Modeling and Evaluating Information Diffusion for Spam Detection in Micro-blogging Networks

Kan Chen, Peidong Zhu, Liang Chen and Yueshan Xiong
School of Computer, National University of Defense Techology
Changsha, Hunan - CN
[e-mail: jeffee,pdz,liang,ysx@nudt.edu.cn]
*Corresponding author: Kan Chen

Received Spetember 3, 2014; revised February 19, 2015; revised April 8, 2015; revised May 26, 2015; accepted June 26, 2015; published August 31, 2015

Abstract

Spam has become one of the top threats of micro-blogging networks as the representations of rumor spreading, advertisement abusing and malware distribution. With the increasing popularity of micro-blogging, the problems will exacerbate. Prior detection tools are either designed for specific types of spams or not robust enough. Spammers may escape easily from being detected by adjusting their behaviors. In this paper, we present a novel model to quantitatively evaluate information diffusion in micro-blogging networks. Under this model, we found that spam posts differ wildly from the non-spam ones. First, the propagations of non-spam posts mostly result from their followers, but those of spam posts are mainly from strangers. Second, the non-spam posts relatively last longer than the spam posts. Besides, the non-spam posts always get their first reposts/comments much sooner than the spam posts. With the features defined in our model, we propose an RBF-based approach to detect spams. Different from the previous works, in which the features are extracted from individual profiles or contents, the diffusion features are not determined by any single user but the crowd. Thus, our method is more robust because any single user's behavior changes will not affect the effectiveness. Besides, although the spams vary in types and forms, they're propagated in the same way, so our method is effective for all types of spams. With the real data crawled from the leading micro-blogging services of China, we are able to evaluate the effectiveness of our model. The experiment results show that our model can achieve high accuracy both in precision and recall.

Keywords: spam detection, information diffusion, micro-blogging, RBF

Kan Chen's work was supported by the National Natural Science Foundation of China under Grant No. 61170285 and Grant No.61379103.

http://dx.doi.org/10.3837/tiis.2015.08.014 **ISSN : 1976-7277**

1. Introduction

Due to its convenience and flexibility, micro-blogging has quickly attracted a great many of attentions and become one of the hottest online social networks (OSNs) in the world. However, with the increasing popularity, micro-blogging is facing serious problems of spams.

The term spam is used to describe information and messages produced by deliberate, colluded and purposeful human activities. The people who launch spam campaigns are known as spammers. There're many different representations of spam on a system basis. In email, spam is regarded as junk mail [1]. In e-commercial market, spam refers to the action of rating without real trade [2]. In this paper, we define spam in micro-blogging as a group of repeated reposting or commenting to make a post visible and credible to others.

In micro-blogging networks, spams can be used in many scenarios, such as self-promotion and word-of-mouth marketing. They can also be used for many malicious purposes. Notable attacks include advertisements abusing [3], rumor spreading [4] and malware distribution [5]. The corrupt influences are also various. First, spams annoy users with advertisements and junk messages. Advertisement is an important component of spam. Users are easily convinced by the high amount of boastings. With the increase of junk messages, non-spam messages are easily flooded, which greatly declines users' experiences. Second, some spams contain porn, violent, or other illegal contents. They are propagated in OSNs, resulting in heavy pollutions to the Internet environment and a large number of financial wastes to deal with. Besides, virus, malware, and phishing links can also be transmitted by spammers. They invade in victims' machines and steal the victims' information for criminal purpose. Last, rumors or illegal opinions spreading in OSNs may bring about unpredictable turbulence and even cause threat to security and peace. In March, 2011, a message about "salt will be exhausted in a few days" is spread by millions of people in China's leading micro-blogging. As a result, in many countries people rush for salt, resulting in widely selling out in many supermarkets. Although the rumor was proven to be fake later, large-scale social panic has been caused.

In order to deal with the pollutions and malicious effects caused by spams, many OSNs have proposed their own mechanisms for spam detection. These approaches can be categorized as user-based and post-based. User-based methods aim to figure out spammers, while post-based methods focus on spam posts. In user-based approaches, features like user profiles and user behaviors are studied to distinguish the spammers. However, the problem is that in order to prevent from being detected, spammers are likely to pretend to be non-spam by modifying their profiles and behaviors. Even worse, some non-spam users also engage into spam spreading for financial benefits, making it increasingly difficult to distinguish. Post-based approaches can be functionally categorized as text-based [1] and link-based [6] according to the features used. They are effective in certain fields but effectless in others. In our opinion, a good detection method should be:

- *Accurate*: There's no doubt that accuracy is the most important factor to judge a detection work. It directly decides whether the work is effective or not.

- *Robust*: Here robust means that the detection algorithm is difficult to escape. Once a detection tool is proposed, spammers may modify their behaviors to avoid being detected. Unfortunately, many of previous works are not robust enough.
- *Comprehensive*: There're many different types of spam. For example, the advertisements spam, opinion spam, malware spam and so on. A good detection method should not only be capable to deal with one specialized type.
- *Promptness*: Spam is also regarded a type of attack. Many detection tools are designed to distinguish spammers or spam posts after the spam campaigns. Although they're effective and accurate, the attack has been launched successfully and bad influences have been caused. A good detection tool should distinguish spams before they take effect.

In this paper, we attempt to propose a spam detection tool to meet the needs discussed above. Our contributions can be concluded as: First, we proposed a diffusion model to quantitatively analyze the interactions from spammers and benign ones. Different with the features introduced by previous works, the diffusion patterns analyzed in our model are not determined by any single user and won't be affected by users' behavior modifications. Thus, our model is more robust. Second, with real data crawled from China's leading micro-blogging service, we trained a classifier with an RBF Method. The effectiveness is validated by the experimental results. Although spams differ with each other in categories and contents, they're propagated in the same way. So our method can theoretically deal with all kinds of spams.

The remainder of this paper is organized as follows. In section 2 we introduce the backgrounds of our work. Related works are reviewed in section 3. Then we describe our diffusion model in detail in section 4 and analyze with real data in section 5. In section 6 we present a spam detection method with RBF method and examine the effectiveness in section 7. And the conclusion is in section 8.

2. Overview of Spam Diffusion in Micro-blogging Networks

2.1 Micro-blogging networks

Micro-blogging is a newly presented social platform for communication and information sharing. As we can infer from the name, a micro-blogging network is thinner and lighter weight than traditional blog networks. The contents in a micro-blogging network are restricted to be very short, such as 140-characters per post. This restriction allows users to read in trivial times, and quickly becomes popular due to the facilitation and flexibility it provides.

Since the emergence of Twitter, micro-blogging has attracted a great many of attentions and become one of the most popular OSN services. Many Twitter equivalent networks are introduced, such as Plurk, Squeelr, Jaiku and Weibo [7] of China. These networks have generated tremendous amounts of micro-blogging data, which have become important resources for social networks researchers.

Despite the differences in interfaces and functions, these micro-blogging networks all follow the basic conceptions of Twitter.

(1) Post & Poster

The contents published in micro-blogging are termed differently. For example, in Twitter they're known as tweets, while in Weibo of China they're named as weibos. In this paper, we use "post" represent an item published in micro-blogging networks. Texts, pictures, videos, and URLs are all allowed in a post. The user who publishes a post is regarded as the "poster". Every post has only one poster, while every poster can publish many posts.

(2) Repost & Reposter

If a user likes a post and wants to share it in his own board, he can repost it with a symbol "@" indicating the original poster. New contents are allowed to append in front of the original post. Anyone who reposts a post is named as the "reposter". Let's use an example to explain the relations between poster and reposter. *A* publishes a post *p*, and *B* reposts it. Then in *B*'s board, a new post *p'*, which is original from *p*, is shown. *B* is the reposter of *p*, and at the same time, also the poster of *p'*.

(3) Comment & Commenter

Comment is usually submitted to show interest, agreement or disagreement on a post or feedback to another comment. Different from repost, comment only shows in the original poster's board. The user who publishes a comment is named as the "commenter".

(4) Following & Followed

The relations in micro-blogging are directed and start from the process of following. Once a user is followed, his new posts are pushed to his followers as soon as published. Thus, the more followers he gets, the more probably his posts are seen. In some works, the number of followers is treated as a measurement of user influence. The more followers a user gets, the more influential he is.

2.2. Information Diffusion in micro-blogging networks

Before talking about information diffusion, we first would like to introduce how a user's posts are seen by another. In micro-blogging networks, it happens in the following scenarios.

1) If *A* follows *B*, then *B*'s new posts are pushed to *A* as soon as published.

2) If *A* follows *C*, and *C* has reposted *B*'s post, then *C*'s repost is pushed to *A* just as the scenario above. Thus, *A* can see *B*'s post indirectly through *C*. There may be more go-betweens connecting *A* and *B*, and a reposting chain is built.

3) Other scenarios. For example, if *A* gets *B*'s URL, then *A* can visit *B*'s board directly. A possible way to achieve this is to link URLs in some famous blogs or news sites. In this situation, the influence of other sites outside the micro-blogging network itself should be considered first, but it is out of scope in this study. Another example is that *B*'s post is recommended by Weibo in the public board. However, due to the huge amount of posts generated per seconds, the probability of being chosen is tiny.

From the discussions above we can conclude that following and reposting are crucial in information visibility and information diffusion. In fact, following results in post visibility, while reposting brings about post propagation. Through reposting, a post is propagated from a user to another, which increases the volume of followers who can see the post and provides more opportunities for a post to be spread further. It's noting that although the action of commenting does not result in information propagation, it in fact helps to enhance the credibility of the

original post. A highly commented post is always thought to be hot and attracting. People would like to focus and believe what the post says due to the fact of social conformity [8].

As we have recognized the importance of following, reposting and commenting, in this paper, we regard them as the core reasons of information diffusion. For a micro-blogging user, if he wants to make his post known to others, he should either get more followers or get a high number of reposts. However, both the follower number and the repost count are not determined by himself. What the user can do is trying to ingratiate his posts with others, but it requires a long-time maintenance and the result is still unpredictable. Due to the difficulty in promotion, referring to spammers turns out to be a quicker and easier choice.

2.3. How spam works in micro-blogging networks

Micro-blogging networks are updated every few minutes or even seconds, generating tremendous amounts of micro-blogging data. The potential values of the rapid information diffusion have attracted a lot of attentions and made micro-blogging as the main battle field of spam campaigns.

On the urge of commercial, political, or even illegal demands, someone wishes to promote his products or opinions for public attentions. But he could not attract enough focuses if he is not influential enough. In this situation, employing spammers for promotion turns to be a more effective and economical choice. In 2010, a spam campaign broke out between two of China's biggest Internet companies, 360 and Tencent. They all employed a large scale of spammers to demean each other with inflammatory remarks and slurs. This campaign lasted longer than two months and became the most influential spam war in Chinese history.

In this war, 360 and Tencent, who are known as the hidden payers, first published a few of posts with negative information of each other. Then a group of spammers are hired to repost and comment with different fake identities and roles to make the topic hot. In this way, the adversary's reputation is undermined. The relations between the hidden payers and spammers are shown in **Fig. 1**.

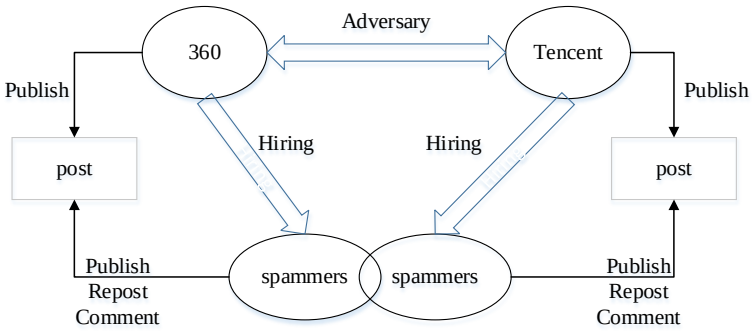

Fig. 1. Relations and interactions of hidden payers (360 and Tencent) and spammers

Spammers are always hired from some specific spamming sites. Hidden payers first release a set of tasks, including posting, reposting, commenting and others. Potential spammers would visit these sites and accept the tasks if the payments are satisfying. Once all the tasks are completed, the spam campaign is finished. People are attracted and convinced by the huge

amount of reposts and comments. And what's more, they may also join in the groups of propagation. In this way, the posts become widespread and the hidden payers successfully achieves their purposes.

The payers' purposes are various. Some want to sell products, while some eager to propagate rumors or libels for private or political reasons. There're also some trying to perform phishing attacks, or install malware onto victim's machine for information stealing. Besides, some other reasons, such as self-promotion, can also lead to a spam campaign. No matter what the payer really aims for, his goal of hiring spammers is to spread his post to a maximum number of users and what's more important, to convince them.

3. Related Work

Spam has been studied for a long time in email systems [9-12]. In recent years, as the popularity of OSN, spams in OSN are attracting more and more attention. Many popular sites have become battle fields of spam campaigns.

Kurt analyzes the suspended accounts in Twitter [13]. Grier's work is also based on Twitter [14]. Gao studies the spams in "wall" messages of Facebook [15]. Irani investigates the spams in Myspace [16]. Except these top known sites, there're also some works based on search engine [17], video sharing sites [18], forums [19], blogs [20] and so on. The basic conception of spam detection is to extract features that can distinguish spams from non-spams. Many features have been proven useful. A comparison of features used by previous works is listed in **Table 1**.

Content features refer to some explicit texts or items that are usually used in spam contents. For example, the terms "discount", "sold-out" and "on sale" are usually used in advertisement spams. Zhang [21] studies the utility of reviews based on natural language features. Jindal [22] studied the spam product reviews of Amazon.com. Duplicated reviews are first treated as spams, and then through supervised learning with manually labeled training examples, other two types of spams are detected. Zhang [23] detected spammers in micro-blogging networks through duplicated contents. Their work is built on the assumption that spammers tend to copy tweets from normal users to pretend to be normal. Wang [24] treats spam microblogs as all kinds of advertisements and uses an SVM machine learning method to perform spam detection. The feature vector is a combination of users' social network features and the textual features of microblogs. The shortcoming of content features is that it's only suitable for specific type of spams. In micro-blogging networks, many users repost without adding any words, or comment only with emoticons. Thus, there're few text features available.

Table 1. Comparison of features used by related works

	Post-based			User-based		
	Text	keyword	link	Profile	Behavior	Relation
Thomas [13]			blacklist			
Grier [14]			url similarity & blacklist			
Gao [15]			blacklist			
Irani [16]				User profile		

Wang [17]			blacklist			
Shin [18]	Text feature		url length & url number		User activity	
Benevenuto [19]					User activity	
Rajadesingan [20]	Text feature					User relation
Zhang [21]	Text feature					
Jindal [22]	Duplicated text	keyword				
Zhang [23]	Duplicated text					
Wang [24]		keyword		Follower count & isverified		
Zhang [25]			Link Similarity			
Liu [26]					Visiting pattern	
Liu [27]					User activity	
Han [28]			url use	follower count	User activity	
Our work					Interaction feature	User relations

Links are commonly used components of spams. Many detection approaches are based on link examination. The blacklist methods leverage famous existing blacklist services, such as Google Safebrowsing, URIBL, and Joewein, to detect spam contents with URLs redirecting to malicious sites [13]. Grier uses blacklists to detect spam posts and comments in Twitter [14]. His result shows that 8% of the URLs posted on Twitter point to phishing, malware, and scams sites listed on blacklists. His work also indicates that blacklists are too slow at identifying new threat, allowing more than 90% of visitors to view a page before it becomes blacklisted. Zhang's work [25] first clustered accounts with link similarity. The posts with similar URLs are regarded in the same campaign. Then multiple features are used to distinguish the spam campaigns.

User profile refers to the information used to describe a user's real identity. Irani uses both categorical and free-form features to detect spam profiles in MySpace [16]. His approach can detect spammers as soon as they registered, but it's easy to modify one's profile and then escape from detection.

User behavior is another important factor for spam detection. Liu [26] studies user's visiting patterns of spam pages to separate spam pages from ordinary ones. The same technology can also be used to detect spams in search engine [27].

Han [28] attempts to evaluate the probability of users being spammers in micro-blogging networks. A probabilistic graphical model is proposed, where user attribute features are taken as the input variables, behavior features are taken as the observed variables and probabilities of spammers are the hidden variables. The performance of his model is evaluated both in Sina and Twitter micro-blogging networks.

Information diffusion has been studied both theoretically and experimentally. Some aim to figure out the diffusion patterns and produce some models to evaluate [29, 30]. Some try to reveal the procedures of diffusion [31]. Gruhl [32] studied the dynamics of information propagation in blogspace at two levels, a macroscopic characterization of topic propagation and a microscopic characterization of individual propagation. With a viral email experiment involving 31,183 individuals, Iribarren [33] studied the impact of human activity patterns on information diffusion. He found that information travels at an unexpectedly slow pace, which contraries to traditional models. Yang [34] developed a linear influence model to analyze the global influence of a node on the rate of diffusion through the implicit network. They found that patterns of influence of individual participants differ significantly depending on the type of the node and the topic of the information.

These works mostly focus on diffusion itself, or solving the problem of maximizing spread circles and influence effects. In this paper we analyzed user interactions, the underlying driving forces of information diffusion, and aim to solve the problem of spam detection. Our assumption is that spammers and normal users perform differently in interactions, and these differences may only be visible in the procedures. Only focus on the consequences would miss useful features.

4. Diffusion Model

4.1. Model Overview

In our model, we analyze information propagation from two views, user relations and user interactions. From the view of user relations, we define two features to measure the relations among followers, reposters and commenters. And from the view of user interactions, we mainly focus in time space and analyze user interactions with timeline.

The ontologies and corresponding relations of our model are shown in **Fig. 2**. Every user u has a number of followers and publishes a set of posts. Every post contains a number of reposts and comments. For each post, we mainly focus on two fields, the poster and the post time. The information of poster is used in the relation-based phase, while the information of post time is used in the interaction phase.

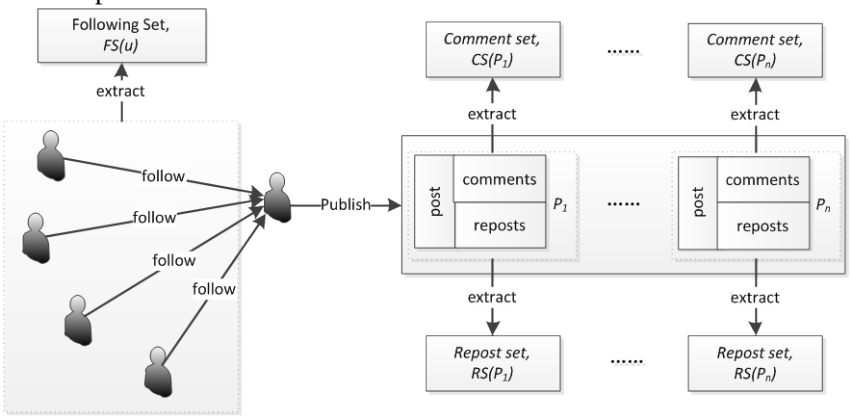

Fig. 2. Ontologies and relations in our model

It's worth to say that in the former section we explain that every repost is actually a new post in the reposter's board. It can also be reposted and commented, and in this way, an interaction chain is created. Of course the chained interactions would affect the propagation of the original post. But through a deep observation of our experiment dataset, we found that more than 99.5% of the reposts do not have any reposts or comments at all. So in this work, we just ignore the chained interactions.

The notations used in our model are shown in **Table 2**.

Table 2. Notations and functions for our proposed model

Symbols	Descriptions
U	The Collection of users
u	A user in micro-blogging
$P(u)$	The collection of u's posts
$FS(u)$	The collection of u's followers
p	A post in micro-blogging
$RS(p)=[r_1,r_2,\ldots,r_n]$	The repost collection of p
$CS(p)=[c_1,c_2,\ldots,c_n]$	The comment collection of p
$Poster(p)$	The poster of p
$\|C\|$	size of collection C
$t(p_i)/t(r_i)/t(c_i)$	Post time of $p_i/r_i/c_i$

4.2. Relation-based phase

Although only the followers can receive notifications of new posts, it's not necessary that all the reposters and commenters have followed the posters. In the relation-base phase, we're motivated to figure out the different relations between followers and reposters/commenters.

Every user has a unique following set represented as $FS(u)$. Each post has a reposting set and a commenting set, represented as $RS(p)$ and $CS(p)$ respectively. It's noting that any of the sets might be empty.

For a given post, two features are defined from the view of user relations, the coefficient of repost (*cor*), and coefficient of comment (*coc*). They're calculated as

$$cor(p) = \frac{\left|FS(u) \cap Poster\left(RS(p)\right)\right|}{\left|Poster\left(RS(p)\right)\right|}, \quad \left(u = Poster(p)\right) \tag{1}$$

$$coc(p) = \frac{\left|FS(u) \cap Poster\left(CS(p)\right)\right|}{\left|Poster\left(CS(p)\right)\right|}, \quad \left(u = Poster(p)\right) \tag{2}$$

The values of *cor* and *coc* in fact reveal the proportions of reposters and commenters who are also followers of the poster. They are built on the following assumption.

Assumption 1. The cor and coc of non-spam posts are much higher than that of spam posts.

Intuitively, a user's posts are mainly reposted and commented by his followers, so the values of *cor* and *coc* should be high. As for the spammed posts, the hired spammers do not necessarily have followed the poster. Thus, the values of *cor* and *coc* of spam posts should be smaller than those of non-spam posts. We will examine our assumption in the procedure of experiment.

4.3. Interaction-based Phase

The interactions in our model refer to the actions of reposting and commenting. We believe that spammers and non-spam users act differently because they participate in micro-blogging networks with different purposes. Non-spam users treat micro-blogging as a platform for communication and information gathering. They would like to enjoy the interactions with others. As for the spammers, they only regard micro-blogging as a tool to earn financial benefits and prefer to finish their tasks as soon as possible. We are motivated to figure out these differences and use them to detect spam posts. First we would like to answer the following questions:

- How long does a post last to attract interests?
- How long does it take a post to become hot?
- How often a post is reposted and commented?

These questions describe how a post is propagated and how users interact throughout the post. In order to answer these questions, four features are produced to characterize the diffusion capabilities. Each of the features is also embedded with an assumption waiting to be validated in the following procedures.

From the definitions of relation-based features we can see that the repost feature *cor* and the comment feature *coc* share similar formula but different data sets. In order to avoid duplicate descriptions, in this section we only consider the features of reposts. The features of comments are calculated with similar formulas with comment set *CS(p)* instead of repost set *RS(p)*.

Average Duration (ADu): Duration is used to describe how long a post can continuously gain focuses and interests. Duration is represented as:

$$Du(p) = t(r_N) - t(r_1), \quad (r_i \in RS(p), N = |RS(p)|) \tag{5}$$

The value of duration is highly related to the number of reposts. So we define average duration to calculate the time every repost lasts. Average duration is calculated as:

$$ADu(p) = \frac{Du(p)}{N}, \quad (r_i \in RS(p), N = |RS(p)|) \tag{6}$$

Assumption 2: With an equal number of reposts, non-spam posts last longer than spam posts.

In the relation-based phase, we assume that spammers contribute to most of the diffusion of spam posts. They repost and comment to get paid. However, the payment is limited. Some will not get paid if they finish the task beyond the limitation. Thus, all the spammers purpose to finish the task as soon as possible and then transfer to another. Besides, once the task is finished, the spammers will not feedback again. As a result, the spam campaigns always take place within short times and end quickly.

By contrast, non-spam posts spread differently. Due to individual differences, users visit micro-blogging networks with their own habits. Thus, the times when they see a post vary greatly. Some may see it at the first time, while some may see hours or even days later. As a result, we can always find slow readers from the followers of the non-spam users. Besides, even after reposting, they may also feedback with each other. Under this consideration, we assume that normal posts generally last longer than spam posts.

Fir-time(FT): Fir-time is used to describe the time interval between the original post and its first repost. Fir-time is calculated as:

$$FT(p) = t(r_1) - t(p), \quad (r_i \in RS(p)) \tag{5}$$

Assumption 3: The fir-times of non-spam posts are generally smaller than that of spam posts.

Micro-blogging is designed lightweight and flexible. Users are allowed to read, repost and comment in trivial times. Besides, with the help of push technologies, micro-blogging is regarded as a nearly real-time platform. New posts are seen by the followers at the first time. And then, these followers can decide whether to repost or not immediately. So the followers of non-spam users contain fast responders. Accordingly, we assume that the fir-time of non-spam posts would be quite small.

However, the spam posts get their reposts from a different way. First the payers publish the target posts in micro-blogging networks. Then a series of tasks are released with URLs of the target posts. Spammers would follow and finish these tasks to repost through the URLs. It's clear that this procedure costs more time. So we assume that the fir-times of spam posts are much higher than the non-spam posts.

Average Interval (AI): Average Interval is the mean of intervals between neighboring reposts. It is calculated as:

$$AI(p) = \sum_{i=1}^{N-1} \frac{t(r_{i+1}) - t(r_i)}{N-1}, \quad (r_i \in RS(p), N = |RS(p)|) \tag{3}$$

Assumption 4: The average intervals of spam posts are much smaller than that of non-spam posts.

As discussed above, spammers rush to repost and comment as soon as the task is released because the payment is limited. As a result, the reposts present a burst property, making small intervals between neighboring reposts.

Variance of Interval (*VI*): VI is defined as the variance of intervals.

$$VI(p) = \sum_{i=1}^{N-1} \frac{[t(r_{i+1}) - t(r_i)]^2}{N-1} - AI(p)^2 \quad (r_i \in RS(p), N = |RS(p)|) \tag{4}$$

Assumption 5: The VI of spam posts are much smaller than that of normal posts.

Variance is used to evaluate how far a set of numbers is spread out. A small variance indicates that the data tends to be close to the mean and hence to each other, while a high variance indicates that the data is very spread out around the mean and from each other. As described in assumption 3, a spam campaign usually bursts in a short time period. Thus, the intervals could be small and close to each other. However, for the normal posts, as every normal user has his own visiting habit, the differences would be larger.

Under the definitions of these features, for a given post p, its interaction-based feature is represented as $\vec{I} = \langle ADu(p), FT(p), AI(p), VI(p) \rangle$. It's noting that although there're four features, every feature has two values. One is from the repost set, and the other is from the comment set.

5. Analysis with Diffusion Model

5.1. Data Collection

Our experiment data is crawled from Weibo, the leading micro-blogging network of China. With a population of 5 billion registered users, Weibo has attracted a significant many of attentions and provides an open platform for spammers.

Our dataset should contain both spam posts and non-spam posts. However, no such a collection is publicly available to indicate whether a post is spam or not. So what we should do first is to collect some pre-distinguished spam and non-spam posts.

(1) Spam Collection

There're many kinds of spams, such as advertisement spam, opinion spam, malware spam et al. Although different in types and contents, they're organized and propagated in the same way. So we only need to analyze one type. In all of them, advertisement spam is much more pervasive and easier to distinguish manually. Many companies have realized the importance of micro-blogging networks in advertising. They hire a number of spammers to promote their products. So we use the spam of advertisement as prototype to study the features.

First we manually collect some advertisement spam posts. New published ones are ignored because they may still be reposted and commented after the crawling, which may affect the propagation features. We only choose the posts published at least three months before. As Weibo is a nearly real-time platform, the update cycles of messages are very small. We assume that after three months, the post seldom gain any reposts or comments.

When we crawl a post, the poster's following set is also collected. There's a problem that between the intervals of getting the first and last repost/comment, the poster's following set may change with time. Actually the effects of this change are so limited that we can just ignore it. The reasons lie in two folds. First, the probability of sudden change of following sets is small. Even if a poster experiences such a sudden change, the others' posts would reflect the real properties. Second, small changes of following sets would not affect the whole features as the number of reposts/comments is too much higher, which is always larger than 1000.

Once the spam posts are gathered, keywords are extracted from the texts. Taking advantage of the search engine provided by Weibo, we could search with these keywords to get correlative posts. This work lasts two weeks, from 1 to 15, March 2014.

Intuitively, most OSNs users just prefer to ignore when they see advertisements. Few involves in reposting or commenting. As a result, most advertisement posts are reposted for few times. This assumption is validated by our experiment result. In the search results, nearly 85% of the posts are reposted less than 50 times. In fact, about 80% of the posts are reposted less than 10 times. About 10% of the posts are highly reposted or commented. They're either too attractive to

refuse or, more likely, spams.

We conduct a filter on repost count to get rid of posts with repost rates less than 100. After this procedure, we get 1524 spam-like posts.

Another factor we use to identify spams is the number of followers. Generally, if a user has few followers, then his post is less probably to be seen and reposted by a great many of others. If it happens, it's more likely to be spam. Of course, we don't deny that there're some coincidences that a post from a poorly followed user gets many reposts and comments, we just argue that the probability is tiny.

We calculate the numbers of followers of all the original posters and compare with the count of reposts. Those with few followers but high repost counts are included. And finally, we get a dataset with 1372 spammed posts. Each of them is averagely reposted for 2453 times and commented for 1320 times.

(2) Non-spam Collection

Weibo provides mechanisms for real-name verification, and those who have been verified are marked with symbol “V”. We assume that a verified user is less likely to be spammer because he’s supervised through his real-name.

Weibo provides a public page named “the celebrity Tang”, in which famous and influential people are verified and added. They are categorized according to their career experiences, such as IT, education, sports, and media and so on. We choose those categorized in IT and education as the posters of our non-spam collection because we assume that they’re less likely to spam or pay for spam. Then we conduct a random sample from their posts. We finally gathered 2946 non-spam posts, each of which is averagely reposted for 1973 times and commented for 1664 times.

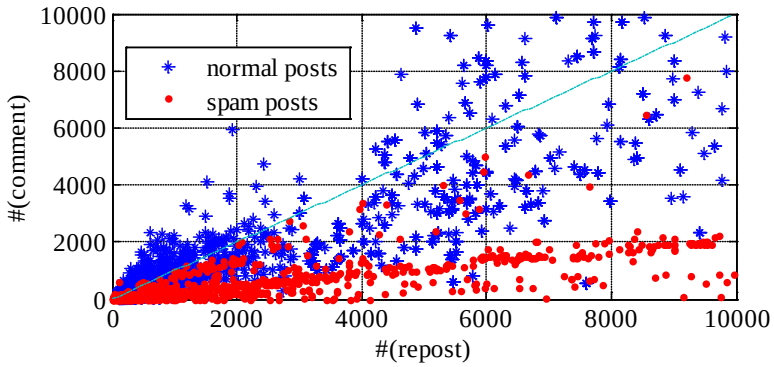

Fig. 3. Distribution of the count of reposts and comments

Finally, we get a dataset with 1372 spam posts and 2946 non-spam posts. The distributions of the count of reposts and comment are shown in **Fig. 3**. We divide our dataset into a training set and a test set randomly. Each set contains the equal number of spam and non-spam posts.

5.2. Analysis of Relation-based Phase

First we would like to analyze the distributions of *cor* and *coc* for both spam and non-spam posts.

In assumption 1 we assume that the *cor* and *coc* of non-spam posts are much higher. The distributions of **Fig. 4** validate our assumption.

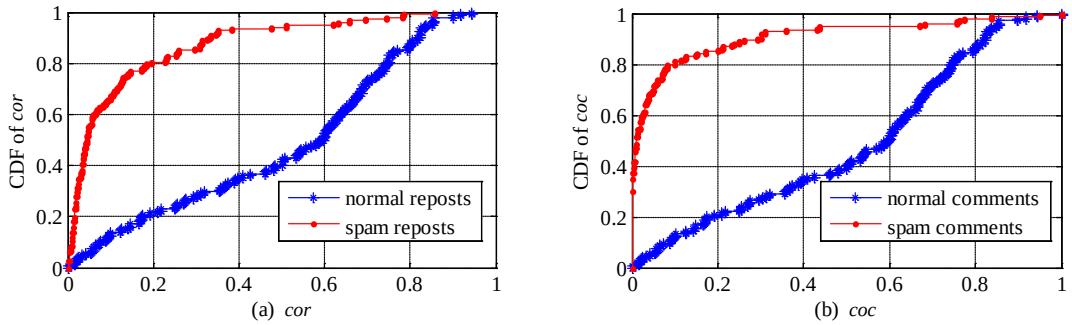

Fig. 4. distributions of *cor* (a) and *coc* (b)

The value of *cor* measures the proportion of reposters who are also followers of the poster. While the value of *coc* measures the proportion of commenters who are also followers of the poster. From the figure we can see that most of the spam posts (more than 80%) get 20% of their reposts and only 10% of their comments from followers. The rest are all from strangers. By contrast, the normal posts get significantly more reposts and comments from followers. About more than 60% of them get more than half of the reposts and comments from followers.

5.3. Analysis of Interaction-based Phase

The interaction-based phase consists of four features, average duration (*ADu*), fir-time (*FT*), average interval (*AI*), and variance of intervals (*VI*). The distributions of these features are shown in **Fig. 5-8**.

Fig. 5 depicts the CDF distribution of average duration. It shows that more than 80% of the spam reposts persist for a shorter time than 20 minutes, comparing with 10% of the normal reposts. Besides, more than 80% of the spam comments last less than 30 minutes, comparing with less than 5% of the normal comments.

As explained in the previous sections, the differences result from the different behaviors of normal and spam users. Normal users visit micro-blogging with their own habits, while the spammers need to rush for limited payments. As a result, the spam posts present a burst property and last less time.

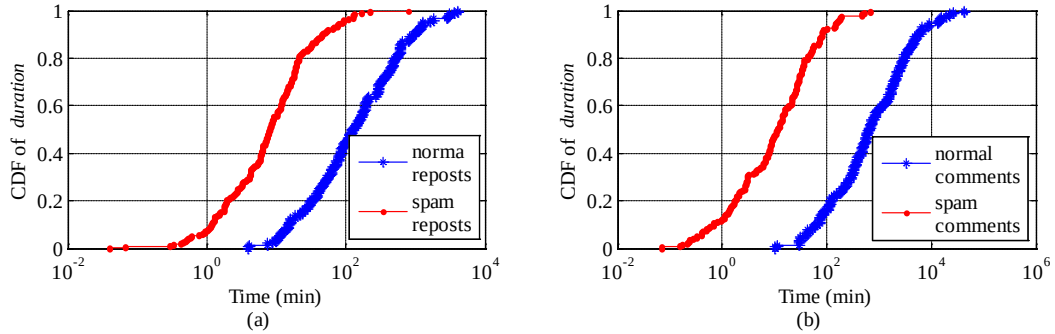

Fig. 5. CDF of average duration under reposts (a) and comments (b)

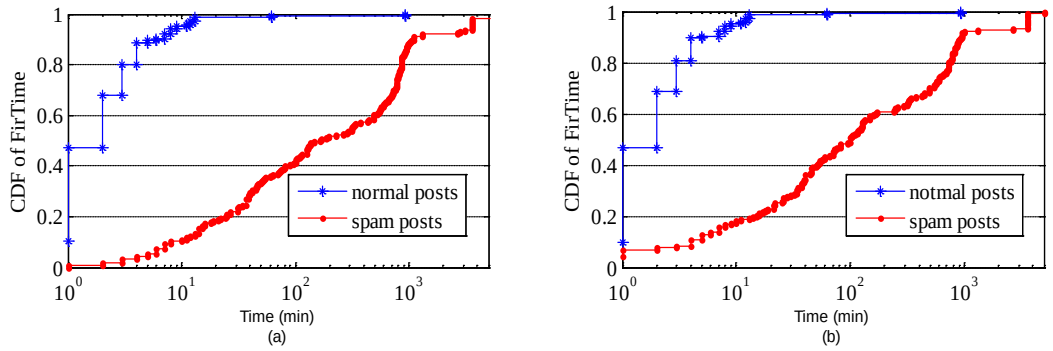

Fig. 6. CDF of fir-time under reposts (a) and comments (b)

Fig. 6 illustrates the difference between spam and non-spam posts in the distributions of fir-time. We can see the differences are obvious. Nearly half of the non-spam posts get their first repost and comment within one minute. By contrast, within the same time, only less than 2% of the spam posts get their first reposts and 6% get their first comments. In ten minutes, about 95% of the normal posts can get their first reposts and comments, comparing with less than 20% of the spam posts.

This consequence can be explained from the procedures of reposting/commenting. Due to the push technologies used in micro-blogging networks, users are able to interact in a real-time way. So normal posts can get their first reposts/comments in short times. Spammers actually don't visit the target post through the pushed notifications. Instead, they get the URLs from the payers, and repost/comment under the payer's direction, which may cost more time.

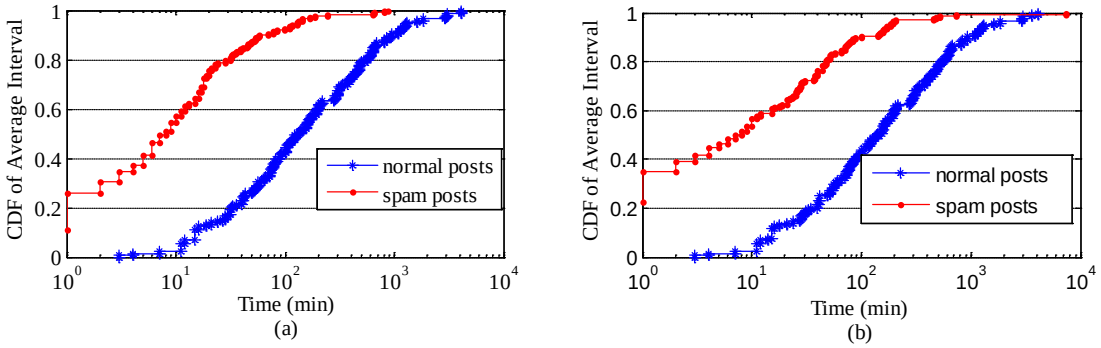

Fig. 7. CDF of Average Interval under repost (a) and comment (b)

The CDFs of average interval are shown in **Fig. 7**. We can also figure out great differences. Nearly half of the spam reposts are published with average intervals of less than 10 minutes, comparing with less than 5% of normal reposts. The comments represent similar distributions with the reposts. Besides, about 25% of the spam reposts and 35% of the spam comments are published with an average interval of 1 minute. The non-spam posts can hardly get that frequency.

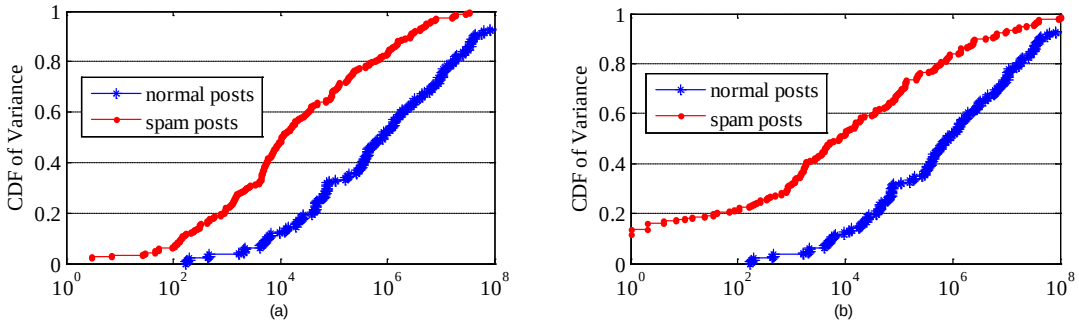

Fig. 8. CDF of variance under repost (a) and comment (b)

The distributions of variance are shown in **Fig. 8**. From the figures we can see that the variances of spam intervals are much smaller than the normal ones. It indicates that comparing with the non-spam reposts and comments, the spam ones are more close to each other.

6. Span Detection with RBF Methods

We model spam detection as a binary classification problem where our goal is to classify each post as spam or non-spam with the features investigated in our model. We use an artificial neural networks (ANNs) classifier toward this goal. Specifically, we run a radial basis function neural network (RBF-NN).

ANNs were proposed to mimic the way biologic neural networks work. With the capability to learn complex relationships between inputs and outputs, ANNs have been widely used in many applications, including function approximation, time series prediction, classification, and system control.

RBF-NN is a feed-forward artificial neural network. It typically has three layers: an input layer, a hidden layer and an output layer. The simpler network structure and well-developed theoretical analysis of RBF-NN introduces a number of significant advantages over multi-layered networks [35].

In an RBF-NN, the input is modeled as a vector of real numbers $\mathrm{X} \in R^n$. Radial basis functions are used as activation functions in the hidden layer. Then the linear output is given by

$$F(x) = \sum_{i=1}^{N} \omega_i \varphi\left(\left\| X - X_c \right\|\right) \tag{7}$$

Where N is the number of neurons in the hidden layer, X_c is the center vector, and ω_i is the weight of neuron i. Let's use σ as the width, and then the radial basis function is commonly taken to be Gaussian and represented as

$$\varphi\left(\left\| X - X_c \right\|\right) = \exp\left[- \frac{\left\| X - X_c \right\|^2}{2\sigma^2} \right] \tag{8}$$

Different with general neural network methods, the training of RBF-NN is typically done in two different phases. First the width and centers should be evaluated and then the weights are fixed.

We use a K-means clustering algorithm to estimate the width and centers. The choice of K significantly influences the quality of training process. Many initializations of K are compared to meet the best performance.

Once K centers have been estimated, the width is calculated as

$$\sigma = \frac{d}{\sqrt{2K}} \tag{8}$$

where d is the Euclidean distance between the center and the nearest center.

The second phase is to fix the weight between the hidden nodes and the output nodes to minimize the errors. The weight is calculated as

$$\begin{aligned} \omega_{ij}(t+1) &= \alpha * w_{ij}(t) + \Delta w_{ij}(t) \\ \Delta w_{ij}(t) &= \eta\left(Y_j - Y_j'\right)\exp\left[-\frac{\left\|x - x_c\right\|^2}{2\sigma^2}\right] + \alpha\Delta w_{ij}(t-1) \end{aligned} \tag{9}$$

where α is the impulse, and η is the learning parameter.

7. Detection Results

With the help of WEKA [36], an open project for machine learning and data mining, we trained an RBF classifier to detect the spam posts. The training set is used to study the diffusion patterns, and the test set is used to validate the effectiveness of our approach.

A sample of the test set is shown in **Table 3**. The column "post ID" is used to distinguish each post with a unique string, while "Tag" is used to label whether a post is normal (0) or spam (1). From the table we can find great differences between spam posts and normal ones, especially in the columns of "CoC", "CoR", "FT", and "ADU".

Table 3. A sample of the test set

Post ID	Tag	Comments					Reposts				
		CoC	FT	AI	VI	ADU	CoR	FT	AI	VI	ADU
00kQpW	0	0.72	3	163	8.2E+05	1133	0.62	0	321	4.4E+07	305
0bpnrD	0	0.84	2	154	1.3E+06	854	0.87	0	76	1.1E+06	69
0YmoZL	0	0.83	2	174	1.1E+06	578	0.71	1	150	1.1E+07	146
11ubGC	0	0.64	0	11	4.2E+03	10	0.53	1	470	6.3E+07	425
17ABVJ	0	0.87	0	107	2.5E+06	2079	0.72	1	194	4.8E+06	189
1A7JNS	0	0.27	1	300	1.3E+06	1896	0.10	0	165	1.2E+07	160
1WvArB	0	0.32	62	217	4.5E+06	337	0.17	0	298	7.6E+07	284
1xmYIG	0	0.53	1	287	3.4E+07	2085	0.55	1	1125	2.9E+07	1082

1Xu1CU	0	0.74	1	42	5.4E+04	122	0.69	0	75	2.5E+06	72
u62GfZ	0	0.81	2	173	4.0E+06	1465	0.67	1	307	7.6E+06	293
ThEj28	1	0.21	355	12	1.3E+05	14	0.32	355	28	1.2E+06	28
6zq1rR	1	0.04	7	53	1.4E+06	48	0.01	16	1	3.9E+03	2
EjhDGu	1	0	10	7	3.1E+04	8	0	14	1	1.4E+03	1
lRDswf	1	0	814	512	1.0E+08	511	0	814	23	7.1E+03	23
pJf8ZV	1	0.02	129	193	4.0E+07	191	0.02	129	41	3.1E+06	40
Nb4Vdt	1	0	14	29	1.0E+05	29	0	14	18	4.8E+03	18
GaplNv	1	0.03	34	26	9.1E+04	26	0.02	39	17	4.3E+03	17
HLFIci	1	0.56	30	52	4.7E+05	53	0.44	31	130	2.9E+07	127
AQ1JkS	1	0.12	102	27	4.7E+05	27	0.21	102	17	2.0E+05	17
IZBXjL	1	0	81	169	3.5E+07	167	0	81	45	2.6E+06	44

The performance of our classifier is judged with well-known metrics, precision, recall and F-measure, which are defined as follows.

$$precision = \frac{True\ Positives}{True\ Positives + False\ Positives}$$

$$recall = \frac{True\ Positives}{True\ Positives + False\ Negatives}$$

$$F - \text{score} = 2 * \frac{precision * recall}{precision + recall}$$

Except RBF methods, there're also some other well-known classification algorithms, such as Bayes, SVM and Logistic. We also conduct these three algorithms as baselines for comparison.

7.1. Performance of Relation-Base Phase

There're two features defined in the relation-base phase, *cor* and *coc*. The importance distributions are outlined in **Fig. 9**. It shows that the two features play comparable roles.

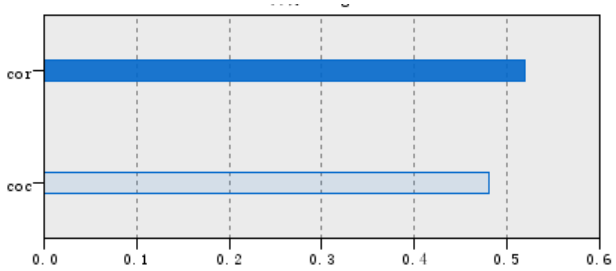

Fig. 9. Importance of relation-based features

The performance is given in **Table 4**. It shows that the RBF method performs the best in the

detection of non-spam posts, while the Bayes method performs the best in the detection of spam posts. By comparing the F-scores of the two methods, we can find that the RBF method provides higher F-score in non-spam detection and comparable F-score in spam detection, which indicates that it relatively provides better performance.

Although there're only two features used, the detection results of relation-based phase are still inspiring. Most of the metrics are higher than 80%.

Table 4. results of relation-based phase

	Non-spam posts			Spam posts		
	Precision	**Recall**	**F-score**	**Precision**	**Recall**	**F-score**
RBF	86.51%	76.22%	81.04%	79.89%	88.82%	84.12%
Bayes	76.35%	79.02%	77.66%	82.04%	90.13%	85.90%
SVM	81.29%	79.02%	80.14%	80.77%	82.89%	81.82%
Logistic	85.37%	73.43%	78.95%	77.91%	88.16%	82.72%

7.2. Performance of Interaction-Base Phase

In the interaction-based phase we defined four features. Each of them is calculated as the repost feature and the comment feature. **Fig. 10** depicts the importance of each feature. It shows that the features don't deviate too much in importance of detection. The most important features are the *ADu* and *AI*, both from the reposts. From the distributions we can conclude that the repost features are more powerful than the comment features in spam detection.

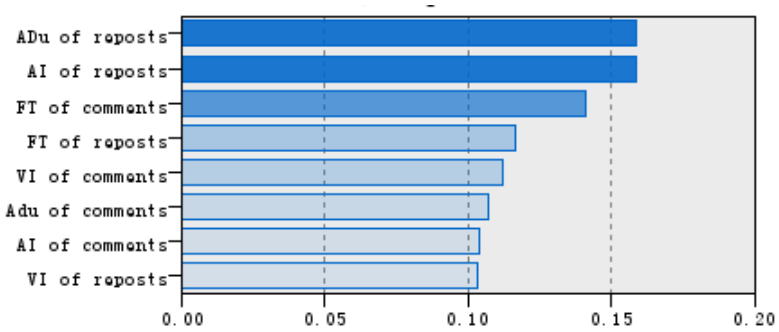

Fig. 10. Importance of interaction-based features

The detection results of interaction-based phase appear in **Table 5**. We notice that the RBF method has distinct advantages than the others. The F-scores of both non-spam posts detection and spam posts detection are as high as 95.77% and 96.39%.

Table 5. Results of relation-based phase

	Non-spam posts			Spam posts		
	Precision	**Recall**	**F-score**	**Precision**	**Recall**	**F-score**
RBF	96.45%	95.10%	95.77%	95.81%	96.97%	96.39%
Bayes	100.0%	9.09%	16.67%	55.93%	100.0%	71.74%
SVM	92.78%	62.94%	75.0%	74.88%	95.76%	84.04%
Logistic	89.74%	73.43%	80.77%	80.10%	92.73%	85.96%

From the experiment we can see that our detection method can provide inspiring results. It also proves the effectiveness of our propagation model. It's worth to say that we launch the relation-based and the interaction-based detection methods separately. Combining them together may provide higher precisions and recalls, but we're actually not sure whether the relation-based method is as robust as the interaction-based method or not. Theoretically the spammers can first follow the original posters and then unfollow them after the spam campaigns, in this way the relation-based method is conquered. However, so far we haven't seen such situations. So we use the relation-based method as a supplement to the interaction based method.

Table 6. Performance comparison

	Precision	**recall**	**F-score**
Our method	0.958	0.970	0.964
Han's method [28]	0.90	0.72	0.80

Due to the differences in spam type and detection environment, it's difficult to compare the performance of different detection methods. Even the same detection tool may perform variously in different networks. Han's work [28] is also conducted with dataset of Weibo. So we choose it as a baseline to examine the performance of our model, which is shown in **Table 6**. From the table we can see that our method outperform Han's work in every field of precision, recall and F-score.

8. Conclusion

With the rapid progress of OSNs, spam has become one of the top threats. Although many detection tools have been introduced, they're either designed for some specific types of spam, or are easy to be avoided. Few can satisfy the demands of accuracy, robustness and comprehensiveness.

In this paper, we propose a model to evaluate information diffusion in micro-blogging networks. Relation-based and interaction-based features are analyzed and trained with an RBF-based approach. Under these features, we could conduct spam detection with high accuracy.

Our model is built on the basis of information diffusion. The diffusion patterns won't be affected by any single user's behaviors. So our model is robust in dealing with users' behavior changes. Besides, despite the differences in contents and types, spams are organized and propagated in the same way. Thus, our method is more comprehensive and can theoretically be

used to deal with all kinds of spams.

There're still a few remaining issues. A combination of the relation-based and the interaction-based features would be useful in improve the quality of the detection results. In this paper, we don't consider the text-based features; they're worth to be analyzed in the future works.

References

[1] W. Peng and M. Uehara, "Multiple Filters of Spam Using Sobel Operators and OCR," in *Complex, Intelligent and Software Intensive Systems (CISIS), in Proc. of 2012 Sixth International Conference on*, 2012, pp. 164-169. Article (CrossRef Link).

[2] R. Bhattacharjee and A. Goel, "Avoiding ballot stuffing in eBay-like reputation systems," in *Proc. of the 2005 ACM SIGCOMM workshop on Economics of peer-to-peer systems*, Philadelphia, Pennsylvania, USA, 2005. Article (CrossRef Link).

[3] H. Tran, *et al.*, "Spam detection in online classified advertisements," in *Proc. of the 2011 Joint WICOW/AIRWeb Workshop on Web Quality*, Hyderabad, India, 2011. Article (CrossRef Link).

[4] D. Liu and X. Chen, "Rumor Propagation in Online Social Networks Like Twitter—A Simulation Study," in *Proc. of presented at the 2011 Third International Conference on Multimedia Information Networking and Security*, 2011. Article (CrossRef Link).

[5] C. Wisniewski. Twitter hack demonstrates the power of weak passwords. http://www.sophos.com/blogs/chetw/g/2010/03/07/twitter-hack-demonstrates-power-weak-passwords/, March 2010. Article (CrossRef Link).

[6] K. Levchenko, *et al.*, "Click Trajectories: End-to-End Analysis of the Spam Value Chain," in *Proc. of Security and Privacy (SP), 2011 IEEE Symposium on*, 2011, pp. 431-446. Article (CrossRef Link).

[7] *Weibo*. Available: http://www.weibo.com. Article (CrossRef Link).

[8] R. B. Cialdini and N. J. Goldstein, "Social influence: Compliance and conformity," *Annu. Rev. Psychol*, vol. 55, pp. 591-621, 2004. Article (CrossRef Link).

[9] K. Thomas, *et al.*, "Design and evaluation of a real-time URL spam filtering service," in *Proc. of 2011 IEEE Symposium on Security and Privacy, SP 2011, May 22, 2011 - May 25, 2011*, Berkeley, CA, United states, 2011, pp. 447-462. Article (CrossRef Link).

[10] L. Ze and S. Haiying, "SOAP: A Social network Aided Personalized and effective spam filter to clean your e-mail box," in *Proc. of 2011 IEEE INFOCOM*, 2011, pp. 1835-1843. Article (CrossRef Link).

[11] S. Khanna, *et al.*, "Inbound & Outbound Email Traffic Analysis and Its SPAM Impact," in *Proc. of 2012 Fourth International Conference on Computational Intelligence, Communication Systems and Networks (CICSyN)*, 2012, pp. 181-186. Article (CrossRef Link).

[12] L. Xiao Mang and K. Ung Mo, "A hierarchical framework for content-based image spam filtering," in *Proc. of Information Science and Digital Content Technology (ICIDT), 2012 8th International Conference on*, 2012, pp. 149-155. Article (CrossRef Link).

[13] K. Thomas, *et al.*, "Suspended accounts in retrospect: an analysis of twitter spam," in *Proc. of the the 2011 ACM SIGCOMM conference on Internet measurement conference*, Berlin, Germany, 2011. Article (CrossRef Link).

[14] C. Grier, *et al.*, "@spam: the underground on 140 characters or less," in *Proc. of the 17th ACM conference on Computer and communications security*, Chicago, Illinois, USA, 2010. Article (CrossRef Link).

[15] H. Gao, *et al.*, "Detecting and characterizing social spam campaigns," in *Proc. of the 10th ACM SIGCOMM conference on Internet measurement*, Melbourne, Australia, 2010. Article (CrossRef Link).

[16] D. Irani, *et al.*, "Study of Static Classification of Social Spam Profiles in MySpace," in *Proc. of presented at the 4th Int'l AAAI Conference on Weblogs and Social Media*, 2010.

Article (CrossRef Link).
[17] Y.-M. Wang, *et al.*, "Spam double-funnel: connecting web spammers with advertisers," in *Proc. of the Proceedings of the 16th international conference on World Wide Web*, Banff, Alberta, Canada, 2007. Article (CrossRef Link).
[18] Y. Shin, *et al.*, "Prevalence and mitigation of forum spamming," in *Proc. of IEEE INFOCOM 2011, April 10, 2011 - April 15, 2011*, Shanghai, China, 2011, pp. 2309-2317. Article (CrossRef Link).
[19] F. Benevenuto, *et al.*, "Identifying video spammers in online social networks," in *Proc. of the 4th international workshop on Adversarial information retrieval on the web*, Beijing, China, 2008. Article (CrossRef Link).
[20] A. Rajadesingan and A. Mahendran, "Comment spam classification in blogs through comment analysis and comment-blog post relationships," in *Proc. of the 13th international conference on Computational Linguistics and Intelligent Text Processing - Volume Part II*, New Delhi, India, 2012. Article (CrossRef Link).
[21] Z. Zhang and B. Varadarajan, "Utility scoring of product reviews," in *Proc. of ACM Conference on Information and Knowleged Management(CIKM2006)*, Arlington, VA, USA, 2006. Article (CrossRef Link).
[22] N. Jindal and B. Liu, "Opinion spam and analysis," in *Proc. of the international conference on Web search and web data mining*, Palo Alto, California, USA, 2008. Article (CrossRef Link).
[23] Zhang, Qunyan, et al. "Duplicate Detection for Identifying Social Spam in Microblogs." In *Big Data (BigData Congress), 2013 IEEE International Congress*. IEEE, 2013. Article (CrossRef Link).
[24] Wang, Kaiyu, et al. "A new approach for detecting spam microblogs based on text and user's social network features," *Proc. of Wireless Communications, Vehicular Technology, Information Theory and Aerospace & Electronic Systems (VITAE), 2014 4th International Conference on. IEEE*, 2014. Article (CrossRef Link).
[25] X. Zhang, *et al.*, "Detecting spam and promoting campaigns in the Twitter social network," in *Proc. of 12th IEEE International Conference on Data Mining, ICDM 2012, December 10, 2012 - December 13, 2012*, Brussels, Belgium, 2012, pp. 1194-1199. Article (CrossRef Link).
[26] Y. Liu, *et al.*, "Identifying web spam with user behavior analysis," presented at *the Proceedings of the 4th international workshop on Adversarial information retrieval on the web*, Beijing, China, 2008. Article (CrossRef Link).
[27] Y. Liu, *et al.*, "User behavior oriented web spam detection," in *Proc. of 17th International Conference on World Wide Web 2008, WWW'08, April 21, 2008 - April 25, 2008*, Beijing, China, 2008, pp. 1039-1040. Article (CrossRef Link).
[28] H.Zhongmin, et al., "Probabilistic Graphical Model for Identifying Water Army in Microblogging System," *Journal of Computer Research and Development*, 2013, S2:180-186. Article (CrossRef Link).
[29] D. Strang, *et al.*, "Soule. Diffusion in organizations and social movements: From hybrid corn to poison pills," in *Annual Review of Sociology*, 1998,24(1):265–290. Article (CrossRef Link).
[30] Goldenberg J, *et al.*, "Talk of the network: A complex systems look at the underlying process of word-of-mouth," *Marketing Letters*, 2001, 12(3):211–223. Article (CrossRef Link).
[31] M. Kimura and K. Saito. "Tractable Models for Information Diffusion in Social Networks," *Knowledge Discovery in Databases: PKDD*, Kimura, Masahiro, 2006. Article (CrossRef Link).
[32] G. Daniel, et al. "Information diffusion through blogspace," in *Proc. of the 13th international conference on World Wide Web*, ACM, 2004. Article (CrossRef Link).
[33] Iribarren, *et al*. "Impact of human activity patterns on the dynamics of information diffusion," *Physical review letters*, 2009. Article (CrossRef Link).
[34] J. Yang and J. Leskovec, "Modeling Information Diffusion in Implicit Networks," in *Proc. of Data Mining (ICDM), 2010 IEEE 10th International Conference*, pp.599,608, 13-17 Dec. 2010.

Article (CrossRef Link).
[35] S. Haykin, *Neural Networks: A Comprehensive Foundation*. NJ: Predice Hall, Upper Saddle River, 1994. Article (CrossRef Link).
[36] M. Hall, *et al*. "The WEKA Data Mining Software: An Update, " in *SIGKDD Explorations*, vol. 11, Issue 1, 2009. Article (CrossRef Link).

Kan Chen is a PhD student in School of Computer, National University of Defense Technology (NUDT)。He **r**eceived his B.S. degree in Network Engieering and M.S. degree in Computer Science in 2007 and 2010 respectively. His research interests include social network and network security

Peidong Zhu is a professor with School of Computer Science of National University of Defense Technology (NUDT), China. He received his PhD degree in computer science from NUDT in 1999. His research interests include network routing, network security and architecture design of the Internet and various wireless networks. He is a member of the IEEE.

Liang Chen is a PhD student in School of Computer, National University of Defense Technology (NUDT). His research interests include social network and network security.

Yueshan Xiong is a professor with School of Computer Science of National University of Defense Technology (NUDT), China. His research interests include Computer Graphics, Computer Simulation and image processing.