control Theory 10(8), 707–710 (1966)
11. Page, L., Brin, S., Motwani, R., Winograd, T.: The pagerank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project (1998)
12. Weng, J., Lim, E., Jiang, J., He, Q.: Twitterrank: Finding topic-sensitive influential twitterers. In: WSDM (2010)