Detecting Spammers from Hot Events on Microblog Platforms: An Experimental Study

Jialing Liang[1], Peiquan Jin[1*], Lin Mu[1], Jie Zhao[2*]
[1]School of Computer Science and Technology, University of Science and Technology of China, Hefei, China
[2]School of Business, Anhui University, Hefei, China
jpq@ustc.edu.cn

***Abstract*—With the development of Web 2.0, social media such as Sina Weibo, Douban, and Zhihu have become an important platform for the dissemination and fermentation of hot events. At the same time, many spammers are hidden in the network, and they are driven by the interests to participate in the process of event dissemination, disseminating information with a propensity, and guiding public opinion through speculation, malicious comments, malicious attacks, etc. It interferes with the network order and the decision making based on social networks, and even affects social stability. Therefore, it is important for government and enterprises to accurately detect spammers from hot events on microblog platforms and further make sure whether a hot event is natural developing or driven by spammers. In this paper, we focus on the hot event list on Sina Weibo and collect relevant microblogs and involved users of each hot event. Then, we employ typical machine learning methods to conduct an experimental study on detecting spammers. Specially, we develop a new set of features based on three aspects, including user profile, user behavior, and user relationships, to reflect various factors affecting the detection of spammers. Finally, we conduct experiments on a real data set from the Sina Weibo, and compare three machine learning models including the Naive Bayes, the J48 Decision Tree, and the Logistic Regression model, concerning various metrics like precision, recall, F-measure and AUC. The results show that the Logistic Regression model achieves the best average F-measure in detecting both spammers and non-spammers.**

Keywords-microblog; spammer detection; user feature; classification model

I. INTRODUCTION

As online users migrate to mobile terminals and social media, social platforms such as Weibo that are instant and convenient have become important channels for netizens to interact, share, and disseminate with others. Weibo users can share a daily life by posting a short text on the web and mobile terminals and can browse the information posted by other users to get attention to current events and hot spots or participate in the discussion of popular topics and the spread of events through methods such as reposting comments. in. According to the CNNIC Report of China [1] released by the China Internet Network Information Center in February 2019, as of December 2018, the number of microblog users in China was 350 million, accounting for the total number of Internet users. The proportion reached 42.3%. In typical social applications, compared to the privacy of the WeChat circle of friends and QQ space, Weibo has suddenly become a mainstream online media with many significant features such as large user scale, strong sociality, fast propagation speed, and fast response speed. For example, on November 17, 2018, the People's Daily published the Weibo topic "China is not a little bit". It has been reposted 1.259 million times in just half a day, received 118,000 comments and 943,000 likes. The topic of reading reached 8.94 billion.

The existence of Weibo provides a fast platform for the propagation of hot events [2-4]. But at the same time, due to the emergence and promotion of the online water army, Weibo has also become a target platform for spreading rumors and hype. Spammers are those that are driven by commercial interests to achieve improper purposes such as influencing public opinion and disrupting the network environment, thereby manipulating software robots or spam accounts and producing and disseminating false information and spam on the Internet. Generally, spammers may manipulate spam accounts to speculate bland blog posts or topics as hot events. The purpose is to gain fame and attention, fight against hostile forces, stir up public sentiment, or guide public opinion, Spammers can mislead Internet users' correct judgment of the situation of events, and even maliciously attack the government and affect social stability Due to its large scale, the high degree of coverage, and wide target range, the network spam makes it difficult to identify spam accounts from a great number of users solely by manual means.

Based on the above analysis, this article focuses on the detection of spammers during the spread of Sina Weibo hot events. Particularly, we use machine learning methods to identify spam accounts from users who participate in the process of promoting an event to become a hot spot, giving evidence of whether the event propagation process is naturally fermented or promoted by the spam, and finally help decision-makers to guide and control public opinion [2].

Briefly, we make the following contributions in this paper:

(1) We design a crawler platform to collect detailed personal information of an event-related Weibo user, making the data set more realistic. We also set up a Weibo user manual labeling platform and design the labeling process so that each Weibo user can judge the labeled results, reducing the errors caused by manual labeling. In addition, the web-based platform makes labeling work much convenient and reliable.

(2) A Weibo spammers detection method combining user attribute characteristics, user behavior characteristics, and user relationship characteristics is proposed. Compared with the existing methods, the method proposed in this paper

* Corresponding author
DOI reference number: 0.18293/SEKE2020-080

comprehensively considers three user characteristics, which is more suitable for the identification of spammers.

(3) Based on the defined feature set, we conduct experiments on a real dataset and compare the performance of three types of classification models, including Naive Bayes, J48 decision tree, and logistic regression model. The results show that the logistic regression model has the best detection effect.

The remainder of the paper is structured as follows. Section II provides a brief literature review on recent research progress. Section III describes the data crawling and cleaning process. Section IV presents feature selection. Section V reports the experimental results, and finally, we conclude the entire paper in Section VI.

II. Related Work

Spammers first appeared in the e-mail field, and then quickly spread to the e-commerce and social fields. Existing detection methods of network spammers are mainly divided into detection based on content features, user features, environment features, and comprehensive features [5].

In the early network environment, the spam was mainly used to create many spam emails and false comments on e-commerce platforms. The content generated by the online spam included obvious characteristics, such as commercial advertisements, spam, duplicate comments, etc. Most of the network spammers' recognition is based on the detection of content features, involving text orientation analysis [6], sentiment analysis [7], and other methods in natural language processing. The filtering and detection of spam have been researched for a long time: the literature [8] analyzes the existing detection and evaluation work in the two fields of electronic spam and image spam; literature [9] has seven differences A comparative study of the version of the Naive Bayes classifier and the linear support vector machine for automatic filtering of e-mail spam was conducted. For fake reviewers in e-commerce platforms and forums, usually by analyzing the text 's propensity analysis to identify fake reviews that deviate from unspammed user reviews [10]; some researchers also look for different rules or groups of rules. Used to detect abnormal comment user behavior [11].

With the rise and development of social platforms such as Twitter, Facebook, and Sina Weibo, and the increase in the number of users on the social network, coupled with the enhancement of user identification, the Internet has continued to improve its concealment and deception strategies. For normal users, its published content no longer has obvious spam features. Therefore, the detection and recognition of the network spam have also gradually shifted from content-based features to user-based features. Benevenuto et al. [12] used tweets related to the three hot topics to manually construct a labeled dataset, determine 39 attribute features related to the content of the tweet and 23 attribute features related to the user, and then use the SVM method is used to classify and finally the analysis of experimental results is performed. Murmann et al. [13] used neighbor nodes with interactive relationships to detect the trust relationship between users in Twitter, and obtained a new relationship feature set, and used this feature set to rank the suspiciousness of all users, with the highest suspiciousness among them. That is judged as the network spam. Wang et al. [14] created a directed social graph to show the relationship between followers and fans. Based on the tweet content features and user relationship graph features, the Bayesian classifier was used for spam detection, achieving an accuracy of 89%. rate. Yang et al. [15] deeply analyzed the concealment and deception strategies of the Twitter network spam and proposed a method to detect the network spam in Twitter based on the characteristics of neighbor nodes. Han Cao et al. [16] constructed a recognition network by taking the user's attribute characteristics as the input variables of the learning model, the user's behavior characteristics as the observation variables, and the probability that the user is a spam force is the hidden variable between the input and the observed variables The spam's probability map model is used to calculate the probability that the user is spam. Bhat et al. [17] found that similar to ordinary users, the network sailors in the social field can also form a certain size network sailor community. To this end, they extracted user interaction diagrams from the behavior logs of Weibo users, found overlapping community maps formed therein, and after manually marking some of the network spam nodes, they calculated each node to be identified. Communicate with the community of marked nodes to classify unknown nodes. In addition, Azad, et al. [18] presented a rapid detection method for spammers through collaborative information sharing across multiple service providers, which showed that fusing muttiple information provided by various providers was helpful for spammer detection.

Compared with the existing work, this paper designs a new crawler algorithm, which takes the keywords of the hot event as seeds to crawl the event-related microblogs and the detailed personal information of the microblog users who participated in this hot event. We also construct a manual labeling platform to tag the dataset. In addition, we propose a new feature set based on user attribute characteristics, user behavior characteristics, and user relationship characteristics. Compared with the existing methods, the method proposed in this paper comprehensively considers three user characteristics, which are more suitable for the identification of spammers.

III. Data Crawling and Labelling

A. Data Crawling

The data crawling part uses Python's crawler framework and configures the Google Chrome driver to simulate login to obtain cookie data. We use popular event keywords and link to the old search interface of Sina Weibo to form a seed URL, and crawl the Weibo details returned by the search page, including Weibo content, likes, retweets, comments, release time and personal information. The crawled data is stored in MongoDB, which maintains Weibo information tables and personal information tables.

B. Data Cleaning

The data cleaning of the original data crawled by the crawler is mainly divided into two steps. The first step is to filter the Weibo or missing important information generated by the dynamic webpage or Weibo anti-crawling caused by the 302 transfer when crawling data. The second step is the manual labeling phase. When the user information is abnormally absent, for example, the number of followers, followers, and tweets of a user is not 0, but the list of followers, followers, and tweets is empty, we cannot judge whether the user is a spam user based

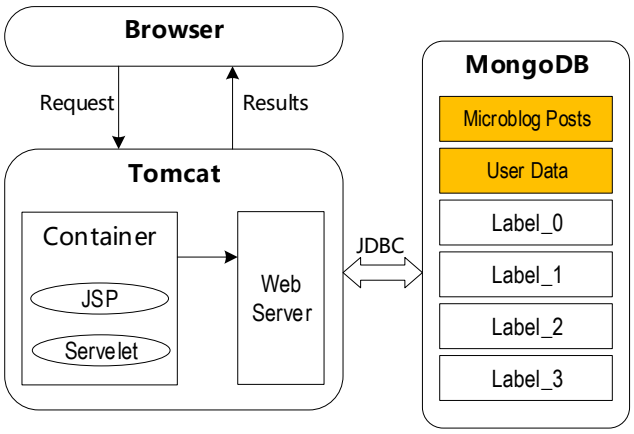

Figure1. Architecture of the labeling platform.

on the existing information. As a result, such users will be removed from the database at this time.

C. The Labeling Platform

The manual labelling platform is a tool we developed and deployed on a Tomcat server. The architecture is shown in Fig. 1.

There are four label tables denoted as Lable_0, Label_1, Lable_2, and Label_3 in Fig. 1. The Label_0 stores the initial unlabeled user ID. The Label_1 stores the user ID and label labeled by one tagger. The Label_2 stores the user ID and label labeled by two taggers, and the Label_3 stores the user ID and labeled mark.

The process of the labeling platform is as follows:

(1) Copy the IDs of all users from the personal information table to Label_0, and set the flag field to -1 to indicate no flag;

(2) The client sends a request, and the web container will perform the JSP conversion and the compiled file. When the Label_0 table is not empty, randomly obtain a user ID from Label_0, and obtain the detailed information of the ID from the personal information table. The result is returned to the browser. If Label_0 is empty, the ID is obtained from Label_1, and so on, until the Label_2 table is also empty, indicating that the mark has been completely completed;

(3) Submit the tag determined by the tagger to the web server and forward it to the servlet container. At this time, delete the ID from $Label_i$, change the value of the tag field, and add the user ID and the tag field to $Label_i + 1$.

(4) The browser refreshes the current page after receiving a response, that is, continues to step (2) until all users have completed the labeling by three users.

IV. Features Selection

To effectively identify the spam users in the user group, in this paper we design the following features (see Table 1).

A. User-Profile Features

(1) Num_Follows

Unspammed users generally only pay attention to the people they are interested in, so the number of followers will be in a relatively reasonable range. To achieve the effect of publicity and hype, spammers often follow a lot of bloggers. Users will have a higher number of followers than non-spammers.

TABLE I. Features of Microblog Users

No.	Type	Feature
1	User-Profile Features	Num_Follows
2	User-Profile Features	Num_Fans
3	User-Profile Features	Num_Tweets
4	User-Profile Features	FAuthentication
5	User-Profile Features	FBriefIntroduction
6	User-Profile Features	FVIP
7	User-Behavior Features	Original_Ratio
8	User-Behavior Features	URLs_Ratio
9	User-Behavior Features	Mentions_Ratio
10	User-Behavior Features	Topics_Ratio
11	User-Behavior Features	Self-Similarity
12	User-Relationship Features	Fans_Follows_Ratio
13	User-Relationship Features	Aggregation_Coef

(2) Num_Fans

Unspammed users will have a circle of friends on the Weibo platform, so there is a certain percentage of followers, and spammers are often fans of other people, but they rarely attract the interest and attention of others. Compared with unspammed users, the number of fans of the spam is very small.

(3) Num_Tweets

Unspammed users use Weibo normally. There will be a certain percentage of Weibo users, who are either new users only publishing or forwarding specific tweets or be active in the comments to promote hype and public opinions. On the other side, their own posts are few. Thus, non-spammer users generally post more than spammers.

(4) FAuthentication

On microblogging platforms, an authenticated account will generally be more credible and authoritative than an unauthenticated account. Therefore, authenticated users are more likely to be unspammed users, while unauthenticated users are more likely to be spammers.

(5) FBriefIntroduction

Spammers generally have relatively low completeness of the information. Few Spam users fill out the personal profile field. Therefore, Spammers are more likely to have no profile, and unspammed users are more likely to have a profile.

(6) FVIP

Generally speaking, spammers do not need to register a VIP because it is costly. However, many normal users will choose to

pay for VIPs to obtain more functions and benefits when using microblogging services. To this end, users who are VIPs are more inclined to be unspammed users, and users who have not registered as VIPs are more likely to be spammers.

B. User-Behavior Features

(1) Original_Ratio

$$Original\_Ratio = \frac{Original\_Tweets}{Num\_Tweets}$$

Spammers are usually controlled by machines or robots, so most spammers are more likely to repost and comment on a certain microblog, and rarely publish original microblogs. On the contrary, normal users will share their daily life around them and will publish a certain percentage of original tweets. Thus, the original ratio of tweets posted by spammers can be generally lower than that of normal users.

(2) URLs_Ratio

$$URLs\_Ratio = \frac{Num\_URLs}{Num\_Tweets}$$

Unspammed users are limited to 140 characters of tweet text for propaganda and hype. The microblogs posted or reposted may contain more URLs of web links than unspammed users, thus inducing users to click on the links to browse the page they want to display. Therefore, the utilization rate of URLs for spam users is generally higher than that of non-spam users.

(3) Mentions_Ratio

$$Mentions\_Ratio = \frac{Num\_Mentions}{Num\_Tweets}$$

The @ method is used to remind users who are @ to view the Weibo content in time. After being logged in by @ users, they can see the reminder information of the Weibo. Spammers will attract @ users' attention through @some unrelated users, to achieve rapid diffusion. Therefore, the @usage rate of spam users may be higher than that of unspammed users.

(4) Topics_Ratio

$$Topics\_Ratio = \frac{Num\_Topics}{2 \times Num\_Tweets}$$

When users participate in the discussion of a hot topic, a # sign is often included outside the topic. Spammers will use the hashtag # more to achieve the hype topic and promote the topic to become a popular purpose. Therefore, the # usage rate of spam users may be higher than that of unspammed users. Since # always appears in pairs, and Num_Topics only represents the number of occurrences of # in tweets, the denominator in the definition needs to be multiplied by 2.

(5) Self-Similarity

The self-similarity among historical tweets refers to the proportion of similar tweets in the total number of posts published by users. To achieve the purpose of publicity and marketing, spammers often use content templates to generate many similar microblogs. Therefore, the historical microblog self-similarity of spammers is generally higher than that of non-spammers.

To calculate the self-similarity, we use a hierarchical clustering method based on the cosine similarity to cluster the historical microblogs of a user to form clusters $S=(C\_1, C\_2, ..., C\_k)$. Here, k is the number of clustered classes. C_J is the jth class that contains N tweets, which can be defined as $C\_J=(T\_J1, T\_J2, ..., T\_JN)$. The N tweets are regarded as similar tweets. Then, we define the self-similarity of tweets as follows [15].

$$Similarity = \frac{\sum_{J=1}^{J=k} G\left(C_J\right)}{Num\_Tweets}, G(C_J) = \begin{cases} N & , N \geq 2 \\ 0 & , \text{otherwise} \end{cases}$$

C. User-Relationship Features

(1) Fans_Follows_Ratio

$$Fans\_Follows\_Ratio = \frac{Num\_Followees}{Num\_Followers}$$

Unspammed users have a normal social circle of friends with a similar number of followees or followers, or large V users have a great number of followees, and the ratio of followees to followers is large. However, spammers tend to follow many users but only a small number of followees; therefore, the follower ratio of spam users will be lower, and the follower ratio of unspammed users will be higher.

(2) $Aggregation\_Coef_i$

$$Aggregation\_Coef_i = \frac{\sum_{j,k=1}^{N} a_{ij}\, a_{jk} a_{ki}}{k_i(k_i - 1)}$$

We construct an undirected graph G = (V, E) using the followee and follower list of all users crawled. The adjacency matrix of graph G is expressed as $A = (a_{ij})_{N \times N}$, and k_i is the degree of node i, $\frac{1}{2}\sum_{j,k=1}^{N} a_{ij}\, a_{jk} a_{ki}$ represents the number of neighbor pairs formed between node i and k_i neighbor nodes. $ggregation_{Coef_i}$ calculates the clustering coefficient of user i. In general, the clustering coefficient is used to evaluate the probability that a user's friends are also friends with each other: for unspammed users, they are closer to their neighbors, that is, the network of friends is closer, yielding a large coefficient. However, as more neighbors of spammers are independent points, their clustering coefficient will be relatively small.

V. PERFORMANCE EVALUATION

A. Settings

Dataset. The data set in this study was collected from the users who participated in comments and reposts under the popular tweets returned by the search keyword "Huawei sued the US government" in the old Sina Weibo search interface. There are 341 popular microblogs related to words and topics. After filtering out users who are restricted by Weibo anti-crawling restrictions and crawling incomplete information, 8149 users have been collected. After manual labeling, 312 of them are non-spammer users. There are 7,837 spammers. To avoid the

category imbalance caused by the large difference between the positive and negative examples, the data of a total of 800 users, including 312 spam users and 488 unspammed users are used for 10-fold cross-validation in the experiment.

Metrics. In the experiments, we use precision, recall, F-measure, and the AUC under the ROC curve as the evaluation indicators for spammer detection. Let TP be the number of spam users correctly classified by the classifier, FP be normal users incorrectly classified as spam users, and FN be spam users incorrectly classified as normal users. The accuracy, recall and F1 values are defined as follows:

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$\mathrm{F1} = \frac{2 \times P \times R}{P + R}$$

B. Classification Models

We select three classification models and compare their performance on detecting spammers.

(1) *Naïve Bayes.* Let *D* be the training set, A be the attribute set of users, we represent each user by an *n*-dimensional vector $X = (x_1, x_2, \ldots, x_n)$, and the results are labeled into *m*=2 classes $C = (C_1, C_2, \ldots, C_m)$.

(2) *J48 Decision Tree*. The J48 decision tree algorithm is a top-down, recursive divide-and-conquer strategy: selecting a certain attribute to place at the root node, generating a branch for each possible attribute value, dividing the instance into multiple subsets, each subset corresponding to a root Node branches, then repeat this process recursively on each branch. When all instances have the same classification, the algorithm stops. Let *D* be the training set, A be the attribute set of users, we represent each user by an *n*-dimensional vector $X = (x_1, x_2, \ldots, x_n)$, and the results are labeled into *m*=2 classes $C = (C_1, C_2, \ldots, C_m)$.

(3) *Logistic Regression*. Let *D* be the training set, A be the attribute set of users, and *m* be number of samples, we represent each user by an *n*-dimensional vector $X = (x_1, x_2, \ldots, x_n)$, and the results are labeled into $C = (C_1, C_2)$.

C. Results

We use Java-based machine learning library Weka for the classification experiments. Tables II and II list the confusion matrix and classification results of the dataset under the Naive Bayes algorithm. Tables IV and V list the confusion matrix and classification results of the dataset under the J48 decision tree algorithm. Tables VI and VII list the confusion matrix and classification results of the data set under the logistic regression algorithm.

We can see that the decision tree achieves the highest precision for detecting spammers, but its recall is not the highest among all the three models. The naïve Bayes model achieves the best recall, but its precision is the lowest among all compared models, which leads to the lowest F-measure in the experiments.

We also list the performance of non-spammer detection in the tables. Generally, the recognition of non-spammers is as important as the detection of spammers. Thus, we calculate the

TABLE II. CONFUSION MATRIX OF NAIVE BAYES

Type	Detected as spammers	Detected as non-spammers
Spammer	96.67%	3.33%
Non-spammer	41.04%	58.96%

TABLE III. CLASSIFICATION RESULTS OF NAÏVE BAYES

Type	Hit Ratio	Error Rate	Precision	Recall	F-Measure	AUC
Spammer	0.967	0.410	0.613	0.967	0.750	0.938
Non-Spammers	0.590	0.033	0.963	0.590	0.731	0.937
Avg.	0.741	0.185	0.822	0.741	**0.739**	0.938

TABLE IV. CONFUSION MATRIX OF DECISION TREE

Type	Detected as spammers	Detected as non-spammers
Spammer	90.00%	10.00%
Non-spammer	9.70%	92.30%

TABLE V. CLASSIFICATION RESULTS OF DECISION TREE

Type	Hit Ratio	Error Rate	Precision	Recall	F-Measure	AUC
Spammer	0.900	0.097	0.862	0.900	0.880	0.925
Non-Spammers	0.903	0.100	0.931	0.903	0.917	0.925
Avg.	0.902	0.099	0.903	0.902	**0.902**	0.925

TABLE VI. CONFUSION MATRIX OF LOGISTIC REGRESSION

Type	Detected as spammers	Detected as non-spammers
Spammer	93.33%	6.67%
Non-spammer	11.19%	88.81%

TABLE VII. CLASSIFICATION RESULTS OF LOGISTIC REGRESSION

Type	Hit Ratio	Error Rate	Precision	Recall	F-Measure	AUC
Spammer	0.933	0.112	0.848	0.933	0.889	0.956
Non-Spammers	0.888	0.067	0.952	0.888	0.919	0.956
Avg.	0.906	0.085	0.910	0.906	**0.907**	0.956

average precision, recall, and F-measure of both spammers and non-spammers detection for all three models. The average value is denoted as the "avg." column in all tables. We can see that in terms of the average F-measure, which can be regarded as a balanced metric of precision and recall, the naïve Bayes performs worst and the Logistic Regression model performs best. The decision tree model gets comparable performance with the Logistic Regressions, indicating that it can also be considered in the detection of spammers.

VI. Conclusions and Future Work

Spammers have severely disrupted network order and decision analysis based on social networks. For hot events on the Weibo platform, judging whether the event is a natural fermentation or a spam promotion is of great significance for the government and enterprises to correctly evaluate the event situation. This article uses Sina Weibo's popular event keywords as a starting point, crawls the details of the Weibo returned under this keyword, and crawls the personal details of the commenting and forwarding users under Weibo. Furthermore, we designed and built an artificial labeling platform for Weibo users of the spammers / unspammed army. By displaying the user's personal information, the tagger will make judgments based on personal information to reduce the error of manual judgment. Based on this, a spam identification method combining user attribute characteristics, behavior characteristics, and relationship characteristics is proposed. We combine the results of artificial labeling to build the input set of the classification model and use the naive Bayes, J48 decision tree, and logistic regression models in the classification model to experimentally verify the real data set. The experimental results show that the logistic regression algorithm has the best classification effect on microblog user spam detection.

In future research, we will investigate a few topics. First, we will carry out the detection and analysis of the spam to obtain the proportion of spammers involved in a hot event. Second, we will study the evolution of public sentiment and topics related to spam [19-21], which is an important indicator to reveal the dynamic feature of spam on social networks. Third, we will crawl real-time hotspot events on social networks and construct a prototype that can monitor real-time spammers on microblogging platforms.

Acknowledgments

This work is supported by the National Science Foundation of China (no. 61672479 and 71273010) and the National Statistical Science Research Project (no. 2019LY66). Peiquan Jin and Jie Zhao are the joint corresponding authors of this paper.

References

[1] The CNNIC Report of China. http://www.cnnic.net.cn/hlwfzyj/hlwxzbg/hlwtjbg/201902/P020190318523029756345.pdf. Accessed on 1 March, 2020

[2] J. Zhao, X. Wang, P. Jin, Feature selection for event discovery in social media: A comparative study . *Computers in Human Behaviour*, 2015, 51(B): 903-909

[3] L. Zheng, P. Jin, J. Zhao, L. Yue: A fine-grained approach for extracting events on microblogs. *Proceedings of the 25th International Conference on Database and Expert Systems Applications (DEXA)*, 2014: 275-283

[4] P. Jin, L. Mu, L. Zheng, J. Zhao, L. Yue: News feature extraction for events on social network platforms. *Proceedings of the 26th International World Wide Web Conference (WWW)*, 2017: 69-78

[5] S. Rathore, et al. SpamSpotter: An efficient spammer detection framework based on intelligent decision support system on Facebook. *Applied Soft Computing*. 2018, 67: 920-932

[6] H. Do, et al. Deep learning for aspect-based sentiment analysis: A comparative review. *Expert Systems with Applications*. 2019, 118: 272-299

[7] D. Zimbra, et al. The state-of-the-art in Twitter sentiment analysis: A review and benchmark evaluation. *ACM Transactions on Management Information Systems.* 2018, 9(2): 5:1-5:29

[8] P. Hayati, et al. Evaluation of spam detection and prevention frameworks for email and image spam: a state of art, *Proceedings of the 10th International Conference on Information Integration and Web-based Applications & Services (iiWAS)*, 2008: 520-527.

[9] T. Almeida, et al. Content-based spam filtering, *Proceedings of The 2010 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2010: 1-7.

[10] F. Li, et al. Learning to identify review spam, *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence (IJCAI)*. 2011: 2488-2493.

[11] N. Jindal, et al. Finding unusual review patterns using unexpected rules, *Proceedings of the 19th ACM International Conference on Information and Knowledge Management (CIKM)*, 2010: 1549-1552.

[12] F. Benevenuto, et al. Detecting spammers on twitter, *Proceedings of the Seventh annual Collaboration, Electronic messaging, AntiAbuse and Spam Conference (CEAS)*, Vol. 6, 2010: 12.

[13] A. Murmann. Enhancing spammer detection in online social networks with trust-based metrics. San Jose State University, 2009.

[14] A. Wang. Don't follow me: Spam detection in twitter, *Proceedings of the 2010 International Conference on Security and Cryptography (SECRYPT)*, 2010: 1-10.

[15] C. Yang, et al. Die free or live hard? Empirical evaluation and new design for fighting evolving twitter spammers, *Proceedings of the International Workshop on Recent Advances in Intrusion Detection*. 2011: 318-337.

[16] J. Cao, et al. Collusion-aware detection of review spammers in location based social networks. *World Wide Web*, 2019, 22(6): 2921-2951

[17] S. Bhat, et al. Community-based features for identifying spammers in online social networks, *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*. 2013: 100-107.

[18] M. Azad, et al. Rapid detection of spammers through collaborative information sharing across multiple service providers. *Future Generation of Computer Systems*. 2019, 95: 841-854

[19] J. Liang, L. Mu, P. Jin. MGP: Extracting multi-granular phases for evolutional events on social network platforms. *Proceedings of the 14th International Conference on Semantics, Knowledge and Grids (SKG)*, 2018: 269-272

[20] L. Mu, P. Jin, L. Zheng, E. Chen, L. Yue. Lifecycle-based event detection from microblogs. *Proceedings of the 27th International World Wide Web Conference (WWW)*, 2018: 283-290

[21] L. Mu, P. Jin, L. Zheng, E. Chen. EventSys: Tracking event evolution on microblogging platforms. *Proceedings of the 23rd International Conference on Database Systems for Advanced Applications (DASFAA)*, 2018: 797-801