Filtering Spam in Weibo Using Ensemble Imbalanced Classification and Knowledge Expansion

Zhipeng Jin[1] Qiudan Li[1] Daniel Zeng[1,2] Lei Wang[1,3]
[1] The State Key Laboratory of Management and Control for Complex Systems,
Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China
[2] Department of Management Information Systems, University of Arizona, Tucson, AZ 85721, USA
[3]College of Management and Economics, Tianjin University, Tianjin 300072, China
{jinzhipeng2013, qiudan.li, dajun.zeng, l.wang}@ia.ac.cn

***Abstract*—Weibo has become an important information sharing platform in our daily life in China. Many applications utilize Weibo data to analyze hot topic and opinion evolution patterns to gain insights into user behavior. However, various spam messages degrade the performance of these applications and thus are essential to be filtered. In this paper, we propose a unified spam detection approach, which utilizes external knowledge sources to expand keywords features and applies an ensemble under-sampling based strategy to handle the class-imbalance problem. The experimental results show the effectiveness and robustness of our approach in Weibo data.**

Keywords—spam detection; external knowledge expansion; ensemble learning; class-imbalance learning

I. INTRODUCTION

Nowadays, Weibo, which is similar to Twitter, has become an important information sharing platform in China. By analyzing hot topic and opinion evolution patterns in Weibo, we can gain insights into user behavior. However, the ubiquitous spam in Weibo make the above analysis process increasingly more difficult. In order to better understand user interactions, it would be essential to filter spam accurately and effectively. Challenges of spam detection in Weibo arise from the following aspects: firstly, characteristics of free written style and short length of tweets remain a large obstacle to explore the features of distinguishing spam from non-spam. Secondly, gathering and labeling of spam are relatively costly and difficult. In Weibo data, the ratio of spam is approximately 10%，thus, how to take full advantage of the information contained in non-spam is an important way to enhance the performance of spam filtering.

Previous works have been mainly focused on spammer detection. These works generally utilize various features like content-based features [1], behavior-based features [2] and graph-based features [3] to detect the spammers. Spam detection shares some similar characteristic such as content-based features and classification-based learning strategy with spammer detection.

We propose a three step approach: 1) extracting distinguished features from Weibo; 2) developing a computational approach to expand keywords features automatically by utilizing external knowledge source; 3) employing an ensemble under-sampling based strategy to handle the class-imbalance problems and boost the performance of spam filtering. We have empirically evaluated the proposed algorithm on a real-world Weibo data. The results show that our algorithm could filter spam accurately.

II. SPAM DETECTION SYSTEM

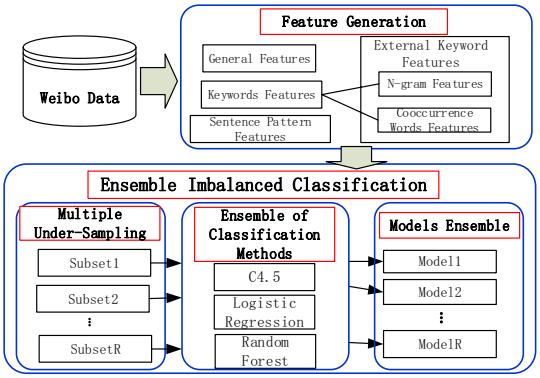

Fig. 1. The framework of spam detection in Weibo

Fig.1 depicts the proposed methodology of our spam detection system. At the feature generation stage, we extract distinguished features and expand keywords features by utilizing external knowledge source. At the stage of classification, we adopt an ensemble learning method to improve the classification performance and an under-sampling based strategy to overcome the imbalance of data distribution.

A. Features Generation

We divide these features into three categories: general features, sentence pattern features and keywords features.

1) General features

As a microblogging service, Weibo has some characteristics like Twitter, which can be utilized to extract our general features.

a) Number of hashtags: Spam often contain many unrelated hot hashtags to lure users to read them.

b) Number of mentions: In spam, this feature is abused by including many "@username" as unsolicited mentions to expand influence.

c) Number of retweets: A retweet may include several formats like "//@username:". Intuitively, spam are unlikely to be reposted.

d) Diversity of punctuations: Spam tend to use various punctuations to express their opinions. Therefore, we use the number of different punctuations to indicate the diversity.

e) Number of special symbols: In spam, some special symbols like ★,◆,▲ are used to draw users' attention.

f) Number Sequence: To make it easier for interested users to contact them, spammers tend to leave their phone numbers or accounts in their tweets.

g) Number of exclamation mark: Exclamation mark is a punctuation which occurs frequently when spammers advertise their products.

h) Length of tweets: To express more details in 140 characters, the length of spam is relatively longer than regular tweets.

i) Social tools: Besides Weibo, Spammers usually mention their other social tools such as QQ and WeChat (two popular social platform in China) for interested users to follow them.

2) Sentence pattern features

Tweets with fixed patterns are usually generated automatically by robot instead of the users themselves. Therefore, we extract several patterns of such spam and encode these templates as regular expressions.

3) Keywords features

Keywords are very common features in spam detection task. We collect a list of keywords that are often found in spam. However, due to the shortage of spam samples and the impracticality of manual approaches, it is difficult to get a comprehensive keywords list. Therefore we propose to make use of the product advertisement in e-commerce websites to expand the list since most of the spam are commercial advertisements. Our method is based on the intuition that the expressions about products in e-commerce websites are very similar to the expressions of the spam in Weibo. We use the words in the keywords list to query the search engine in the e-commerce website and extract the titles of each product from the query results. These titles are employed to generate new features:

- *N-gram features*

We generate 2 n-gram features, namely, unigram (1-gram) feature and bigram (2-gram) feature. Each feature maintains a list of corresponding n-gram characters whose term frequencies are greater than a threshold.

- *Co-occurrence words feature*

Additionally, we tokenize the titles and employ a co-occurrence metric to obtain the top-n words related to the query. For each title t, the correlation score $S_t(i,q)$ between the word i and the query q is defined by:

$$S_t(i,q)=\begin{cases}1, & if\ f_i > f_q\ and\ f_q \neq 0\\ 0, & if\ f_q = 0\\ f_i/f_q, & otherwise\end{cases} \quad (1)$$

where f_i, f_q are occurrence frequencies of the word i and the query q respectively in title t .Therefore, the correlation score $S(i,q)$ between the word i and the query q in all titles is defined by:

$$S(i,q)=\frac{1}{M}\sum_{t\in M} S_t(i,q) \quad (2)$$

where M is the size of title collection where the word i occurs. In this way, we obtain more keywords automatically which are strongly correlated with the query.

B. Classification Methods Ensemble

Ensemble method [4]has been shown to perform better than single classification model in many tasks. Accordingly, based on the above features, three classical classification algorithms including C4.5, logistic regression and random forest are employed to detect spam respectively. Then, the labels of the instances are determined by a majority vote of each independent classifier.

C. Ensemble Under-Sampling based Strategy

In our spam detection task, the number of spam is much less than that of the non-spam, which causes the class imbalance problem. Therefore, we employ the under-sampling strategy [5] and split the non-spam into R subsets randomly in which the number of instances is slightly larger than the spam to overcome the imbalance. Subsequently, each subset is combined with the whole spam set to make a new balanced dataset. We use these datasets separately to train ensemble classifiers mentioned in Section B. Finally, R different models are generated and the labels are determined by the majority vote of each model.

III. EXPERIMENTAL EVALUATION

A. Dataset And Parameter Settings

To evaluate the performance of our spam detection model, we collect the Weibo dataset randomly from various topics. The details of the dataset are shown in TABLE I:

TABLE I. STATISTICS OF TRAINING AND TEST SET

	total	***# of spam***	***# of non-spam***
Training set	5728	543	5185
Test set	15578	1748	13830

At the keywords expansion stage, we crawl 800 product titles for each query from Taobao website (http://s.taobao.com) which is one of the largest e-commerce websites in China. As to the thresholds of term frequency when extracting unigram and bigram features, we set them to be 100 and 30 respectively. In addition, we select top 20 co-occurrence words which are related to each query tightly.

In our spam detection task, we unify three classification algorithms, including C4.5, logistic regression and random forest which are implemented in WEKA [6] software. Additionally, for the sampling strategy, we set R to be 5 which makes the ratio of spam and non-spam to be approximately 1:2.

B. Evaluation Measures

We employ the measures adopted in the TREC Spam Track [7] to evaluate the performance of our spam detection model. The confusion matrix is depicted in TABLE II.

TABLE II. THE CONFUSION MATRIX FOR THE EVALUATION OF SPAM DETECTION

		Gold Standard – Human Classification	
		Spam	Ham
System's Classification	Spam	a	b
	Ham	c	d

Accordingly, the various effectiveness measures can be defined by: $hm = \frac{b}{b+d}$, $sm = \frac{c}{a+c}$, $lam = logit^{-1}(\frac{logit(hm) + logit(sm)}{2})$, $tp = \frac{a}{a+c}$ and $sr = \frac{c}{c+d}$,where a , b , c and d refer to the number of tweets falling into each corresponding classification category. The hm refers to the ham misclassification rate and sm refers to the spam misclassification rate. The lam metric is to combine both of the two above measures. The logit functions are defined by: $logit^{-1}(x) = e^x / 1 + e^x$ and $logit(x) = ln[x/(1-x)]$. In addition, the ratio of true positive tp denotes the fraction of all spam identified by the system and the sr refers to the spam rate after the system filters the spam. Both tp and sr reflect the effectiveness of the spam detection system.

C. Results and Analysis

1) Performance of classification methods ensemble

To evaluate the performance of the ensemble strategy, we apply three classifiers on the test set separately. The comparative performance is depicted in TABLE III:

TABLE III. PERFORMANCE OF VARIOUS CLASSIFICATION METHODS

	hm%	sm%	lam%	tp%	sr%
C4.5	5.76%	24.10%	12.23%	75.90%	3.31%
Logistic Regression	**3.34%**	23.94%	**9.45%**	76.06%	3.21%
Random Forest	3.97%	23.94%	10.24%	76.06%	3.23%
Our Approach	3.77%	**22.48%**	9.63%	**77.52%**	**3.04%**

As can be observed, the logistic regression method performs slightly better than our model in the lam percentage (9.45%) due to smaller hm percentage. However, we achieve the best result in the tp percentage and the sr percentage, which means we detect more spam and reduce the spam rate remarkably at the expense of a few misclassified non-spam.

2) Effectiveness of sampling strategy

TABLE IV. COMPARATIVE PERFORMANCE OF SAMPLING STRATEGY

	hm%	sm%	lam%	tp%	sr%
Average	4.25%	23.18%	10.36%	76.82%	3.14%
Our Approach	**3.77%**	**22.48%**	**9.63%**	**77.52%**	**3.04%**

To validate the effectiveness of the sampling strategy, we test the training models respectively and obtain the evaluation results of each model. We compare our approach with the average results of these models. The experiment results in TABLE IV show that, our approach achieves better results in all five metrics.

3) Effectiveness of keywords expansion

To validate the expanded keywords features, we generate a new feature set which removes the n-gram features and the co-occurrence features.

TABLE V. COMPARATIVE PERFORMANCE OF DIFFERENT FEATURE SET

	hm%	sm%	lam%	tp%	sr%
Shrunken feature set	4.28%	41.26%	15.07%	58.74%	5.46%
Our Approach	**3.77%**	**22.48%**	**9.63%**	**77.52%**	**3.04%**

From TABLE V, we can see that our approach has a remarkable performance compared with the method without the external keywords expansion. Additionally, since the keywords are expanded automatically, it is unavoidable to obtain some noisy keywords. However, our model is not impacted by these noisy words according to the results, which reflects the robustness of our approach.

IV. CONCLUSIONS

In this paper, we present a unified spam detection model to filter the spam in Weibo. To overcome the shortage of spam instances and the huge costs of manual approach, we utilize advertisements from e-commerce websites to expand our keywords features automatically. In addition, an ensemble under-sampling based strategy is used to overcome the imbalance of data distributions and boost the performance of spam filtering. Preliminary experiments have demonstrated the effectiveness of our model. We also develop a prototype based on this spam detection model and achieve remarkably performance in Weibo data. In our future work, we will examine more features such as the users' profiles to improve the performance of our spam detection model.

ACKNOWLEDGMENT

This research is supported in part by NNSFC grants #91024030, #91224008, #61172106, #U1435221, #71462001 and grant #2013ZX10004218.

REFERENCES

[1] G. Stringhini, C. Kruegel, and G. Vigna, "Detecting spammers on social networks," in Proceedings of the 26th Annual Computer Security Applications Conference, Austin, Texas, pp. 1-9, 2010.

[2] E. Tan, L. Guo, S. Chen, X. Zhang, and Y. Zhao, "Spammer Behavior Analysis and Detection in User Generated Content on Social Networks," in IEEE 32nd International Conference on Distributed Computing Systems (ICDCS), pp. 305-314, 2012.

[3] A. H. Wang, "Detecting spam bots in online social networking sites: a machine learning approach," in Proceedings of the 24th annual IFIP WG 11.3 working conference on Data and applications security and privacy, pp. 335-342, 2010.

[4] S. Y. Bhat, M. Abulaish, and A. A. Mirza, "Spammer Classification Using Ensemble Methods over Structural Social Network Features," in IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT),, pp. 454-458, 2014.

[5] G. Geng, C. Wang, Q. Li, L. Xu, and X. Jin, "Boosting the Performance of Web Spam Detection with Ensemble Under-Sampling Classification," in Fourth International Conference on Fuzzy Systems and Knowledge Discovery, pp. 583-587, 2007.

[6] O. Maimon, and L. Rokach, *Data mining and knowledge discovery handbook*: Springer, 2005.

[7] R. Y. Lau, S. Liao, R. C. W. Kwok, K. Xu, Y. Xia, and Y. Li, "Text mining and probabilistic language modeling for online review spam detecting," *ACM Transactions on Management Information Systems,* vol. 2, no. 4, pp. 1-30, 2011.