Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2015, Article ID 876218, 11 pages
http://dx.doi.org/10.1155/2015/876218

*Research Article*

# Weibo Information Propagation Dissemination Based on User Behavior Using ELM

**Huilin Liu and Yao Li**

*College of Information Science and Engineering, Northeastern University, Shenyang, Liaoning 110819, China*

Correspondence should be addressed to Huilin Liu; liuhuilin@mail.neu.edu.cn

Received 21 September 2014; Revised 1 January 2015; Accepted 16 January 2015

Academic Editor: Tao Chen



Information dissemination prediction based on Weibo has been a hot topic in recent years. In order to study this, people always extract features and use machine learning algorithms to do the prediction. But there are some disadvantages. Aiming at these deficiencies, we proposed a new feature, the dependency between the Weibos involved in geographical locations and location of the user. We use ELM to predict behaviors of users. An information dissemination prediction model has also been proposed in this paper. Experimental results show that our proposed new feature is real and effective, and the model we proposed can accurately predict the scale of information dissemination. It also can be seen in the experimental results that the use of ELM significantly reduces the time, and it has a better performance than the traditional method based on SVM.

## 1. Introduction

With the development of the web 2.0, social networks have become an indispensable part of people's lives. Large social networking site like Facebook, Twitter, and so forth brings a lot of happy time to people. Sina Weibo, as one of China's largest online social networks, has more than 500 million registered users. Every day these users produce a lot of social network data through continuously released and forwarded microblogging. These social network data researches help enterprises and government find the users network behavior rules and make the corresponding measures. Thus, the study of Weibo is a hot issue in recent years.

There are a lot of directions on the study of Weibo, including sentiment analysis based on Weibo [1] and Weibo personalized recommendation research [2]. One high practical value direction of the researches in Weibo is studying online behavior of users and corresponding information propagation. This aspect of the study can help enterprises to understand the user behavior mode, grasp the user interest preference and recommend the interest topics, other users, and groups to the user. It can also help the government to understand the range of the spread of news, judge social public opinion direction and reactions, and adjust corresponding policies in time.

There are many researches about user behavior in online social networks and information dissemination exists. One of the common methods is extracting user behavior characteristics and use machine learning algorithm to classify and predict user behavior [3–6]. In general, the researchers adopt support vector machine (SVM) algorithm. The features they widely use are the influence of user, the intimacy between the users, the interest similarity of user, Weibo content importance, and so forth.

In life, people are more concerned about the information around their side. This can also be extended to Weibo. So if a Weibo involves the geographical location, the users who are near the location will pay more attention to the Weibo than users in other areas. Although there are a lot of social network applications which use the geographical position, for example, Lingad et al. [7] studied the extraction of Weibo position related to the disaster, Hosseini et al. [8] studied location oriented phrase detection in microblogs. But on the analysis of user online behavior and information dissemination, the dependency between the geographical locations in which Weibos are involved and location of the user has not been mentioned.

Therefore, on the basis of summarizing the work before, we take the dependency between the geographical locations in which Weibos involved and location of the user as a new

feature to analyze user behavior and information dissemination. At the same time, because extreme learning machine algorithm runs fast and can get the optimal solution rather than the sub-optimal solutions, we adopt ELM to replace SVM. The main contribution of this paper is shown as below.

(1) We propose a new feature, the dependency between the geographical locations in which Weibos involved and location of the user. We use this feature and other proposed feature to analyze user behavior and information dissemination.

(2) We test the different performance between the different value of $\Delta t$ in ignore dataset and found that when $\Delta t$ is 30 minutes, the performance is the best.

(3) We use ELM instead of SVM to predict user behavior and information dissemination.

Our experimental results show that, with the new feature we proposed, we get a higher forecasting accuracy than without the new feature. Our experimental results also show that ELM gets higher accuracy than SVM in the same dataset.

The rest of this paper is organized as follows. Section 2 briefly introduces the related work about online social network and ELM. Section 3 introduces the data and feature we use to predict user behavior and the information dissemination model. And the experimental results are reported in Section 4. Finally, we present our conclusions and future work in Section 5.

## 2. Related Work

*2.1. Online Social Network.* Due to the popularity of social networks, there are many studies of social networks. For example, Marques and Serrão [9] proposed using rights management systems to improve the content privacy of social network users; Quang et al. [10] found the cluster of actors in social network based on the topic of messages; Tseng and Chen [11] proposed incremental SVM model to detect unwanted email, and so on.

Our main work in this paper is analyzing user behavior and information dissemination. There are a lot of related works of this aspect. Song et al. [3] proposed 4 features to predict if user will forward the Weibo or ignore it. The features are the authority of user, the activity of user, the preference of user, and the social relations of user. The four features can reflect the user behavior to a certain extent, but they did not consider the importance of Weibo content and the dependency between the geographical locations, which are involved Weibos, and locations of the user. Zaman et al. adopted the model of collaborative filtering based on probability [12, 13]. They select the user name, the number of attention, and number of words that Weibo contains to predict the forward behavior of user. Although these features have some influence on user behavior and information dissemination, these features are not the main factor affecting the user's behavior. Cao et al. [4] improved the prediction model, added the Weibo content length, Weibo importance, whether the user is authenticated user, and some other features. The added features improved the prediction accuracy of user behavior and information dissemination, but they still did not consider the relationship between Weibo mention place names and users.

Some other works also give us some help. For example, some people analyzed the flow of information within the scope of the blog and made a prediction model of information transmission in [6]. Sina Weibo and the traditional blog have certain similarities. We can draw lessons from the spread of the blog. Webberley et al. [14] studied the transmit delay, the depth and breadth of information dissemination on Twitter. They preliminary studied user behavior patterns and forwarding rules and have certain reference significance.

Some researchers have studied the influence of mentioned location on information dissemination. For example, Bandari et al. [15] put forward an algorithm to predict whether the news is popular enough on Twitter or whether it can trigger a heated discussion on social networking sites. This paper puts forward four features: article categories, the degree of objective, the article mentioned geographical name and people name, and the sources of article. But the study only gives the effect of the popular places to information dissemination, does not take the dependency between the geographical name and users into account.

In conclusion, we propose a new feature: the dependency between the geographical locations, which are involved Weibos, and locations of the user.

*2.2. ELM.* Extreme Learning Machine (ELM) is put forward by Huang at Nanyang technological university in 2004 [16]. It is a more simple and effective algorithm of single hidden layer feed forward network (SLFNs) algorithm. It can automatically choose the input weight and analyze decision output weight. It provides the best generalization ability and very fast learning speed. Huang has proved in Extreme Learning Machine a New Learning Scheme of Feed forward Neural Networks, that under the same condition of the classification, ELM rate is much higher than the SVM. According to Professor Huang previous studies [17, 18], we summarize the ELM theory is as follows.

For $n$ different samples $(x_i, t_i)$, $x_i = [x_{i1}, x_{i2}, \ldots, x_{in}]^T \in R^n$ and $t_i = [t_{i1}, t_{i2}, \ldots, t_{im}]^T \in R^m$. If the SLFNs has $\widetilde{N}$ hidden nodes and its activation function is $g(x)$, then we get the formula as follows:

$$\sum_{i=1}^{\widetilde{N}} \beta_i g_i\left(x_j\right) = \sum_{i=1}^{\widetilde{N}} \beta_i g\left(w_i \cdot x_j + b_i\right) = o_j, \quad j = 1, 2, \ldots, N. \tag{1}$$

In the formula, $w_i = [w_{i1}, w_{i2}, \ldots, w_{in}]^T$ is the weight vector which connects $i$th hidden node with the input vector. $\beta_i = [\beta_{i1}, \beta_{i2}, \ldots, \beta_{im}]^T$ is the weight vector which connects $i$th hidden node with the output vector. $b_i$ is the threshold of the $i$th hidden node. $w_i \cdot x_j$ is the inner product of $w_i$ and $x_j$.

The $N$ samples approximate to zero mean error, so we have $\sum_{j=1}^{\widetilde{N}} \|o_j - t_j\| = 0$; then, we get the formula as follows:

$$\sum_{i=1}^{\widetilde{N}} \beta_i g\left(w_i \cdot x_j + b_i\right) = t_j, \quad j = 1, 2, \ldots, N. \tag{2}$$

The above formula can be written into $H\beta = T$. Then, the process of ELM can be mathematically modeled as the following formula:

$$\text{Minimize: } \left\|H\beta - T\right\|^2, \ \left\|\beta\right\|. \tag{3}$$

Here, $H$ can be expressed as

$$H\left(w_1, \ldots, w_{\widetilde{N}}, b_1, \ldots, b_{\widetilde{N}}, x_1, \ldots, x_{\widetilde{N}}\right) = \begin{bmatrix} g\left(w_1 \cdot x_1 + b_1\right) & \cdots & g\left(w_{\widetilde{N}} \cdot x_1 + b_{\widetilde{N}}\right) \\ \vdots & \cdots & \vdots \\ g\left(w_1 \cdot x_N + b_1\right) & \cdots & g\left(w_{\widetilde{N}} \cdot x_N + b_{\widetilde{N}}\right) \end{bmatrix}_{N \times \widetilde{N}}. \tag{4}$$

Therefore, we get a solution for the parameter $i$ as

$$\widehat{\beta} = H^{\dagger} T, \tag{5}$$

where $H^{\dagger}$ is the Moore-Penrose generalized inverse of matrix $H$.

Based on the above analysis, the machine learning-based algorithm without iterative tuning can be divided into three steps. The specific process of ELM is summarized as follows.

*Step 1.* Randomly assign input weight $w_i$ and bias $b_i$, $i = 1, 2, \ldots, N'$.

*Step 2.* Calculate the hidden layer output matrix **H**.

*Step 3.* Calculate the output weight $\beta$, where $\beta = H^{\dagger} T$.

Compared with SVM, ELM can be directly applied in many kinds of classification problems. In professor Huang Extreme Learning Machine for Regression and Multiclass Classification study, he has proved that the SVM obtains sub-optimal solution and needs higher computational complexity [19]. Therefore, ELM has the advantages that SVM does not have and has a broad application prospect.

## 3. User Behavior and Information Dissemination Prediction

In this paper, we analyze people's behavior and information dissemination on Weibo. First of all, we need to get the data from Sina Weibo. The behaviors of users in Sina Weibo are releasing, browsing, commenting, and forwarding. Release and forward behaviors are associated with information dissemination. However, the release behavior is decided by users self and we cannot control it. So our main study is forward behavior of users.

In this section, we will introduce the data and features we use and give the information dissemination prediction model we proposed. First of all, we give the dataset description.

*3.1. Dataset Description.* When we get the Sina Weibo data, first of all, we choose one user and get its fans list. Second according to the fans list, we get the fans list of each user in fans list. In this method, finally we get a user's dataset. We got 96438 users in this dataset. Sina Weibo users can be roughly divided into three categories: release active users, forward active users, and inactive users. If a user does not have forward or release activity in 1 month, we think it is an inactive user. Because the inactive users do not have any contribution to the user behavior and information dissemination prediction, so we excluded these users. Finally we got 89377 users in the dataset. Then, we crawl all Weibos of these users which published between May 1, 2014, and May 31, 2014, and get 564835 Weibos. In these Weibos, there are 114943 Weibos related to geographical locations. Most of the Sina Weibos are Chinese Weibos, the geographical locations in them are Chinese location. So the small amount of Weibos which contain foreign geographical locations are consider to have nothing to do with the geographical location. We select the data from the whole Weibo dataset to build forward and ignore datasets. Because we cannot see the ignore behavior directly, we need to define the ignore dataset first. The definition of ignore dataset shown as follows.

*Definition 1* (ignore dataset). If user $u$ forwarded the Weibo published at time $t$, the Weibos which published by the friends of the user at $[t - \Delta t, t + \Delta t]$ and are not forwarded by the user are the ignore samples. All the ignore sample constitute ignore dataset.

Users ignore the Weibos not only because users do not like them, but also because they are leaving and not seeing the Weibos. So we selected 10 minutes, 30 minutes, 1 hour, 2 hours, and 12 hours as $\Delta t$. We also studied influence of different ignore datasets to the final accuracy. Algorithm 1 is used to find ignore dataset.

In order to facilitate our location keywords extraction, we established the province tree to identify the place name. Figure 1 is the structure of the province tree.

As we can see in Figure 1, China, according to the position, is divided into east China, south China, central China, north China, northwest, southwest, and northeast. Each region contains some provinces, and each province contains a number of cities. According to the province tree, we can identify the key word belonging to which geographical locations. We can also get the subordinate situation of the key word.

In province tree, we only consider the city name, without regard to the block name. This is because, in China, different city may contain the same blocks name. We cannot be able to accurately determine the block belongs to which city.

Our study is based on the above data. In the next section, we will introduce the features we use and the corresponding evaluation index.

*3.2. Feature Description.* In this section, we will introduce the features we use. First of all, we will introduce the new feature we proposed. And then we will introduce other features we use.

Inputs: Weibos set $P$ which published by the friends of the user $u$;
Weibos set $Q$ which user $u$ forward.
Output: Weibos set $R$ which user $u$ ignore.
(1) Any Weibo $m$, $m \in Q$, read the publish time $t_m$;
(2) Find Weibo $w$, $w \in P$
(3) while (the publish time of $w$ satisfy $t \in [t_m - \Delta t, t_m + \Delta t]$)
(4) $w \in S$; //$S$ is an intermediate variable.
(5) While ($\forall w$, $w \in S$, $w \notin Q$)
(6) Add $w$ to $R$;
(7) Output $R$;

Algorithm 1: Ignore($P$, $Q$).

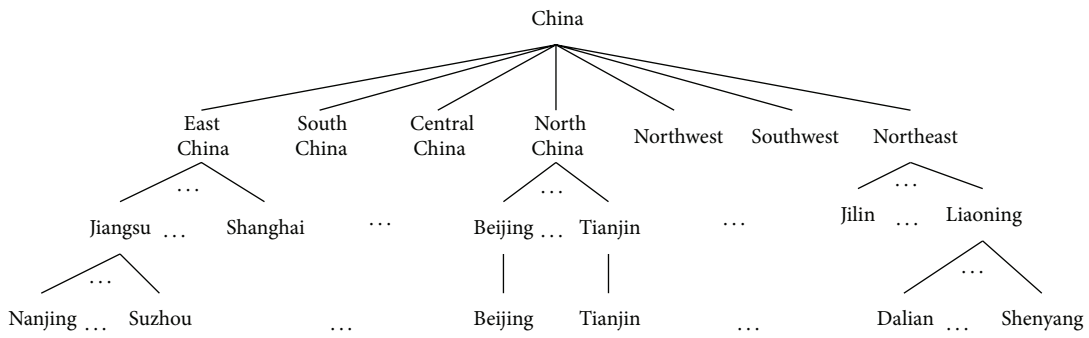


Figure 1: The structure of the province tree.

*3.2.1. The Dependency between the Weibos Involved Geographical Locations and Location of the User.* The Weibo involved geographical locations have been proposed before. However, they only concern whether the location name is famous and do not connect it with the locations of users. As the government starts carrying out internet political communication on Weibo, this connection becomes more and more important. Information published by the local government is likely to be paid attention to in the local and surrounding areas. The further area users will give less attention to it. We use Peking University PKUVIS Weibo visual analysis tools [20] to analyze 150 Weibos and one of it is shown as follows:

#毛絮是虫子#【😭 南京满天飞的“毛絮”竟是长着白毛的虫子!】@现代快报: 这两天, 南京一些地方飘着柳絮一样的东西, 漫天飞舞。 南京林业大学森环院的专家发现, 其实它们根本不是柳絮, 而是活物小虫子!! 叫“榆四脉绵蚜”! 今年的气候有利于它们的繁殖, 所以数量非常多! 😅 我整个人都不好了!

In this Weibo, we can extract the location name Nanjing. According to the province tree, it belongs to Jiangsu province. We guess the users in Jiangsu may have high attention in this Weibo. The users far from Jiangsu may pay less attention. So we count users number in every province who forward this Weibo. According to the province field of the data, we obtained the province of these Weibos users. Sina Weibo use code to represent the provinces and cities. Table 1 shows the provinces and its corresponding code. For convenience, in the following figure, we all use the province codes in Table 2 to represent the province. Figure 2 shows the number of users in every province.

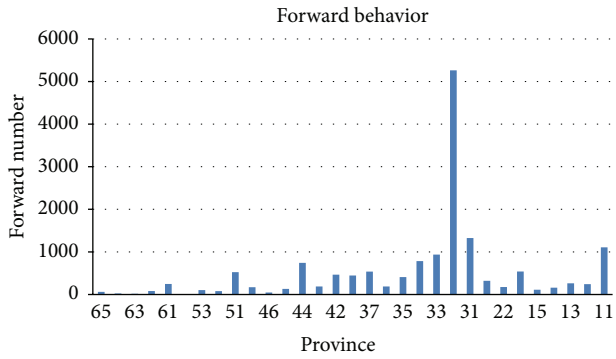


Figure 2: The number of users in every province.

We can see in Figure 2, the local users in Jiangsu pay the most attention to the Weibo. The locations which near Jiangsu also pay much attention to it (like Anhui, Shanghai, Zhejiang, and Shandong).

According to the theory of probability, to other provinces and cities, the forwarding quantity percentage should have the same regularity with the registered users' percentage in each province. It is hard to get the registered users' percentage. But in Figure 1 we can see the economically developed provinces, such as Beijing and Guangzhou, have higher forward number than some underdeveloped areas like

Table 1: Province and its code.

| Provinces | Beijing | Tianjin | Hebei | Shanxi | Inner Mongolia | Liaoning | Jilin |
|---|---|---|---|---|---|---|---|
| Code | 11 | 12 | 13 | 14 | 15 | 21 | 22 |
| Provinces | Heilongjiang | Shanghai | Jiangsu | Zhejiang | Anhui | Fujian | Jiangxi |
| Code | 23 | 31 | 32 | 33 | 34 | 35 | 36 |
| Provinces | Shandong | Henan | Hubei | Hunan | Guangdong | Guangxi | Hainan |
| Code | 37 | 41 | 42 | 43 | 44 | 45 | 46 |
| Provinces | Chongqing | Sichuan | Guizhou | Yunnan | Tibet | Shaanxi | Gansu |
| Code | 50 | 51 | 52 | 53 | 54 | 61 | 62 |
| Provinces | Qinghai | Ningxia | Sinkiang | | | | |
| Code | 63 | 64 | 65 | | | | |

Table 2: Ignore samples.

| $\Delta t$ | Quantity |
|---|---|
| 15 minutes | 72996 |
| 30 minutes | 119392 |
| 1 hour | 188376 |
| 2 hours | 307483 |
| 12 hours | 431645 |

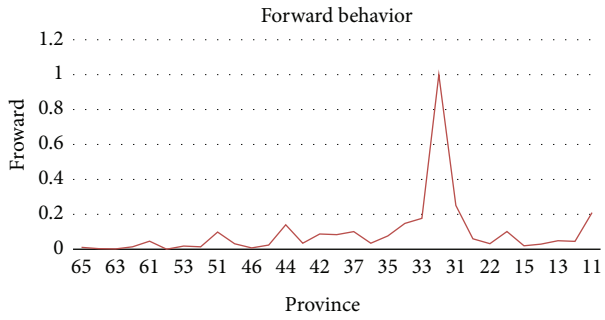


Figure 3: The normalization of forwarding number.

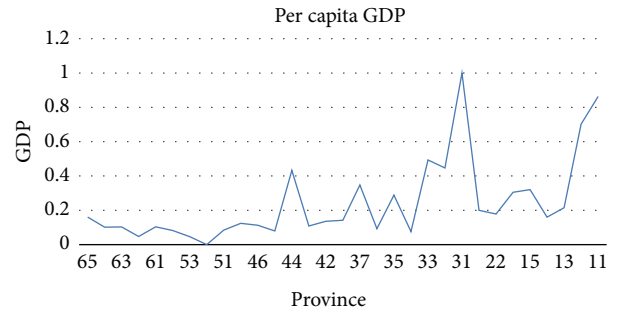


Figure 4: The per capita GDP in 2013.

Sinkiang and Ningxia. We guess this is because people in developed cities occupy more network resources and can easily get the website, so the users in developed cities may be larger than underdeveloped city. The other Weibos also have this rule.

To represent the cities' development, we found the per capita GDP in each province in 2013. Forward number and per capita GDP are not in the same magnitude. So we normalized these data. Figure 3 shows the normalized forward number. Figure 4 shows the normalized per capita GDP.

In Figures 3 and 4 we can see, in addition to geographical location mentioned in the Weibo, the forward quantity and the per capita GDP in other province are in the same regularity. For example, in Beijing, Guangdong and other regions, two figures both have a local peak. The geographical location mentioned in the Weibo makes this feature not obvious. This further proves that the geographical location mentioned in the Weibo has a stronger influence on the users who are close to it.

All the Weibos we tested have this conclusion. So we use the per capita GDP to represent the registered users' percentage. And then the per capita GDP can represent the possibility of users forwarding. When the province is the geographical locations involved Weibos, we add 0.5 to the per capita GDP, which means this geographical location plays a predominant role in the forwarding behavior. The final value represents the dependency between the Weibos involved geographical locations and locations of the users.

Besides the new feature we put forward, other features are widely applied to the user behavior analysis and Information Propagation Dissemination. Researchers in [3] selected 4 features to judge user forward behavior. The features are The User's Authority, User's Activity, User's Preference, and User's Social Relations. However, the user's authority is relevant to the user's forward behavior, but the correlation is weak. Researchers in [4] selected 15 features. But some features are covered by other features. For example, when we compute the PageRank, the user fan numbers are used. This kind of features is useless and should not be used in the user forward behavior prediction.

Another research RT to Win! Predicting Message Propagation in Twitter [21] divided features into two categories. There are 7 social features (i.e., number of followers, friends, statuses, favorites, number of times the user was listed, is the user verified, is the user's language English) and 7 tweet features (i.e., number of followers, friends, statuses, favorites, number of times the user was listed, is the user verified, is the user's language English).

To summarize the features in the above and other papers, we selected 5 features to forecast user forwarding behavior. They are the influence of user, user release activity, and forward activity, the intimacy between the users, the interest similarity between user and content or between users and Weibo content importance. The following is these features in detail.

*3.2.2. The Influence of User.* People always use PageRank to compute the influence of user [22]. The PageRank algorithm is used to measure the importance of specific pages relative to other pages in the search engine. The PageRank formula they use is shown as

$$\mathrm{pr}_i = \frac{1-q}{N} + q \sum_{j \in \mathrm{Follower}(i)} \frac{\mathrm{pr}_j}{\left|\mathrm{Friend}\left(j\right)\right|}. \tag{6}$$

In this formula, $\mathrm{pr}_i$ represent the PageRank value of user $I$, $\mathrm{Follower}\,(i)$ represents the fans list of user $I$, $\mathrm{Friend}\,(j)$ represents the collection of users that user $j$ pays attention to, $q$ is the damping coefficient, and $N$ is the total number of users.

*3.2.3. User Release Activity and Forward Activity.* Because of the different behaviors of the user, the user activity can be divided into two aspects, the user release activity and forward activity. The user release activity is the Weibo number published over a period of time. We can use formula (7) to compute it:

$$PA = \frac{n}{t}. \tag{7}$$

The $PA$ in formula (7) represents the Weibo number published over a period of time, $n$ is the total number of Weibo, $t$ is the unit time. In general, we set $t$ to 1 day.

The forward activity is percentage of users forwarding Weibo account for all published Weibo in one day. We use formula (8) to compute it:

$$RA = \frac{\sum_{i \in t} r_i}{\sum_{i \in t} p_i}. \tag{8}$$

$r_i$ is the number of users forwarding Weibo in $i$th day, $p_i$ is the number of users releasing Weibo in $i$th day, and $RA$ represents the forward activity. The higher the $RA$ is, the more active the users are. Users with high forward frequency play a bigger role in information dissemination.

*3.2.4. The Intimacy between the Users.* Because the forward behavior in Weibo can reflect the interaction between the users better, we compute the intimacy between the users by calculating the percentage of Weibo published by the upstream user in the forwarding Weibo of the user. The formula we use is

$$f_{uv} = \frac{n_{uv}}{n_u}. \tag{9}$$

In this formula, $n_{uv}$ represents the number of the Weibos of user $v$ which appears in the forward Weibo of user $u$. $n_u$ represents the total number of forward Weibo of user $u$.

*3.2.5. The Interest Similarity between User and Content or between Users.* Weibo can reflect the interests of users. The larger the interest similarity between user and content, the greater the chance user forward. The larger the interest similarity between user and upstream user, the greater the chance user forward. So we need to compute the interest similarity. Because the user's interest is the change over time, we need to analyze the Weibo which release time near a few days. Interest space is extracted from weibo, and the following is the process of compare.

(1) Collect user interest. We select a user and collect the user $m$ Weibo published nearly five days. These form the user interest space $I_s = \{s_i, 0 < i < m\}$. $I_s$ is the interest space of user $s$ and $s_i$ is the $i$th Weibo of user $s$.

(2) Participle. For Weibo in Chinese, we use the Chinese Academy of Sciences Chinese lexical analysis system ICTCLAS do the word segmentation [22]. For Weibo in English, we use space. We get the words level interest space $I_\omega = \{\omega_i\}$. $\omega_i$ is the $i$th word.

(3) Remove the stop. We remove the stop word and get the new words level interest space $I = \{\omega_i\}$.

(4) Repeat (2) and (3); we get the interest space of user $s$ $J_s = \{\omega_j\}$.

(5) For the two users $s$ and $s'$, we calculate the similarity of $J_s$ and $J_{s'}$. For the user $s$ and the content $t$, we calculate the similarity of $J_s$ and $J_t$. We use Jaccard formula to calculate the similarity [23]. The Jaccard formula is

$$\mathrm{Jacard}\left(J_s, J_{s'}\right) = \frac{J_s \cap J_{s'}}{J_s \cup J_{s'}}. \tag{10}$$

*3.2.6. Weibo Content Importance.* Usually if a Weibo contains significant events or popular information, the forward rate will be high. So the importance of Weibo content can help us analyze Weibo information dissemination. Based on computing weight of TF-IDF (term frequency inversed document frequency) algorithm on the text classification field, we calculate the importance of Weibo [24]. The thought of this algorithm is that in a specific document the higher the frequency of word appears in the document, the more important the word is; the lower the frequency of word appears in other document, the more important the word is. We can use formula (11) to calculate the importance:

$$tf\,(d) = n_\omega \times \log \frac{N}{n_d}. \tag{11}$$

In this formula, $d$ represents the word $d$ in the Weibo $\omega$, $n_\omega$ represents the number of $d$ appearing in $\omega$, $N$ represents the number of Weibo that Weibo set $W$ contains, and $n_d$ represents the number of Weibos containing $d$ in the Weibo set $W$. The TF-IDF of Weibo $\omega$ can be computed by adding the TF-IDF of all the word in $\omega$:

$$tf\,(\omega) = \sum_j tf\left(d_j\right). \tag{12}$$

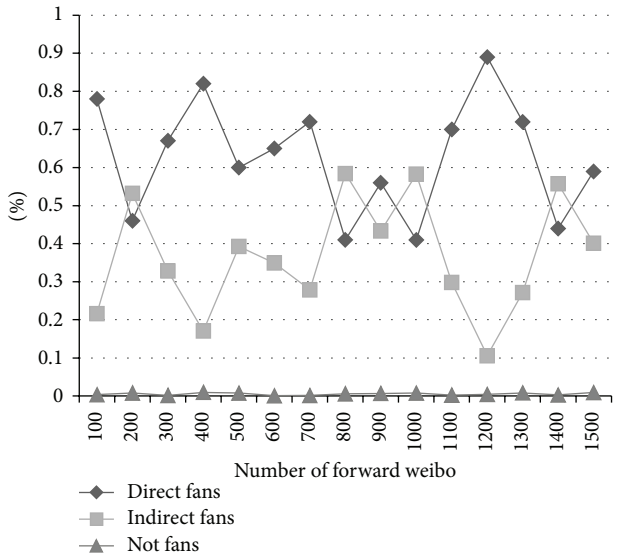


Figure 5: Each percentage of 3 forward behaviors.

*3.3. Information Dissemination Prediction Model.* According to the features in Section 3.2, we use ELM to forecast the user forward behavior. According to the predicted forward behavior, we forecast information dissemination scale.

The forward behaviors in Weibo can be divided into 3 aspects: direct fans forwarding, indirect fans forwarding, and not fans forwarding. We count the each percentage of 3 forward behaviors in different scales of Weibo. Figure 5 shows each percentage of 3 forward behaviors from the size of 100 to the size of 1500.

We can see from Figure 5 that forward behaviors are mainly composed of direct fans and indirect fans. The percentage of not fan users is almost 0. So we ignore the forward behaviors of not fan users.

When we make the prediction, we start from Weibo publishers. And then we traverse its list of fans and predict if the fan will forward the Weibo. If the fan forwards it, the forwarded number increases 1. Then, traverse the fans list of this user. We repeat iteration like this until no users forward the Weibo. The prediction model can be represented by a tree. Figure 6 is a simple example of prediction model tree.

The gray point in Figure 6 is the publisher of Weibo. The black points are the users who will forward the Weibo. The white points are the users who will not forward the Weibo. When we make the prediction, we start from the user $U_0$ and traverse its fans list. We find the fans list contains 3 users: $U_1$, $U_6$, and $U_7$, and the $U_1$ is the forwarding point. Thus, the forwarded number increases 1 and we traverse the fans list of $U_1$. The fans list of $U_1$ contains 2 points, $U_2$ and $U_3$. $U_2$ is not the forwarding point, but $U_3$ is the forwarding point. So the forwarded number increases 1 and we traverse the fans list of $U_3$. The points in fans list of $U_3$ are $U_4$ and $U_5$, and both of them are not the forwarding points. So we come to $U_6$. The handling of $U_6$ is similar to the above, and in this method, we finally got all the forwarding nodes.

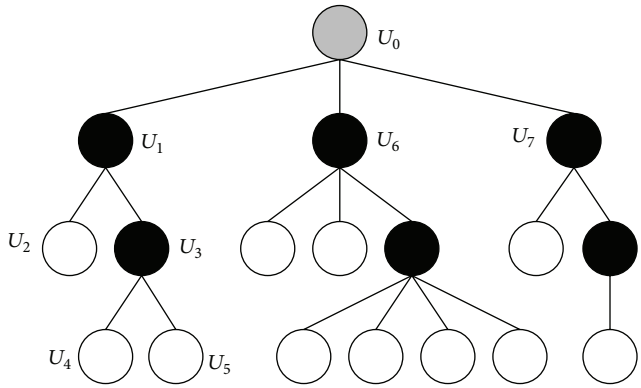


Figure 6: Prediction model tree.

Table 3: User forward behavior prediction results using ELM.

| $\Delta t$ | Accuracy | Recall | *F1*-score | Time (s) |
|---|---|---|---|---|
| 15 minutes | 0.861 | 0.865 | 0.863 | 0.0312 |
| 30 minutes | 0.878 | 0.882 | 0.88 | 0.0312 |
| 1 hour | 0.864 | 0.873 | 0.868 | 0.0312 |
| 2 hours | 0.852 | 0.865 | 0.858 | 0.0574 |
| 12 hours | 0.729 | 0.706 | 0.717 | 0.0621 |

We use Algorithm 2 to build the information dissemination prediction model. In this algorithm, we assume that each user forwards the Weibo once and the publisher will not forward the Weibo.

## 4. Experiments and Results

In this section, the predicting performance is evaluated by using ELM. In addition, we compared the results between ELM and SVM based on adding the new feature we proposed and do not use the new feature. We also test the proposed information propagation prediction model and give it performance in this section.

*4.1. Users Behavior Prediction.* According to the data we crawl from Sina Weibo, we select 133190 forward data as the forward sample. According to Section 3.1, the numbers of each ignore sample are shown in Table 2.

We use ELM to forecast forward or ignore behavior of users. The source code of ELM can be obtained from the website (ELM Source Codes: ELM Source Codes: http://www.ntu.edu.sg/home/egbhuang/). We also compare the results between ELM and SVM. The tool of *lib-SVM* is used in this paper, which can be obtained from the website (data set: http://www.csie.ntu.edu.tw/~cjlin/libsvm/). In order to evaluate the effect of forecast model, we choose the evaluation index of information retrieval, including accuracy, recall, and the value of $F1$. With 10 times of cross validation method validation algorithm, we get the user forward behavior prediction results shown in Tables 3 and 4. Table 3 shows the performance using ELM and Table 4 shows the performance using SVM.

```
Input: Weibo publisher U_0.
Output: Forwarded number C.
(1)  C = 0;
(2)  Read the fans list L(N) of U_0 (N is the number of fans, N ≥ 0)
(3)  for (i = 0; i < N; i++)
(4)      if(the user L(i) is predicted as forwarding uesr)
(5)          C = C + 1;
(6)          Information_forecast(L(i));
```

Algorithm 2: Information_forecast($U$).

Table 4: User forward behavior prediction results using SVM.

| $\Delta t$ | Accuracy | Recall | $F1$-score | Time (s) |
|---|---|---|---|---|
| 15 minutes | 0.867 | 0.875 | 0.871 | 0.0983 |
| 30 minutes | 0.869 | 0.88 | 0.874 | 0.0983 |
| 1 hour | 0.855 | 0.862 | 0.858 | 0.0983 |
| 2 hours | 0.846 | 0.861 | 0.853 | 0.1492 |
| 12 hours | 0.749 | 0.743 | 0.745 | 0.2094 |

Table 5: Predicted results without the new feature using ELM.

| $\Delta t$ | Accuracy | Recall | $F1$-score |
|---|---|---|---|
| 15 minutes | 0.854 | 0.86 | 0.857 |
| 30 minutes | 0.858 | 0.869 | 0.863 |
| 1 hour | 0.847 | 0.866 | 0.856 |
| 2 hours | 0.836 | 0.858 | 0.847 |
| 12 hours | 0.702 | 0.736 | 0.719 |

If we compare Tables 3 and 4, we can find ELM has a better performance than SVM. No matter what algorithm we used, when we take $\Delta t$ as 30 minutes, we get the best performance. Because 15 minutes is too short, some people may not have had time to release or forward Weibo. People will not spend much time in browsing Weibo once. So when the $\Delta t$ is taken as 2 hours, the performance is much lower than 30 minutes. We can also see that the performance of 12 hours is the lowest. This means people ignore Weibos not only because they do not like it, but also because they are not online. When $\Delta t$ is too long, the absent behavior plays a dominant role.

At the same time, in order to consider the time factor, we also measured the running time of ELM and SVM. And the time of ELM is far lower than the SVM.

In order to test the effectiveness of the feature we proposed, we also test the performance without the new feature. Tables 5 and 6 show the predicted results without the feature we proposed.

We can see Tables 5 and 6 also have the same conclusion with Tables 3 and 4, in which 30 minutes has the highest performance. So when we do the information propagation prediction, we choose the dataset whose $\Delta t$ is taken as 30 minutes. By comparing the Tables 3 and 5, we find using the new feature we proposed has a better performance than without the new feature. This can also be found by comparing Tables 4 and 6. In order to give a more intuitive description of this conclusion, we draw the figures to show the details. And Figures 7 and 8 show comparison charts of using ELM and using SVM.

Table 6: Predicted results without the new feature using SVM.

| $\Delta t$ | Accuracy | Recall | $F1$-score |
|---|---|---|---|
| 15 minutes | 0.849 | 0.852 | 0.850 |
| 30 minutes | 0.856 | 0.859 | 0.857 |
| 1 hour | 0.842 | 0.854 | 0.848 |
| 2 hours | 0.828 | 0.84 | 0.834 |
| 12 hours | 0.692 | 0.726 | 0.709 |

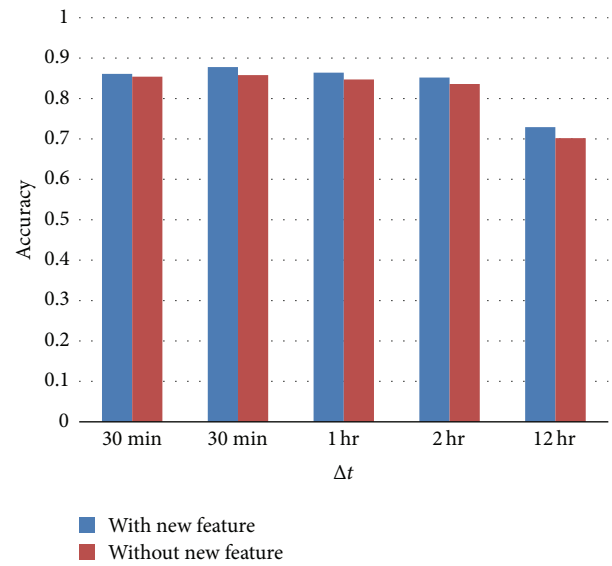


Figure 7: Comparison charts of using ELM.

As can be seen from the Figures 7 and 8, when using the dependency between the Weibos involved geographical locations and location of the user feature, the prediction results are better than without the feature.

To give a more intuitive description of the comparison, we also show the performance of ELM and SVM in a figure. Because 30 minutes has the best performance in both algorithm, we only compare the performance in this case. Figure 9 shows the comparison between ELM and SVM.

We can see in both cases that the predicted results obtained by ELM are higher than the SVM prediction results.

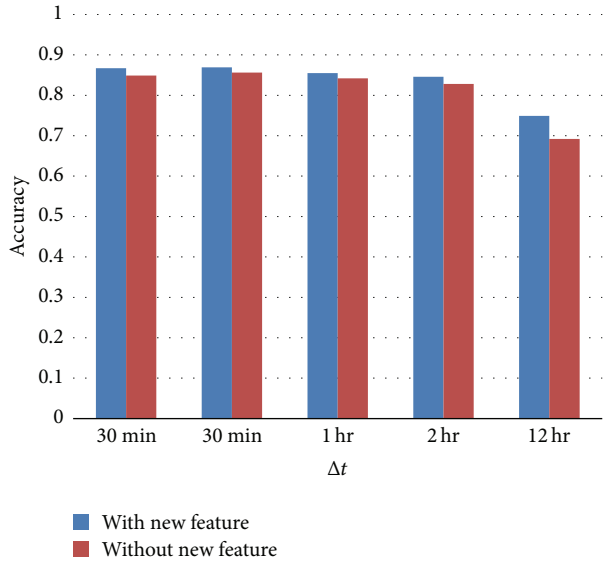


Figure 8: Comparison charts of using SVM.

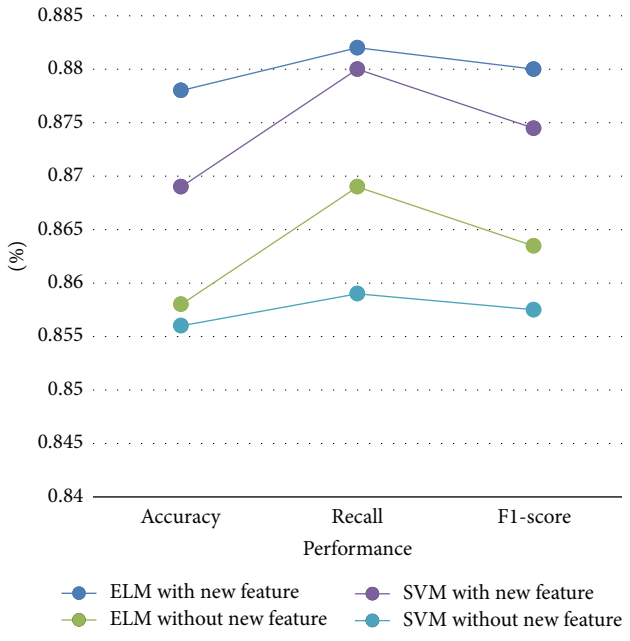


Figure 9: Comparison between ELM and SVM.

This proves that using ELM algorithm is better than using SVM algorithm. ELM algorithm has good performance. We can also see the new feature brings better performance.

*4.2. Information Propagation Prediction.* According to the algorithm in Section 3.3 of and prediction results of ELM, we predict the scale of the Information propagation. We choose 30000 original Weibos of 15375 users to verify our model. We count average user forward quantity proportion in every jump from the initial release users (jump: the shortest distance from users to the initial release user). Figure 10 shows the average of users' forward percentage in each jump.

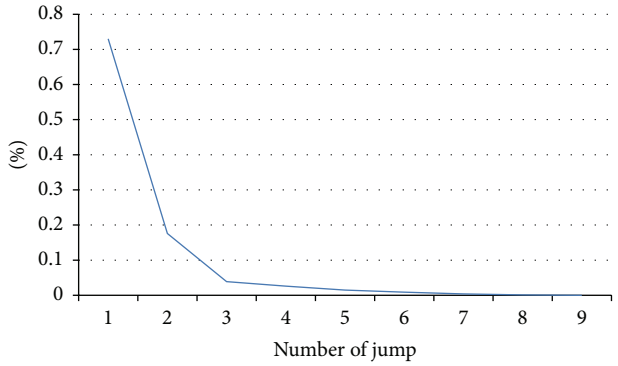


Figure 10: The average users' forward quantity proportion of each jump.

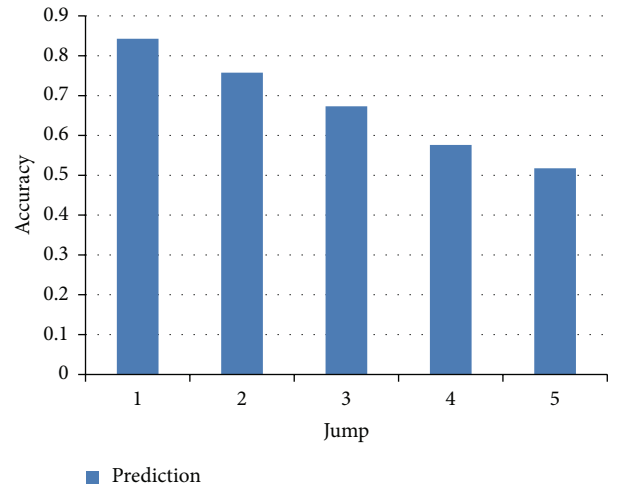


Figure 11: The accuracy in every jump.

It can be seen from Figure 10 that after 5 jump, the percent approach to 0. This proves that in our dataset all the forward behaviors happen at the first five jumps. This proves that the Weibo is a widely spread but deep low social network.

Based on this theory, our information propagation prediction stops at the fifth jump. This can avoid the excessive iteration. Figure 11 shows the accuracy we predict in every jump. The accuracy of jump 1 is the accuracy of the first forward layer of 30000 Weibos. Others are the same.

We can see in Figure 11 that the accuracy of the first jump is the highest. Accuracy reduces with the increase of the jump count. This is because when we do the prediction, the error is constantly accumulation. When the jump comes to 5, the error has been accumulated to a considerable scale. So the accuracy becomes very low.

In order to determine the scale of information dissemination, we divide the scale according to the $10^n$ order of magnitudes. If the information dissemination scale we predicted is in the same order of magnitude which is the actual information dissemination scale, we can say the prediction is right. We calculated the average predict information dissemination

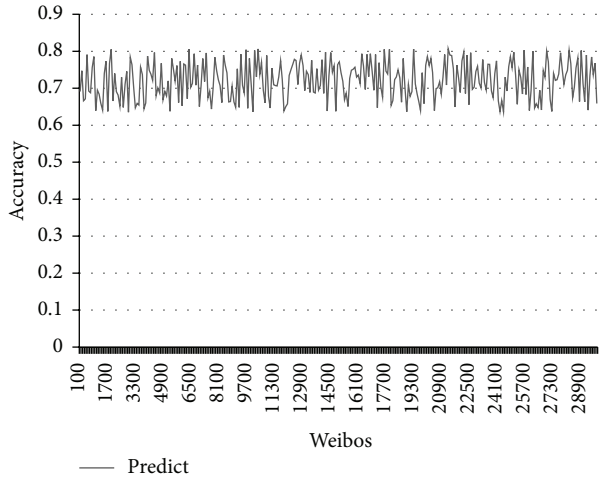


Figure 12: Average prediction accuracy of each user.

scale accuracy of 30000 Weibos. Figure 12 shows the average prediction accuracy of each Weibo.

As can be seen from Figure 12, for different Weibo from different users, our algorithm accuracy is around 70%. This is because for the Weibos whose forward deep close to 5 or more than 5 jumps, the error of our model has been accumulated to a considerable scale and brings decline in accuracy.

For the selected data, our predicting result is very stable. This proves that our algorithm is real and effective.

## 5. Conclusions

Online behavior of Weibo users and information dissemination analysis is a hot issue nowadays. In this paper, we analyzed the features of Sina Weibo user behavior and predicted the information transmission. We proposed 8 features to analyze user behavior. They are the dependency between the Weibos involved geographical locations and location of the user, the influence of user, user release activity, user forward activity, the intimacy between the users, the interest similarity between user and content, the interest similarity between users, and Weibo content importance. The feature (i.e., the dependency between the Weibos involved geographical locations and locations of the users) is the new feature we proposed. We used ELM to analyze if users will forward or ignore a weibo. Our experiment results show that the feature we proposed is very effective and ELM gets better results than SVM. We also test the different performance between the different values of $\Delta t$ in ignore dataset. We found that when $\Delta t$ is 30 minutes, the performance is the best. So we use the 30 minutes ignore dataset to build the training set. Based on that, we proposed information propagation prediction model and calculate the scale of the information propagation. The experiment results show that our model has a good performance.

The features and model we proposed in this paper can give some help to businesses and government. They can use our model to predict the scale of the information propagation before they publish it. If the scale is small, they can use our feature to adjust the information text. The model and features has very high practical value.

However, there is still something we need to improve in this paper. For example, when considering information dissemination size, we do not concern users forward their own Weibo and people may forward the Weibo many times. We will take it into consideration in the future.

## Conflict of Interests

The researchers claim no conflict of interests.

## Acknowledgments

This research was partially supported by the National Natural Science Foundation of China under Grant nos. 61332006 and 61100022, the National Basic Research Program of China under Grant no. 2011CB302200-G, and the 863 Program under Grant no. 2012AA011004.

## References

[1] G. Ou, W. Chen, B. Li, T. Wang, D. Yang, and K.-F. Wong, "CLUSM: an unsupervised model for microblog sentiment analysis incorporating link information," in *Database Systems for Advanced Applications*, vol. 8421 of *Lecture Notes in Computer Science*, pp. 481–494, Springer, 2014.

[2] J. Sun and Y. Zhu, "Microblogging personalized recommendation based on ego networks," in *Proceedings of the 12th IEEE/WIC/ACM International Conference on Web Intelligence (WI) and Intelligent Agent Technologies (IAT '13)*, pp. 165–170, Atlanta, Ga, USA, November 2013.

[3] G. Song, Z. Li, and H. Tu, "Forward or ignore: user behavior analysis and prediction on microblogging," in *Advanced Research in Applied Artificial Intelligence*, vol. 7345 of *Lecture Notes in Computer Science*, pp. 231–241, Springer, Berlin, Germany, 2012.

[4] J.-X. Cao, J.-L. Wu, W. Shi, B. Liu, X. Zheng, and J.-Z. Luo, "Sina microblog information diffusion analysis and prediction," *Chinese Journal of Computers*, vol. 37, no. 4, pp. 779–790, 2014.

[5] A. Mogadala and V. Varma, "Twitter user behavior understanding with mood transition prediction," in *Proceedings of the ACM Workshop on Data-Driven User Behavioral Modeling and Mining from Social Media (DUBMMSM '12)*, pp. 31–34, October 2012.

[6] J. Leskovec, M. McGlohon, C. Faloutsos, N. Glance, and M. Hurst, "Patterns of cascading behavior in large blog graphs," in *Proceedings of the 7th SIAM International Conference on Data Mining*, pp. 551–556, Philadelphia, PA, USA, April 2007.

[7] J. Lingad, S. Karimi, and J. Yin, "Location extraction from disaster-related microblogs," in *Proceedings of the 22nd International Conference on World Wide Web Companion (WWW '13)*, pp. 1017–1020, 2013.

[8] S. Hosseini, S. Unankard, X. Zhou, and S. Sadiq, "Location oriented phrase detection in microblogs," in *Database Systems for Advanced Applications: Proceedings of the 19th International Conference, DASFAA 2014, Bali, Indonesia, April 21–24, 2014, Part I*, vol. 8421 of *Lecture Notes in Computer Science*, pp. 495–509, Springer International Publishing, Cham, Switzerland, 2014.

[9] J. Marques and C. Serrão, "Improving user content privacy on social networks using rights management systems," *Annals of Telecommunications*, vol. 69, no. 1-2, pp. 37–45, 2014.

[10] H. T. Quang, H. V. H. Tien, H. N. Le, T. H. Trung, and P. Do, "Finding the cluster of actors in social network based on the topic of messages," in *Intelligent Information and Database Systems*, vol. 8397 of *Lecture Notes in Computer Science*, pp. 183–190, Springer, New York, NY, USA, 2014.

[11] C.-Y. Tseng and M.-S. Chen, "Incremental SVM model for spam detection on dynamic email social networks," in *Proceedings of the IEEE International Conference on Computational Science and Engineering (CSE '09)*, pp. 128–135, Vancouver, Canada, August 2009.

[12] T. R. Zaman, R. Herbrich, J. van Gael, and D. Stern, "Predicting information spreading in twitter," in *Proceedings of the Workshop on Computational Social Science and the Wisdom of Crowds*, NIPS 2010, 2010.

[13] F. Wu and B. A. Huberman, "Novelty and collective attention," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 104, no. 45, pp. 17599–17601, 2007.

[14] W. Webberley, S. Allen, and R. Whitaker, "Retweeting: a study of message-forwarding in Twitter," in *Proceedings of the 1st IEEE NSS Workshop on Mobile and Online Social Networks (MOSN '11)*, pp. 13–18, Milan, Italy, September 2011.

[15] R. Bandari, S. Asur, and B. A. Huberman, "The pulse of news in social media: forecasting popularity," in *Proceedings of the International Conference on Weblogs and Social Media (ICWSM '12)*, pp. 26–33, Dublin, Ireland, June 2012.

[16] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: a new learning scheme of feedforward neural networks," in *Proceedings of the IEEE International Joint Conference on Neural Networks (IJCNN '04)*, pp. 985–990, Budapest, Hungary, July 2004.

[17] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: theory and applications," *Neurocomputing*, vol. 70, no. 1–3, pp. 489–501, 2006.

[18] G.-B. Huang, D. H. Wang, and Y. Lan, "Extreme learning machines: a survey," *International Journal of Machine Learning and Cybernetics*, vol. 2, no. 2, pp. 107–122, 2011.

[19] G.-B. Huang, H. Zhou, X. Ding, and R. Zhang, "Extreme learning machine for regression and multiclass classification," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 42, no. 2, pp. 513–529, 2012.

[20] D. Ren, X. Zhang, Z. Wang, J. Li, and X. Yuan, "Weibo events: a crowd sourcing weibo visual analytic system," in *Proceedings of the 7th IEEE Pacific Visualization Symposium (PacificVis '14)*, pp. 330–334, Yokohama, Japan, March 2014.

[21] S. Petrović, M. Osborne, and V. Lavrenko, "RT to win! predicting message propagation in twitter," in *Proceedings of the 5th International Conference on Weblogs and Social Media (ICWSM '11)*, pp. 586–589, Barcelona, Spain, July 2011.

[22] L. Page, S. Brin, R. Motwani, and T. Winograd, "The PageRank citation ranking: bringing order to the web," Tech. Rep. SIDL-WP, Stanford University, 1999.

[23] X.-M. Lin and W. Wang, "Set and string similarity queries: a survey," *Chinese Journal of Computer*, vol. 34, no. 10, pp. 1853–1862, 2011.

[24] C. Shi, C. Xu, and X. Yang, "Study of TFIDF algorithm," *Journal of Computer Applications*, vol. 6, no. 29, pp. 167–170, 2009.