Fighting Against Social Spammers on Twitter by Using Active Honeypots

Hangcheng Zhu

Department of Electrical & Computer Engineering
McGill University
Montreal, Canada

May 2014

2014/08/03

Abstract

With the popularity of social network in recent years, the spam problem of this web application becomes more and more serious. Many methods have been proposed to fight against spam problem on social networks. In this thesis, we first review the motivation and basis of spamming activities on social networks as well as discuss the previous spam detection strategies on social networks. Then we put forward a new spam detection approach to fight against spammers on Twitter. We find that there is a certain type of accounts on Twitter, termed as active honeypots, which are 8,000 times more efficient in trapping social spammers than manually created honeypot accounts used in previous works. Active honeypots are existing accounts on Twitter. Among the accounts interacting with active honeypots everyday, there are a large portion of social spammers. To understand why these active honeypots are so attractive to social spammers, we conduct in-depth investigation to reveal what properties and why these properties are attractive to social spammers. We also create accounts to imitate the behaviors of active honeypots to further learn about the attractiveness of each behavior. Based on these investigations, we design an active honeypot based spammer detection system, which can identify effective active honeypots and detect social spammers interacting with these active honeypots. Especially, we propose a new kind of features named active honeypot based features to improve the performance of our system and conduct a comparative study with previous work. Final evaluation results on data crawled from Twitter demonstrate that our proposed system can achieve a false positive rate of 0.019. With 1,819 active honeypots we can trap about 40,000 social spammers on Twitter every day which are about 4% of the daily new registered Twitter users.

Abrégé

Avec la popularité des réseaux sociaux au cours des dernières années, le problème du spam de ces applications devient de plus en plus grave. De nombreuses méthodes ont été proposées pour lutter contre problème du spam sur les réseaux sociaux. Dans cette thèse, nous examinons d'abord la motivation et la base des activités spam sur les réseaux sociaux ainsi que discutons des stratégies existantes de détection de spam sur les réseaux sociaux. Ensuite, nous avons mis en avant une nouvelle approche de détection de spam pour lutter contre les spammeurs sur Twitter. Nous constatons qu'il ya un certain type de comptes sur Twitter, nommé comme le pot de miel actif, qui est 8000 fois plus efficace pour piéger les spammeurs sociaux que les pots de miel créées manuellement et utilisés dans des travaux précédents. Pots de miel actifs sont comptes existants sur Twitter. Parmi ces comptes interagir avec les pots de miel actifs tous les jours, il ya une grande partie des spammeurs sociaux. Pour comprendre pourquoi ces pots de miel actifs sont si attrayants pour les spammeurs sociaux, nous menons une enquête approfondie pour révéler quelles propriétés et pourquoi ces propriétés sont attrayantes aux spammeurs sociaux. Nous créons aussi des comptes pour imiter les comportements des pots de miel actifs afin d'apprendre davantage sur l'attractivité de chaque comportement. Basé sur ces enquêtes, nous concevons un système de détection de spammeur la base des pots de miel actifs, qui permet d'identifier les pots de miel actifs efficaces et de détecter les spammeurs sociaux qui interagissent avec ces pots de miel actifs. Notamment, nous proposons un nouveau type de fonction sappuyant sur les pots de miel actif pour améliorer la performance de notre système et pour mener une étude comparative avec des travaux antérieurs. Des résultats de l'évaluation finale sur les données de Twitter démontrent que notre système proposé peut atteindre un taux de faux positif de 0,019. Avec 1819 pots de miel actifs, nous pouvons piéger environ de 50.000 spammeurs sociaux sur Twitter chaque jour qui sont environ 5% des nouveaux utilisateurs de Twitter enregistrés quotidiennement.

Acknowledgements

Foremost, I would like to express my sincere gratitude to my supervisor Prof. Haibo Zeng for his continuous support of my master study both in finance and academy. My research topic is not social network security at the beginning of my master study. I met some difficulties at the end of my first term and had to change my research topic. Prof. Zeng gave me enough freedom to choose any research topic I was interested in at that time. Finally, I chose social network security as my new research topic. Prof. Zeng supported my decision and gave me as much help as possible on my research. No matter how busy he was, Prof. Zeng kept meeting with me every week to help me with my research. I still remember the night before I submitted my paper to ACSAC conference. Prof. Zeng was travelling abroad at that time. But he still discussed with us about the paper through Skype and helped to revise my paper until late at night.

I would like to express my gratitude to my senior and research mate Hanqiang Cheng. Hanqiang is just like my brother. He is older than me and started research in social network security earlier than me. He is always so selfless and patient. The first time I met Hanqiang, he told me a lot about the research on social network security for about one hour. When I began my study in this area, I was not familiar with Twitter API and Python programming language. Hanqiang not only gave me lots of guidance but also wrote some sample programs to help me understand. Though Hanqiang was busy in his last year of Ph.D. study, he insisted on joining my weekly meeting with Prof. Zeng and gave me many constructive suggestions in the meeting.

I would express my appreciation to my colleagues in the office — Chuansheng Dong, Di Wu and Chen Jiang. Though our research fields are different, they were always glad to help me when I met difficulties, no matter in research or in life. Our conversation in lab brought me lots of fun and left me such precious memories.

My sincere thanks also goes to my room mate Peicheng Liao. Peicheng is my most important friend during my two-year study in McGill University. He was always so kind and helpful. When I was busy with my research, he would cook for me, when I felt deeply depressed he always gave me so much comfort and support. He will soon go to United State and pursue his Ph.D degree in University of South California. I wish him a successful Ph.D career there.

Last but not the least, I would like to thank my parents. They are my strong backing

both in finance and spirit. Without their financial support I cannot finish my study abroad, their unconditional love and support is my faith to overcome any difficulties in my life.

Contents

List of Figures

List of Tables

List of Acronyms

OSNs	Online social networks (OSNs)
FB-ratio	Follow back ratio
TF-IDF	Term frequency-inverse document frequency
SVR	Support Vector Regression
FP	False Positive
TP	True Positive
ROC	Receiver Operating Characteristic

Chapter 1

Introduction

1.1 Background

No web-based application can attract more users and become more popular than online social networks (OSNs) in recent years. By the end of September 2012, Facebook, the world's largest social network, was reported having over one billion users. And Twitter, the world's largest micro blogging service, claimed 500 million users. The overwhelming popularity of OSNs brings a revolution to communication among people and the spreading of news. It becomes much easier for people to keep in touch with friends and share valuable news across the whole Internet. Unfortunately, the spammers which has been a long existing problem in search engine and email system, is also becoming a serious problem for OSNs. According to a research [3] in 2011, 80 million among 1.8 billion randomly crawled Twitter accounts are social spammers. Even worse, the click through rate of spam links on OSNs is of magnitude higher than its email counterpart. This is because people are more willing to trust the spam messages from their friends on OSNs. To solve this problem, a lot of spam detection strategies were proposed. Honeypot is one of the common used detection approaches.

Honeypot is a widely used technique in network security and email spam detection [4–6]. Compared to other techniques, honeypot can provide early warning about potential new attacks and allow in-depth examination of the behaviors of adversaries [4]. Recently, honeypots are also used to fight against the rampant spamming activities on popular OSNs such as Facebook and Twitter [7–9]. In these works, fake accounts were manually created to trap social spammers on OSNs. Though these systems can detect various social

spammers pretty accurately, they are inefficient in trapping social spammers, which make them insufficient for fighting against social spammers. For instance, according to [9], 300 manually created honeypots on Twitter can only trap 361 social spammers in one month. One primary reason for the low efficiency of these systems is the lack of knowledge about what properties of honeypots can attract social spammers.

In this thesis, we come up with a brand new kind of honeypots − active honeypots. Active honeypots are certain kind of existing accounts on Twitter which are extremly attractive to social spammers. There are lots of social spammers in the daily new followers, new friends or tweets mentioned users of active honeypots. Active honeypots are proven to be $8,000$ times more efficient than traditional honeypots in trapping social spammers. Based on active honeypots we build a spammer detection system which is able to detect 50,000 social spammers on Twitter every day. If not specified, all the spammers in the left part of the thesis refer to social spammers on OSNs.

1.2 Contribution

In this thesis, we focus on analysing the properties and mechanism of attractiveness of active honeypots. Based on this, we design an active honeypot based spammer detection system which first identifies active honeypots from Twitter, then detects spammers with the help of these active honeypots.

In the analysis part, unlike several recent works [3, 10–12] which focused on analysing the properties of spammers, we study the properties of accounts which are attractive to spammers. We try to reveal the potential reasons why these properties are attractive. To achieve this goal, we observe the daily behaviors of 4 million accounts randomly sampled from Twitter's public stream for a few days, such as changes in friends (*i.e.*, accounts being followed by these randomly sampled accounts), followers (*i.e.*, accounts following these accounts), and tweets (*i.e.*, messages posted by these accounts). Among the 4 million accounts, we identify 1,841 accounts which are extremely attractive to spammers. Each account can attract at least 10 spammers every day in average. And for each of these 1,841 accounts, there are at least 20% spammers among all the new accounts interacting with them every day. We believe that these 1841 accounts can serve as honeypots which can efficiently trap spammers. As opposed to the honeypots proposed in existing works [7–9], we refer to these 1841 attractive accounts as *active honeypots* and the honeypots in previous

works as *passive honeypots*. Compared to the passive honeypots, active honeypots are much more efficient in trapping spammers. In addition, no manual effort is needed for creating and maintaining active honeypots when using them to trap spammers.

In order to understand why active honeypots are so attractive to spammers, we conduct comparative studies between these 1,841 active honeypots and 100,000 random influential accounts for an extra half year. Our studies reveal that the following five types of potential rewards offered by active honeypots make them attractive to spammers: (i) some active honeypots follow back spammers which follow them; (ii) some active honeypots mention spammers in their tweets; (iii) some active honeypots offer retweeting service for spammers; (iv) some active honeypots post certain sensitive keywords in tweets to attract spammers; (v) some active honeypots buy followers from spam campaigns. To further prove the attractiveness of these behaviors, we conduct an experiment in which we create some accounts to imitate the behaviors of active honeypots and check whether they can become attractive to spammers. The results show that not all these behaviors can make our accounts attractive to spammers. Some other factors may also be necessary for active honeypots being attractive to spammers.

In the system design part, we design an active honeypot based spammer detection system based on our analysis of active honeypots. At first, by using only $10,000$ accounts suspended by Twitter as input seeding set, our system can successfully identify efficient active honeypots among all the accounts interacting with the seeding set. Then we extract a kind of active honeypot based features to build an enhanced feature based spammer detector. This new kind of features is proven to be effective in improving the performance of spammer detector. At last we evaluate the overall performance of our system. Our system can detect about 40,000 spammers suspended by Twitter each day with high precision, which is about 4% of the new registered Twitter users according to [13].

As a summary, we claim the following contributions for this thesis:

1 We propose to use active honeypots for trapping spammers. It is the most efficient approach currently known to trap a large number of spammers. In addition, no additional cost is needed for creating and maintaining active honeypots.

2 We conduct comparative studies to expose why active honeypots are attractive to spammers.

3 We create some accounts to imitate the behaviors of active honeypots to further learn about the mechanism behind attractiveness of active honeypots.

4 We propose a new kind of active honeypot based features to build an enhanced feature based spammer detector and conduct a comparative study with corresponding previous research.

5 We design an active honeypot based spammer detection system, which is effective in trapping spammers.

1.3 Thesis Organization

This section outlines the organization of the thesis.

Chapter 1

In chapter 1, we first discuss the motivation and basis of spamming activities on web and OSNs. The ultimate goal of spamming activities is earning profit including commercial interest and political interest. The underground account market, which offers millions of fake accounts, serves as the basis of spamming activities on OSNs. Then we review the strategies to fight against spammers proposed by researchers. The research of social network spam detection starts from 2007 and our review covers most of the main research works in this area from the beginning. We divide the detection strategies into six categories including: feature based strategy, ranking based strategy, blacklist based strategy, honeypot based strategy, clustered based strategy and some other strategies.

Chapter 2

In chapter 2, we first demonstrate the existence of active honeypots and our method in collecting active honeypots from Twitter. Then we discuss our observation on several interesting properties of active honeypots. In addition, we will analyse why active honeypots are attractive to spammers by observing the behaviors of active honeypots as well as analysing the potential reward mechanism for spammers. At last we build some accounts to imitate the potential attractive behaviors. By imitating these behaviors we further learn about the mechanism behind the attractiveness of active honeypots.

Chapter 3

In chapter 3, we first introduce the system overview of our active honeypot based spammer detection system. Then we give an detailed presentation about the design of active honeypot identifier which is used to identify effective active honeypots from billions of Twitter accounts. The identifier is composed of three stages of ranking: graph based ranking, feature based ranking and history based ranking. At last we evaluate the performance of the active honeypot identifier.

Chapter 4

In chapter 4, we introduce a new kind of features named active honeypot based features. We build an enhanced spammer detector with this new kind of features and conduct comparative studies with previous feature based spammer detector. We also discuss the feature discrimination power as well as the tuning of threshold under an unbalanced dataset. Then we give out an overall evaluation of our active honeypot based spammer detection system.

Chapter 5

Chapter 5 concludes this thesis.

1.4 Notations

In this thesis, we use the following notation conventions.

A_{apa}	Number of accounts attracted by per action
A_{spa}	Number of spammers attracted by per action
$M_{accounts}$	Number of accounts attracted by a group
$M_{spammers}$	Number of spammers attracted by a group
N_{action}	Total number of actions of a group
$spam\_num$	Number of spammers trapped by an account
$spam\_ratio$	Ratio of spammers among accounts trapped by an account
β	A congurable parameter controlling the trade-off between $spam\_num$ and $spam\_ratio$
G_{fr}	Friend graph of graph based ranking

G_{fo}	Follower graph of graph based ranking
G_m	Mention graph of graph based ranking
c_{fr}	Ranking score in friend graph
c_{fo}	Ranking score in follower graph
c_m	Ranking score in mention graph
c	Ranking score in joint graph
α	Damping factor in TrustRank
c_f	Ranking score after feature based ranking
c_h	Historical attractive score
γ	A configurable parameter controlling the trade-off between c_f and c_h
C_{tw}	The total number of tweets posted by an account
M	Number of mentioned users in tweets of an account
H	Number of hashtags in tweets of an account
U	Number of URLs in tweets of an account
t_i	Posted time of the i-th tweet of an account
R_i	Retweet count of the i-th tweet of an account
F_i	Favorite count of the i-th tweet of an account
P	The set of possible tweet-to-tweet combinations among any two tweets posted by an account
p	A single pair of tweets posted by an account
$c(p)$	A function calculating the number of words two tweets share
l_a	The average length of tweets posted by an account
l_p	The number of tweet combinations

Chapter 2

Literature Review

In this chapter, we first discuss the motivation of spamming activities on OSNs. The ultimate goal of spamming activities is earning profit including commercial profit and political profit. Then we specially discussed the underground account market on OSNs which serves as the basis for most spamming activities. After discussing the motivation and basis of spamming activities, we review the strategies to fight against social network spammers. We will discuss six categories of spam detection strategies including: feature based strategy, ranking based strategy, blacklist based strategy, honeypot based strategy, cluster based strategy and some other strategies.

2.1 Spamming Activities on Social Network

2.1.1 The Motivation of Spamming Activities on OSNs

Before we start our discussion of spamming activities on OSNs, we first need to figure out the motivation behind the spamming activities. The answer should be straightforward, profit lies at the heart of the spamming activities. Approaches to pursing profit of spam vary from promoting the sales of products to stealing information from legitimate users. In table 2.1 [2], we present the breakdown of spam categories for spamming activities on Twitter, based on tweet text. We can see that the spam content on Twitter is related to various fields in our daily life and mainly focus on commercial profit. For example, free music, jewellery, gambling, prizes and loans in table 2.1 are obviously related with commercial profit.

Table 2.1 Breakdown of spam categories for spamming activities on Twitter [2]

Category	**Fraction of Spam**
Free music, games, books, downloads	29.82%
Jewelery, electronics, vehicles	22.22%
Contest, gambling, prizes	15.72%
Finance, loans, realty	13.07%
Increase Twitter following	11.18%
Diet	3.10%
Adult	2.83%
Charity, donation scams	1.65%
Pharmacutical	0.27%
Antivirus	0.14%

Thomas [2] presented an overview of the approaches to earn commercial profit for spammers and estimated how much revenue they earned. He divided the techniques used by spammers to earn commercial profit into five categories including: (i) spamvertized goods, (ii) fake software, (iii) clickfraud, (iv) banking theft, (v) and commoditizing compromised hosts. With these techniques spammers can (i) promote the sale of a product, (ii) cheat users' money for non-existent software function, (iii) attract clicks on pay-per-click advertisements, (iv) steal personal banking information, (v) and compromise an user's account and sell it for other spamming activities. They estimated that miscreants can earn a revenue of 12∼92 million dollars with spam affiliate programs and 5∼116 million dollars with anti-virus scammers. In addition to traditional profit motivated spamming activities on OSNs, Thomas *et al.* [3] found there existed another evolutionary form of spamming activity on Twitter named as spam-as-a-service. Spam-as-a-service includes affiliate programs, ad-based shortening services and account sellers. This kind of service acts as a supporter to spammers on Twitter. It allows spammers to specialize their efforts, decoupling the process of distributing spam, registering domains and hosting content, and if necessary, product fulfillment.

Besides commercial profit, social networks have emerged as a significant tool to earn political profit in both political discussion and dissent. Thomas *et al.* [14] undertook an in-depth analysis of the infrastructure and accounts that facilitated this kind of censorship-based attack. The attackers leveraged the spam-as-a-service market to acquire thousands

of fraudulent accounts which they used together with compromised hosts to manipulate political speech.

Since profit is the ultimate goal of spammers, it is meaningful to measure the amount of profit produced by a spamming activity. Kanich *et al.* [15] introduced the "conversion rate" of spam to measure the probability that an unsolicited email will ultimately elicit a "sale". The author made use of an existing spamming botnet and convinced it to modify a subset of the spam it already sent, thereby directing any interested recipients to servers under their control. So that they could record the number of sales and calculate the "conversion rate". Together with the cost to send spam and the marginal profit per sale they could finally get the profit produced by an spamming activity. According to their result, the spam campaigns would produce roughly 3.5 million dollars of revenue in a year and about a annual net revenue of 1.75 million dollars. This number could be even larger if spam-advertised pharmacies experience repeat business. Though their work was based on email system, we can use the similar method to measure the "conversion rate" for OSNs.

2.1.2 The Basis of Spamming Activities on OSNs

For whatever kind of spamming activities on OSNs, a large number of spamming accounts are needed to spread the spam content. So the underground account market is the basis of spamming activities on OSNs. In the underground account market, fraudulent accounts − automatically generated by bot used to spread scams, phishing and malware − are sold in bulk. In order to deter spamming activities, we need to learn about this underground account market and try to prevent it from offering millions of fraudulent accounts to carry out spamming activities.

Thomas *et al.* [16] investigated the underground account market of Twitter and presented the general situation of this market. They studied how the market operated, the impact of the market on Twitter spam levels, and how account merchants avoided the registration barriers. Their result reveals that merchants thoroughly understand Twitter's existing barriers against automated registration, so there are always thousands of accounts available and the price is stable. They estimated that $10\% - 20\%$ of spamming accounts on Twitter originated from the merchants they identified as well as the merchants could generate about a total revenue between \$127,000 to \$459,000 from the sale of accounts. To deter the underground market the author developed a classifier which could retroactively

detect fraudulent accounts. Besides, the author also investigated the failures of existing defense against automated registration and provided a set of recommendations to increase the cost of generating fraudulent accounts.

Many accounts bought from underground account market are used as fake followers. According to a recent report [1], fake followers are becoming a general problem on Twitter. Fig. 2.1 [1] shows the ratio of fake accounts among followers for top 10 Twitter accounts. Almost 50% of Justin Bieber's fans are fake accounts. Stringhini *et al.* [17] conducted a research which focused on this Twitter follower market. "Twitter follower market" sells followers to consumers. According to their investigation, some merchants used fake accounts to boost the follower number of their customers, while others relied on a pyramid scheme to let non-paying customers follow each other or follow paying customers. The author developed a detection system which could detect the customers of follower markets based on follower dynamics.

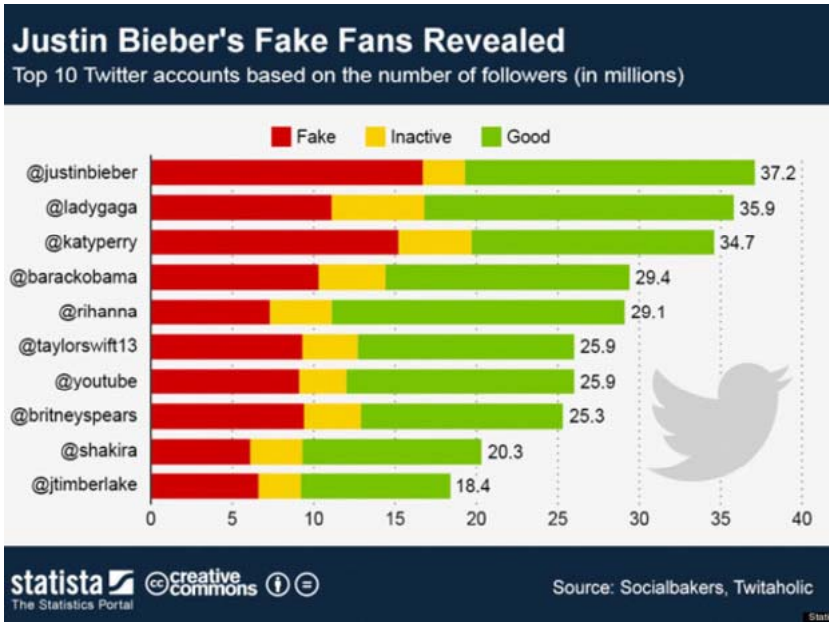

Fig. 2.1 Follower composition of top 10 accounts on Twitter [1]

Besides fake accounts, the underground account market offers compromised accounts for spamming activities. Fake accounts usually exhibit highly anomalous behaviors, consequently they are easy to detect. So attackers have started to compromise and abuse legitimate accounts to spread spam content. Egele *et al.* [18] investigated the compromised

accounts on Twitter, they found that compromised accounts were more effective in spreading spam content than fake accounts since attackers could leverage the trust relationships that the account owners had established in the past. What's worse, compromised accounts were more difficult to clean up than fake accounts because the real owners were legitimate users and the service provider could not simply delete these accounts. The author designed a novel approach to detect compromised accounts. They adopted a composition of statistical modeling and anomaly detection to identify accounts that experiencing a sudden change in behavior. In order to distinguish between malicious and legitimate changes the author checked if a group of accounts that all experiencing similar changes within a short period of time. This kind of changes can be the result of a malicious campaign that is unfolding.

2.2 Spam Detection Strategies

In this section we will discuss the spam detection strategies on OSNs. Spam detection has been studied for a long time in email system. Lots of efficient strategies have been proposed to fight against email spam and Google had claimed [19] that their anti-spam system could lower the spam rate in Gmail service to 1%. On OSNs, however, the anti-spam studies started just a few years ago and few researchers can claim a performance as good as email system. This is because OSNs are more open than email system. Spammers can hit targeted audiences more easily and precisely. Besides, people are more willing to believe related people on OSNs than an unsolicited sender in email system. According to a recent report [20], the main battle field of anti-spam has changed from email system to OSNs. In 2007, Heymann *et al.* [21] first made a survey of approaches and future challenges about fighting against spam on OSNs and proposed some simple anti-spam methods. After that many new strategies have been proposed to solve the challenging spam problem on OSNs.

2.2.1 Feature Based Strategy

The most straightforward strategy to detect spammers on social network is applying machine learning methods for classifying. It first collects and identifies features which can distinguish spammers from legitimate users and then build a binary classifier to separate these two kinds of users. We call this kind of strategy as feature based strategy.

Krause *et al.* [22] first adopted the classical approach in machine learning to detect spammers in a social bookmarking system. Topological, semantic and profile-based features were adopted by them and they also used feature selection to realize better performance.

The specific feature based classifier for Twitter was introduced by Benevenutotrained *et al.* [23]. Some Twitter specific features which based on tweet content and user social behavior were introduced in his work. He claimed a false positive rate of 4.3%, which was still too high for practical spam detection on OSNs.

An enhanced spam detection approach for Twitter was proposed by Moh *et al.* [24]. In Moh's work, the learning process consisted of two steps: at first a classifier was trained to distinguish between spammers and legitimate users on basic user features then they used this trained classifier to generate new features for a user which depend on a user's followers being spammers or legitimate users. This enhanced approach could achieve a precision and recall both of 0.86.

In 2011, more robust features for Twitter spam detection was introduced by Yang *et al.* [25]. They focused on relations between spammers and their neighbors such as a bidirectional link ratio and betweenness centrality. Other features based on timing and automation were also introduced in their papers. Similar to Yang's work Song *et al.* [26] considered the relations between spam senders and receivers such as the shortest paths and minimum cut to extract features.

Rather than classifying spammers and legitimate users, Chu *et al.* [27] extracted features and adopted machine learning methods to classify human, bot and cyborg on Twitter. This is close to spam detection because according to Chu's analysis most legitimate users on Twitter are human while most spammers are controlled by bot and cyborg.

After 2012, there were few works on feature based strategy which simply propose new features and build classifier with these features. This is because feature based strategies have two critical limitations. First, some features, such as account age, friend number and tweets interval time, used in these approaches can be manipulated by spammers. Secondly, these approaches are able to detect spammers only after spammer had already violated Twitter rules because user history data is needed to decide whether a user is a spammer or not.

2.2.2 Ranking Based Strategy

Another anti-spam strategy on social networks is designing ranking algorithms to reduce the prominence of spam contents. Noll *et al.* [28] proposed a graph-based algorithm − SPEAR to rank experts in a collaborative tagging system. The algorithm was based on two factors: the quality of resource identified by a user and the ability to discover useful resource before others. This system was efficient for identifying high-quality resources posters, but some legitimate users on OSNs may not be a contributor of valuable resource and could be ranked as low as spammers.

Later, a Twitter specific ranking algorithm was proposed by Yamaguchi *et al.* [29]. They proposed TURank − an algorithm that measured the Twitter users' authority scores considering both a Twitter social graph and how tweets actually flowed among users. They focused on retweeting and introduced the dynamic user-tweet graph which consisted of nodes, corresponding to user accounts and tweets, and edges, corresponding to following and retweeting relationships.

A problem of social graph based ranking algorithms such as the TURank is that their ranking results may be influenced by spamdexing caused by link farm. For example, an user reciprocally exchanges links with unrelated users to gain influence so that his tweets will be ranked high because of his fake high influence score. Ghosh *et al.* [10] investigated the link farm problem on Twitter and proposed a ranking system, named Collusionrank, to deter link farm on Twitter. The Collusionrank is a Pagerank-like [30] algorithm in which a set of identified spam set is used to penalize users who connect to spammers by lowering the influence scores of these users.

Besides socially connected, spammers usually share similar topic/keywords/URLs to attract victims. Based on this intuition, Yang *et al.* [11] designed a Criminal Account Inference Algorithm (CIA) to infer unknown criminal accounts on Twitter by starting from a seeding set of known criminal ones. In other works, Duan *et al.* [31] and Uysal *et al.* [32] both used machine learning based approaches to rank high quality instance to the top and remove those low quality ones.

Ranking based strategy for preventing spam fully considers the relations among spammers and achieves good performance in many cases. However, there are some problems with this kind of strategy. In some cases, spammers in a spam campaign may not connect with each other, they directly target legitimate users on Twitter, so relation based ranking

cannot achieve good performance. Besides, in all the ranking algorithms, we need to label a set of accounts or content as the seeding set. This labeling process usually needs experts to select seeding accounts manually which can be a time-consuming work.

2.2.3 Blacklist Based Strategy

Blacklist based strategy can be used for spam detection and spam content validation on OSNs. A blacklist is a basic access control mechanism which allows access except for those items (email addresses, users, URLs, *etc.*) on the list. Blacklist strategy has been widely used for anti-spam approaches in email system. In recent years, researchers introduced this method, especially URL blacklist, into spam detection on OSNs. In table 2.2 we list the main URL blacklist service used by researchers.

Table 2.2 URL blacklist service

Service Name	**Target of Blacklist**
Google Safebrowsing	phishing or malware
URIBL	domains present in spam email
Joewein	domains present in spam email
SURBL	domains present in spam email
Spamhaus	domains present in spam email
McAfee SiteAdvisor	malicious sites and malware
SquidGuard	sites for which access is redirected
Wepawet	web pages that launch drive-by-download attacks

URL Blacklist can be used to validate spam content on OSNs. Gao *et al.* [33] used URL blacklist service to validate if a campaign which share a common URL in tweets was a spam campaign. Thomas *et al.* [34] and Lee *et al.* [35] both used URL blacklist service to label spam URLs as ground truth data for train and test set.

In addition to validating spam content, URL blacklist can also be used for spam detection directly. Grier *et al.* [36] examined whether using URL blacklist could help to effectively deter the spread of Twitter spam. According to his result, URL blacklist was too slow in identifying new threats, allowing more than 90% of visitors to view a page before it became blacklisted. They also pointed out that even they tried to reduce blacklist delays, the URL shortening service used by spammers for obfuscation was still a thorny

problem. In order to solve the problems caused by URL shortening service, Wang *et al.* [37] proposed a feasible approach of detecting short URL spam by classification based on the click traffic features. To overcome the time lag drawback of blacklist based strategy, Tan *et al.* [38] put forward a blacklist-assisted runtime spam detection (BARS) system which utilized non-textual features, with the help of an auto-expanding spam blacklist, and a high priority non-spam whitelist to detect spammers.

Most blacklist based approaches used in anti-spam system are URL based blacklist. Unlike these approaches, Ramachandran *et al.* [39] came up with an behavior blacklisting approach to detect spammers. Though his approach was specified for email system, the email-sending patterns conception he raised could be a constructive inspiration for future work on social network blacklist based strategy.

The main drawback of blacklist strategy is its time lag. According to [36], it usually takes 4 to 20 days for a spam URL to be flagged in blacklist. Besides, the coverage of blacklist is low as well as short URL service will weaken the effectiveness of blacklist based strategy.

2.2.4 Honeypot Based Strategy

Honeypot is a trap set specially used for detecting and trapping spam. It has been widely used in email system. In 2008, Webb *et al.* [7] first introduced this idea into social network anti-spam field. They built social honeypots to harvest deceptive spam profiles from social networking communities. They built 51 social honeypots on MySpace and received 1,570 friend requests in a four months period. 97.7% friend requests received by their social honeypots came from spammers. Similar to Webb *et al.*'s work [7], Stringhini *et al.* [9] created a large and diverse set of "honey-profiles" on three large social networking sites to collect the data about spamming activity. On Twitter, they created 300 honey profiles and received 397 friend requests during a period of 11 months among which 361 (90.1%) were from spammers. Lee *et al.* [8] proposed a social honeypots + machine learning approach for spam detection. It does not have much difference with other feature based spam detection strategies except for using social honeypots to harvest deceptive spam profiles in data collection. Actually, all the three above methods used honeypots to detect or trap a small set of spammers. They did not use active honeypots to detect or trap a large number of spammers because of the low efficiency of honeypots in trapping spammers.

The advantages of honeypot based strategy include high spam rate among friend requests as well as being easy to harvest spammers. While this kind of strategy has three main disadvantages: (i) low efficiency in trapping spammers, (ii) time-consuming to create honeypots, (iii) and honeypots themselves are kind of spammers to social networks.

2.2.5 Cluster Based Strategy

Some tricky spammers created spam campaigns as an effective way to spread malwares and phishing attacks. Cluster based strategy can be used to detect spam campaigns effectively. Gao *et al.* [33] first quantified and characterized spam campaigns on social network. He modelled each wall post on Facebook as a node in a graph, and created edges between any two nodes containing the same URL, or any two nodes sharing similar text content as defined by an textual fingerprint. Each connected subgraph extracted from the whole graph was regarded as a campaign. They identify a campaign as a spam campaign with dual behavioral hints of bursty activity and distributed communication.

One problem of above work [33] is simply clustering nodes in a connected subgraph into a campaign. This may cluster two unrelated campaigns into a single one or incorporate unrelated accounts into a campaign. A better approach to cluster accounts into campaigns from graph was proposed by Lee *et al.* [40]. They proposed three graph-based approaches for extracting campaigns including: loose extraction, strict extraction and cohesive extraction. They evaluated a content-driven framework which effectively connect text posts with common "talking points". In addition, they also identify five major types of campaigns including: spam, promotion, template, news, and celebrity campaigns from millions of Twitter messages.

Besides common URL, text similarity and "talking points" we mentioned above, another method to define the similarity between two nodes in a graph was proposed by Zhang *et al.* [41]. They adopted Shannon information theory to measure the similarity between each two accounts purposes of posting URLs. To cluster campaigns, they defined a dense but not fully connected subgraph rather than a connected graph as a campaign.

All the three clustering strategies we discussed above build an off-line graph to cluster campaigns. To realize real-time spam campaign clustering, Gao at al. [42] put forward an online spam filtering system which could inspect messages generated by users in real-time. He adopted incremental clustering and parallelization to detect campaigns with low

overhead and used a set of novel features to effectively distinguish spam campaigns from legitimate campaigns.

Cluster based strategy has many advantages over individual spam detection: First, it can detect spammers before they contact with legitimate users. Secondly some robust features such as the size of a campaign, the bursty of posting contents can be used to identify spammers. However, there are some challenges for cluster based strategy. Everyday Twitter will produce over 200 million tweets, it is almost impossible to realize clustering with so many tweets. Besides, how to define a sub-graph as a campaign properly is also a thorny problem. Since most cluster based approaches compare tweets similarity based on common URL or similar tweet textual content. So cluster based approaches are useless for spam tweets which do not contain URLs or spam tweets which are produced with diverse textual templates.

2.2.6 Other Strategies

Besides the five kinds of spam detection strategies we discussed above, there are some other kinds of spam detection strategies proposed by researchers in recent years.

One of these spam detection strategies is URL based strategy. Lee *et al.* [35] proposed WARNINGBIRD, a suspicious URL detection system for Twitter. He considered correlated redirect chains of URLs contained in a number of tweets and found that attackers had limited resources and thus had to reuse them so part of their URL redirect chains were shared. They focused on these shared resources to detect suspicious URLs rather than investigated the landing pages of individual URL in each tweet which may not be successfully fetched. A more comprehensive URL based detection system was introduced by Thomas *et al.* [34]. They designed a real-time system named Monarch which crawled URLs as they were submitted to web services and determined whether the URLs direct to spam. This system can work for both email system and social network. They also explored the distinctions between email and Twitter spam with their system and revealed several difference between them. The advantages of the above two URL based strategies include: they can offer real-time spam filtering service and the underlying characteristics of spam are general for most web service. However, the URL based strategies are powerless to detect spam content without URL and expensive in maintaining a real-time URL tracking system.

Flores *et al.* [43] combined web search with spam detection on OSNs. They proposed a system which can determine whether a given account was fraudulent or not based on web search result. They found that legitimate users often register on multiple social networks with the same, or similar names. In contrast, spammers seldom have such a dynamic web presence. So they adopted web search to measure the online presence of a user and regarded accounts which had insufficient web presence to likely be fraudulent. This web search based strategy does not depend on social graph information or content posted by the users, so it can detect spammers before they take any spam actions.

Chapter 3

Study of Active Honeypots

In this chapter, we demonstrate the existence of active honeypots and discuss our observation on several interesting properties of active honeypots. Then we will analyse why active honeypots are attractive to spammers by observing the behavior of active honeypots as well as analysing potential reward mechanism for spammers based on these behaviors. At last we build some accounts to imitate the potential attractive behaviors of active honeypots. The experiment results reveal the in-depth mechanism behind the attractiveness of active honeypots.

3.1 Collection of Active Honeypots

In this section, we demonstrate the collection of active honeypots from Twitter which serves as the basis for the observation. The data set is obtained between November 2012 and December 2013. We crawled 4 million randomly selected Twitter accounts from the public stream of Twitter. Due to the limitation of our hardware resource and Twitter API, our observation of these accounts is divided into two stages. In the first stage, for each account, we observe its daily variation for a time period of 3 days. The daily variation includes new friends, new followers and recent 100 tweets each day. New friends (new followers) can be obtained by comparing friend lists (follower lists) between two consecutive days. Since most accounts (99.7%) have fewer than 300 new followers per day, we only record 300 new followers (new friends) at most for each account per day. Note that due to the limitations of Twitter API requests, the observations for all the accounts may be crawled during different time periods. In the second stage, we first select out accounts which are most attractive to

spammers based on the observation of first stage. Then we monitor the daily variation of these accounts for at least one month to check whether they are real attractive to spammers.

We rely on Twitter's account suspension service to label social spammers, which is adopted in previous works [3, 10, 12]. As shown in [3], 97% suspended accounts are suspended within two weeks by Twitter. Therefore, two weeks after the crawling period, we checked the number of suspended accounts which followed, or were followed by, or were mentioned in tweets by each account every day. For each account, we calculate the average number and ratio of spammers interacting with this account during the one month observation period.

In Fig. 3.1 we presents the average number and ratio of spammers of 500,000 accounts which are randomly sampled from the 4 million accounts crawled in the first stage observation. Each point in Fig. 3.1 represents one account. Among the 500,000 accounts, 86.3% accounts attracted no spammers. Based on the average number and ratio of spammers each account can trap, we manually select 1,841 accounts, each of which is followed by more than 10 spammers per day for average with an accumulated ratio of spammers higher than 0.2. These accounts are taken as active honeypots. The reason why we consider both the number and ratio of spammers is that some popular accounts like Justin Bieber have a large number of spammers among their new followers but with a very low spam ratio, which is actually not efficient in trapping spammers. Accounts in the grey area of Fig. 3.1 are the active honeypots we selected. In order to gain more insights, the observation of these accounts continue for at least one month, some accounts are observed for more than a half year.

3.2 Properties of Active Honeypots

In this section we will discuss our observation on several interesting properties of active honeypots. Before we start our discussion, we first introduce several terminologies. Active followers (active friends) are followers (friends) of active honeypots. Accounts interacting with active honeypots refer to all the followers, friends, mentioned users in tweets of active honeypots and users who mention active honeypots. Influential accounts refer to legitimate accounts on Twitter which own more than 2000 followers. Random accounts refer to those account which we randomly crawled from public stream of Twitter.

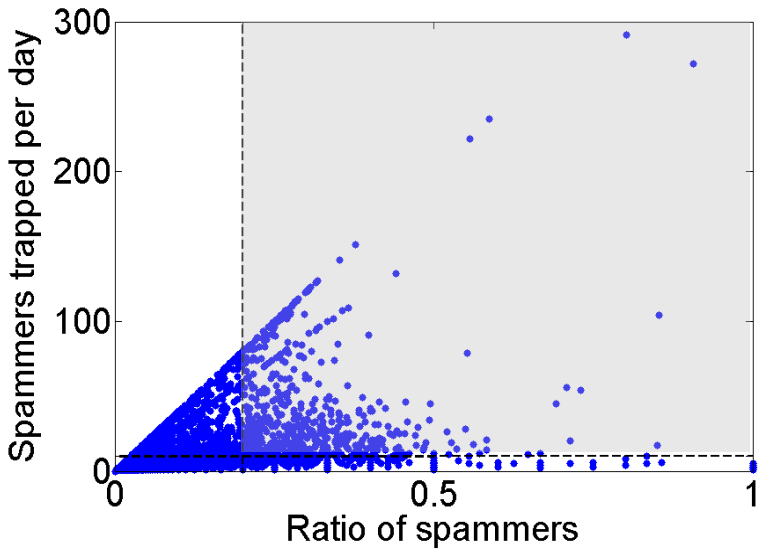

Fig. 3.1 Ratio vs. Number of social spammers trapped by seeding accounts

3.2.1 Active Honeypots Are Efficient in Trapping Spammers

Active honeypots are efficient in trapping spammers. As we mentioned in section 3.1, the active honeypots we select out can attract at least 10 spammers per day with a spam ratio higher than 0.2.

In order to demonstrate active honeypots can stably attract spammers, we also present the number and ratio of spammers attracted by 243 active honeypots everyday during a one month period shown in Fig. 3.2. Each point on the curve represents the total number of spammers attracted by these 243 active honeypots on a certain day and each cross represents the ratio of spammers among all the accounts attracted by these active honeypots on a certain day. We don't present all the 1,841 active honeypots we crawled because those active honeypots are crawled in different time, some early crawled accounts are no longer efficient after several months and for some recent crawled active honeypots the spammers among their interacting accounts haven't been fully suspended by Twitter. As we can see, the ratio of spammers among followers of active honeypots is quite stable which only drop 3% in one month period. However, The number of spammers trapped by these active honeypots everyday has obvious decrease. The daily crawled spammers have dropped 13% in one month period. The decrease is because some active honeypot become no longer attractive to spammers. According to our analysis in the rest of the paper, some active

honeypots buy followers from spam campaigns and the business only last for a certain period of time. In a long term observation, 64% of the active honeypots we crawled half a year ago are no longer attractive to spammers, and 12% of them have been suspended by Twitter, but there are still 36% active honeypots attractive to spammers after such a long time.

To illustrate the advantages of active honeypots in trapping spammers, we compare the number of spammers trapped by active honeypots with the number of spammers trapped by passive honeypots on Twitter in 2 previous works [8, 9]. Table 3.1 shows the results. As we can see, the 243 active honeypots are 8000 times more efficient than the passive honeypots in trapping spammers. Note that according to our statistics the number of spammers trapped by the 243 active honeypots only include the spammers which have been successfully identified by Twitter. Since many spammers and fake accounts may not be successfully identified by Twitter, the actual number of spammers trapped by the 243 active honeypots can be even larger and the spam ratio can be much higher. Furthermore, with long term accumulation, we can use much more than 243 active honeypots to trap spammers.

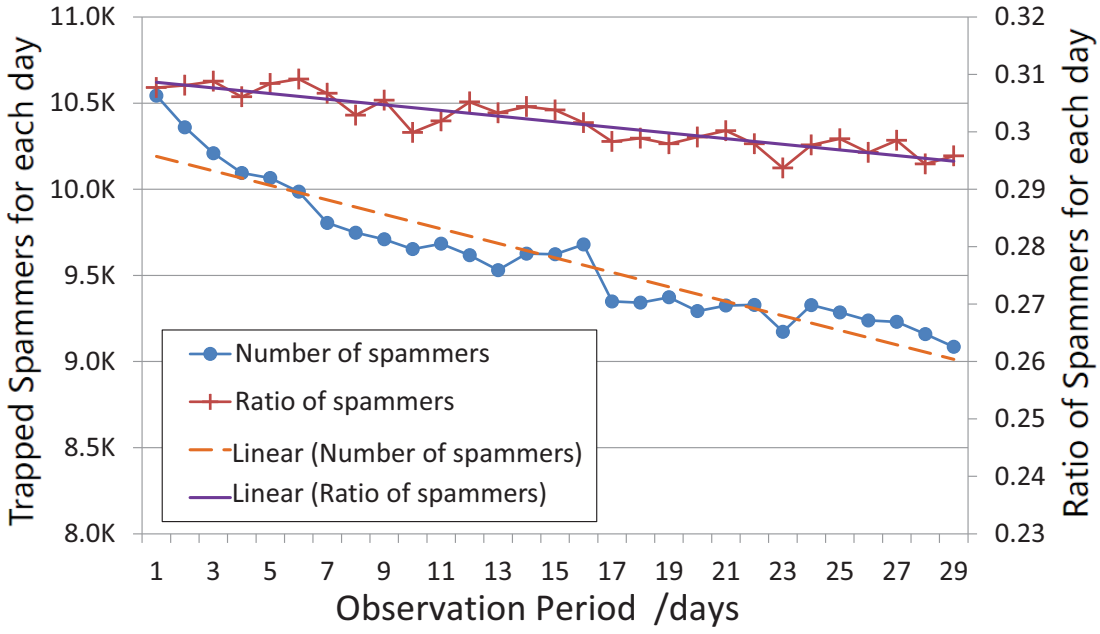

Fig. 3.2 Social spammers trapped by 243 active honeypots

Table 3.1 Active honeypots vs. passive honeypots

	Num. of honeypots	Period	Num. of spammers	Ratio of spammers
honey profile [9]	300	11 months	361	0.91
social honeypots [8]	N/A	1 months	< 500	N/A
active honeypots	243	1 months	**232,037**	**0.30**

Though a fairly large portion of the accounts trapped by the active honeypots still have not been flagged as spammers by Twitter, the behaviors of these followers are very suspicious to be fake accounts which are probably to be used in future spamming activities. To demonstrate this, we randomly select 4 influential accounts and 4 active honeypots. Fig. 3.3 shows the distribution of the created time for those accounts following them in one day. Each sub-graph represents accounts following one active honeypot(influential account). The X axis represents the created time and 0 represents the day these accounts follow an active honeypot(influential account), so -x represents x days before they follow this active honeypot(influential account). We only show the distribution of the latest 80 days. For active honeypots (in blue), more than 50% accounts are created within 10 days before they follow active honeypots. For influential accounts (in red), less than 30% followers are created within this time period. Besides, there is obvious hopping in the curve of active honeypots, demonstrating that many followers of active honeypot are created in a short time period. We guess these followers are created by bots automatically and controlled by the same spam campaign.

3.2.2 Active Honeypots Are Influential Accounts

Follower number and the ratio between friend number and follower number (FF-ratio) are two important metrics used in measuring the influence of an account [3, 11]. Tweets posted by a user will appear in his timeline and can be seen by all his followers. Accounts with a large number of followers will influence a lot of accounts by posting tweets. FF-ratio indicates the popularity of an account, popular account usually have few friends but a large number of followers so they get a small FF-ratio, spammers will follow a lot of accounts but few accounts will follow them so they get a big FF-ratio, most normal users usually follow their friends and their friends will follow back so they get a FF-ratio close to one. FF-ratio can influence the ranking of accounts and tweets in OSNs. If an account has a large follower

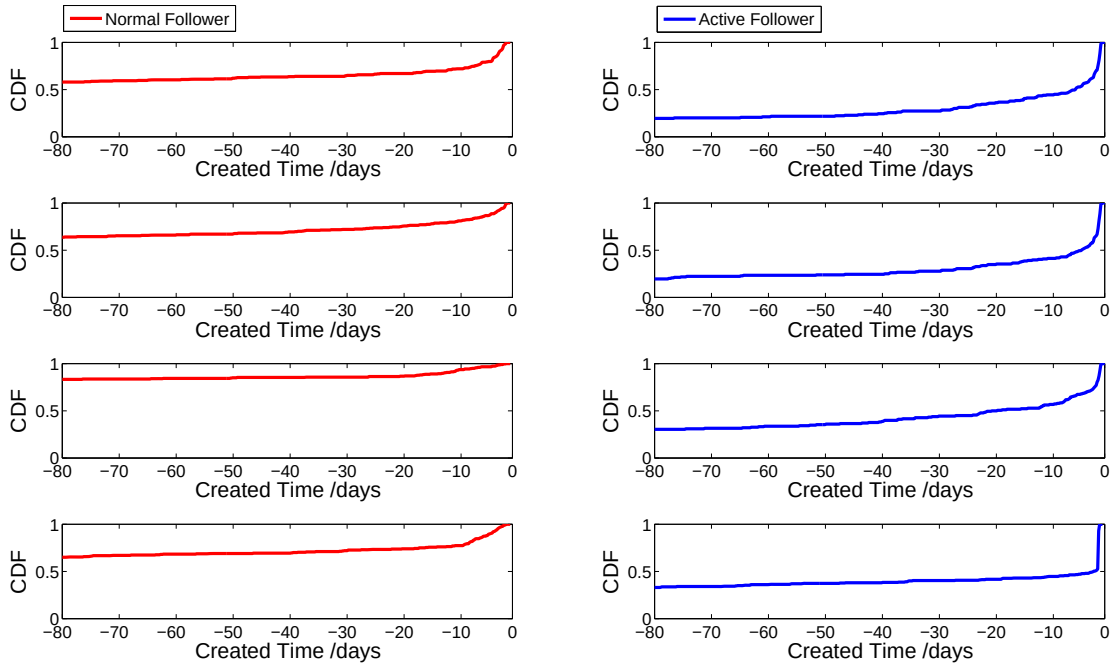

Fig. 3.3 Followers created time distribution of active honeypots and influential accounts

number and small FF-ratio, then this account is probably an influential account. Fig. 3.4 shows the follower number and FF-ratio for active honeypots and random selected accounts from public stream (random accounts). As we can see, 96.4% active honeypots have more than 2000 followers and FF-ratio smaller than 1.2, which means that most active honeypots are influential accounts. Compared with active honeypots, there are only 6.53% random accounts which have more than 2000 followers and FF-ratio smaller than 1.2.

3.2.3 Active Honeypots Hide Themselves behind Other Influential Accounts

Though active honeypots interact with many spammers every day, active honeypots succeed in hiding themselves behind other influential accounts in terms of their profiles and tweets. To prove it, we compare the profile settings between active honeypots and other accounts which have more than 2000 followers and FF-ratio smaller than 1.2 (random influential accounts). Table 3.2 compares profile settings between active honeypots and random influential accounts. As we can see, except the ratio of biography with URLs, all the other settings are quite similar.

The most important approach for spammers to spread spam content is posting tweets

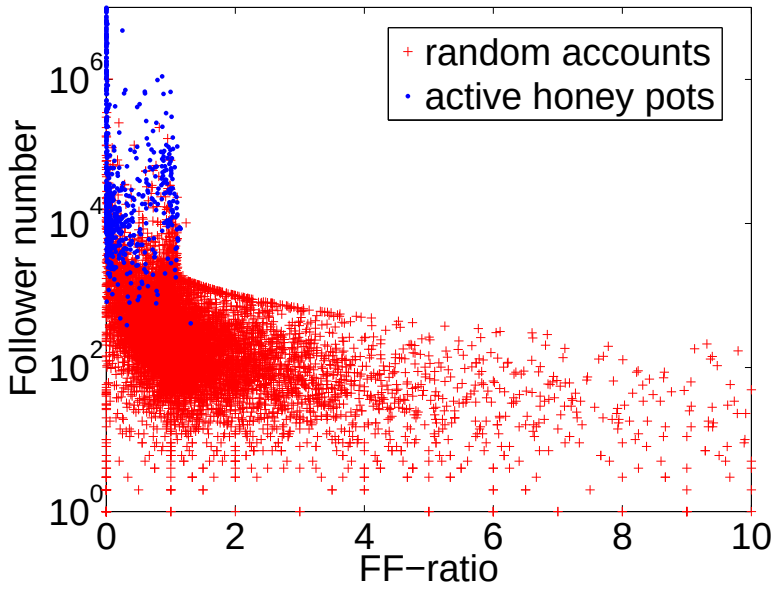

Fig. 3.4 Follower number vs. FF-ratio of active honeypots and random accounts

Table 3.2 Comparison of profiles between active honeypots and random influential accounts

	Random influential	Active honeypots
default profile image	2.1%	0.3%
default background image	96.4%	93.7%
with description	86.3%	86.9%
description with hashtags	15.2%	13.0%
description with URLs	5.2%	15.0%

with spam URL. To figure out whether active honeypots take part in spamming activities directly by posting spam URL, we extract the URLs contained in tweets of active honeypots and random influential accounts and checked these URLs with URL blacklist service. The URL blacklist service we used including: Google Safebrowsing [44], URIBL [45], Joewein [46], SURBL [47] and McAfee [48]. If an URL is blacklisted in one of the five URL blacklist service, we regard this URL as a spam URL. After checking, only 0.32% URLs posted by active honeypots and 0.11% URLs posted by random influential accounts have been identified as spam URLs by URL blacklist service. This demonstrates that active honeypots seldom post spam URLs in tweets the same as the random influential accounts do.

Active honeypots maintain their profiles similar to other influential accounts and seldom post spam URLs in tweets so that active honeypots can hide themselves behind other influential accounts and prevent themselves from being suspended by Twitter.

3.3 Attractiveness Analysis

In this section, we analyse why active honeypots are attractive to social spammers. We first examine whom active honeypots interact with. Then we analyse 4 behaviors of active honeypots including following back, mentioning, retweeting and posting tweets with sensitive keywords. These behaviors are proven to be highly related with spamming activities in previous works [10,12,49]. Based on the analysis of these behaviors, we propose 5 potential reward mechanisms which make active honeypots attractive to social spammers. The data set used in this analysis are 1,841 active honeypots and 100,000 random influential accounts identified from the 4 million accounts.

3.3.1 Whom Active Honeypots Interact with

To reveal whom active honeypots interact with, we examine the characteristics of the followers of all the 1,841 active honeypots (termed as "active followers") by comparing them with all the followers of random influential accounts (termed as "random influential followers"). Here we only care about the following behavior of active followers and random followers. We think they follow active honeypots or random influential accounts because they are attracted by these accounts and want to have interaction with them. We compare the follower number and tweet number between active followers and random influential followers. Fig. 3.5 and Fig. 3.6 show the results. The sharpest contrast in Fig. 3.5 results

from about 18% active followers have less than 10 followers, while only 7% random influential followers have less than 10 followers. The sharpest contrast in Fig. 3.6 results from about 27% active followers have less than 10 tweets while only 13% random influential followers have such few tweets. Since accounts with less than 10 followers and 10 tweets are inactive accounts [3, 12], probably fake accounts created by spamming organizations, the comparison results imply that a large amount of active honeypots are probably promoted as influential accounts with the help of social spammers. This is because active honeypots attract followers mainly through buying fake followers from spam merchants or offering some rewards to followers. So they mainly attract spam or fake accounts to follow them. While random influential accounts are those really popular users on Twitter, they attract followers with their special personal charming or by contributing valuable content. Thus they can attract lots of normal users on Twitter which are active users.

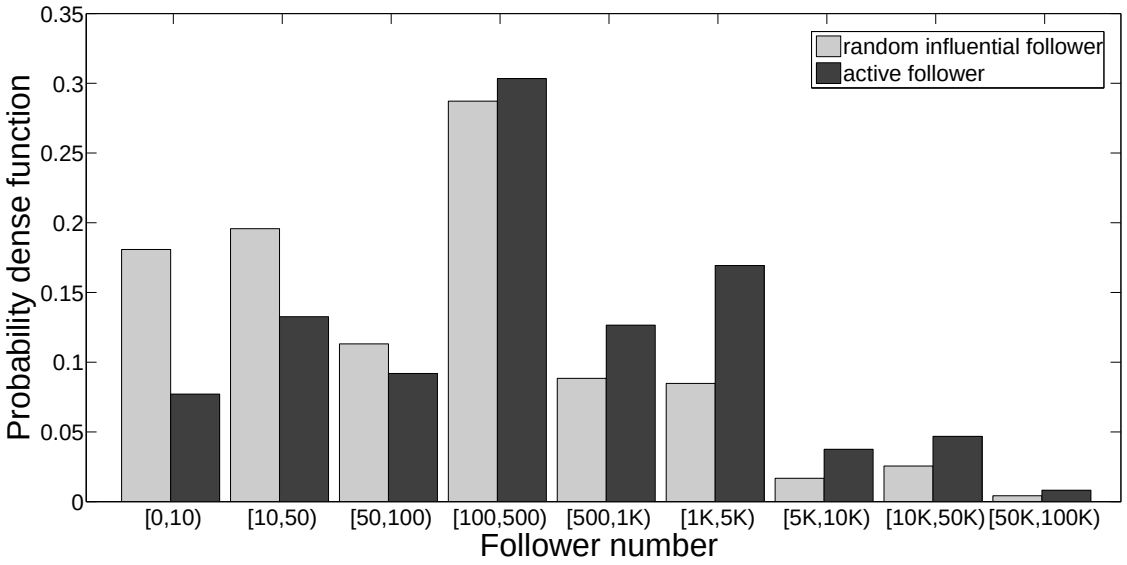

Fig. 3.5 Follower number comparison between active followers and random followers

3.3.2 Active Honeypots Follow Back Their Followers

Some existing works [10, 11] find that social spammers tend to follow those accounts which will follow them back. In order to verify whether active honeypots are exactly this type of accounts, we calculate the follow back ratio (FB-ratio) of each account, which is defined

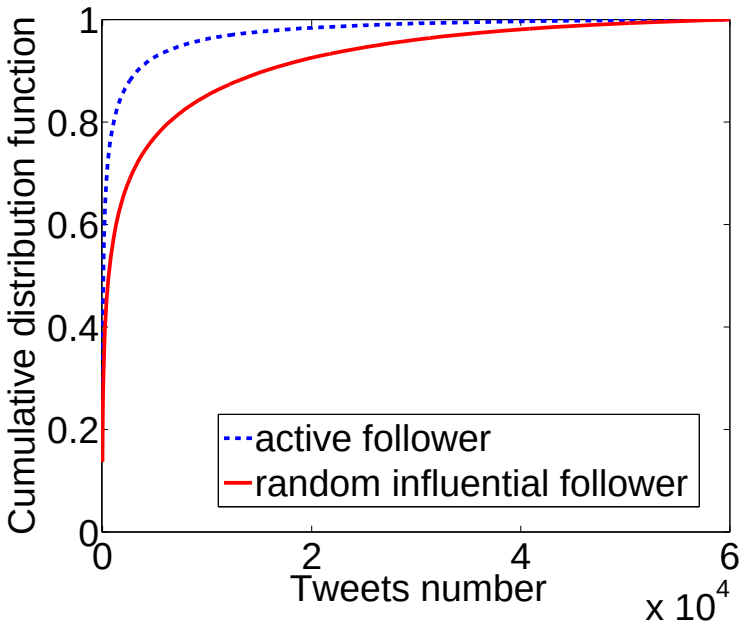

Fig. 3.6 Tweet number comparison between active followers and random followers

as:

$$\frac{\|(friend\,set\ \cap follower\,set)\|}{\|follower\,set\|} \tag{3.1}$$

where $\|\cdot\|$ denote the size of a set. We compare the FB-ratio between the active honeypots and random influential accounts. Fig. 3.7 shows the follower number and FB-ratio for both active honeypots and random influential accounts. Each point represents one account. According to Fig. 3.7, only 21% active honeypots have FB-ratio larger than 0.5. These active honeypots follow back half of their followers. While 57% active honeypots have FB-ratio smaller than 0.05. Obviously, a significant portion of active honeypots do not follow back their followers. On the other hand, we observe that 49% random influential accounts who have FB-ratio larger than 0.5. Though these random influential accounts have large FB-ratio, they seldom follow back social spammers. We guess this is probably because the random influential accounts avoid to follow accounts which look like social spammers. This also indicates the difficulty for social spammers to gain influential followers even though there exists a large portion of influential accounts which have high FB-ratio on Twitter.

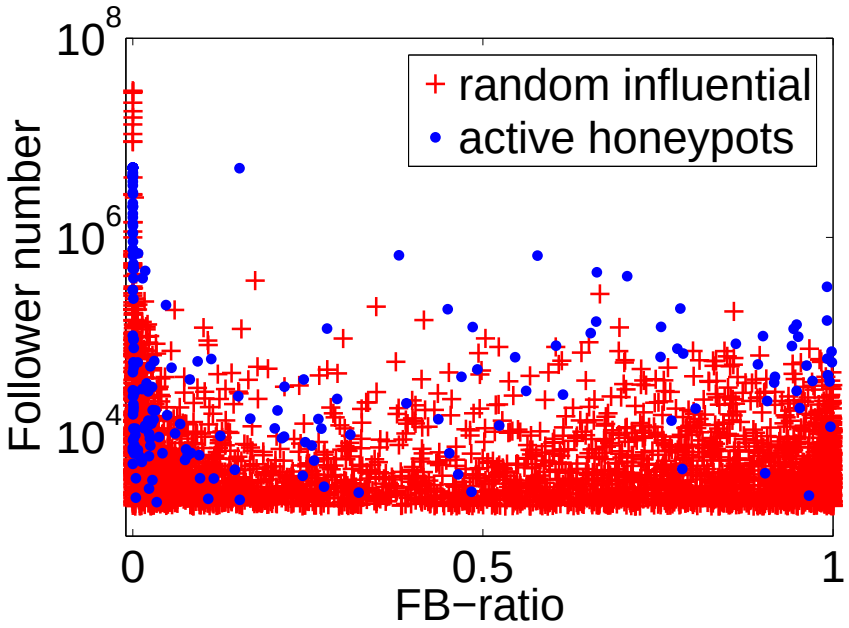

Fig. 3.7 Follower number vs. FB-ratio of active honeypots and random influential accounts

3.3.3 Active Honeypots Mention Spammers and Unrelated Accounts

Based on our observation, we found active honeypots like to mention spamming accounts and unrelated accounts. To prove our conclusion, we analysed users mentioned by active honeypots in tweets. For convenience, we call users mentioned by active honeypots as mentioned users and the relation as mention relation. We first analysed the ratio of spammers among mentioned users for active honeypots and random influential accounts. Fig. 3.8 shows the cumulative distribution function for the ratio of social spammers among mentioned users for active honeypots and random influential accounts. As we can see, about 88% random influential accounts have never mentioned spammers, while only 27% active honeypots have never mentioned spammers. About 38% active honeypots have more than 25% mentioned users as social spammers. On the contrary, only 4% random influential accounts have more than 25% mentioned users as social spammers. Obviously, there are clear differences in mentioned users between active honeypots and random influential accounts. Active honeypots are more willing to mention spammers than random influential accounts.

Normally the mention relationship takes place between an account and its related accounts(*i.e.* friends or followers). To verify this, we further checked difference in the ratio

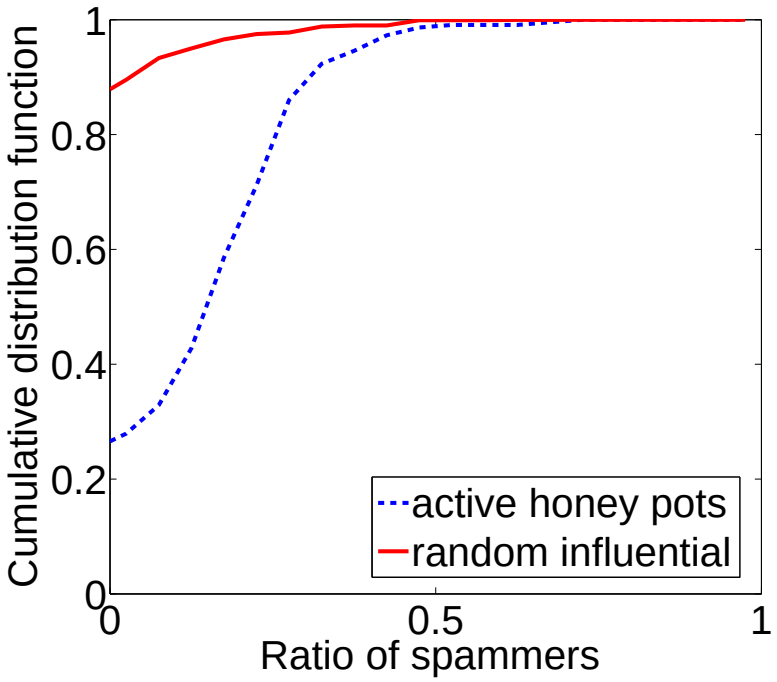

Fig. 3.8 Ratio of social spammers among mentioned users

of unrelated mention between active honeypots and random influential accounts. Here unrelated mention means an account mentions another user which is neither its friend nor its follower. Since accounts suspended by Twitter will be invisible in the friend lists and follower lists which used to follow or be followed by the suspended accounts, we remove the suspended accounts from the mentioned users in this experiment. Fig. 3.9 shows the result. As we can see, 55% random influential accounts have unrelated mention ratio smaller than 0.2, while only 24% active honeypots have unrelated mention ratio smaller than 0.2. It is also worth noting that about 15% active honeypots completely mention unrelated users. So we can say active honeypots mention more unrelated accounts in tweets than random influential accounts.

3.3.4 Active Honeypots Retweet for Spammers

Active honeypots offer retweeting service for spammers. To prove this, we compared the retweet ratio, retweet count and spam ratio of original posters between active honeypots and random influential accounts. We define retweet ratio as the ratio of retweeted tweets

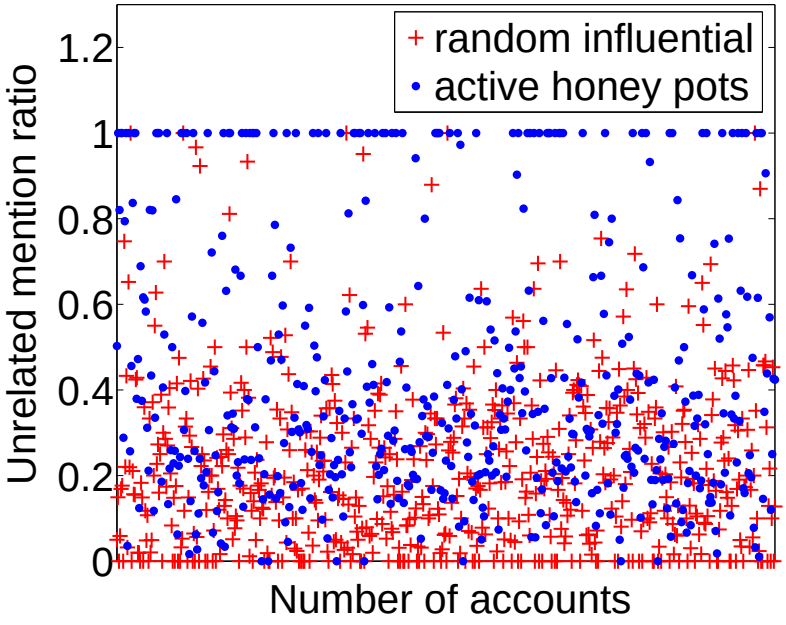

Fig. 3.9 Ratio of unrelated mention of active honeypots and random influential accounts

among all the posted tweets of a user. In Fig. 3.10, we present the probability dense function of retweet ratio for active honeypots and random influential accounts. About 70% active honeypots have a retweet ratio higher than 0.5 while only 35% random influential accounts have a retweet ratio higher than 0.5. In addtition, about 13% random influential accounts have a retweet ratio smaller than 0.05 which means they seldom retweet tweets of others, while only fewer than 1% active honeypots do like this. On the contrary, about 12% active honeypots have a retweet ratio higher than 0.95 which means almost all their tweets are retweeted tweets from others, while only 4% random influential accounts do the same. All in all, active honeypots are more willing to retweet tweets for others.

To further learn about the quality of those retweeted tweets, we present the distribution of retweet count for active honeypots and random influential accounts in Fig. 3.11. Retweet count represents how many times the tweet has been retweeted. Generally, a larger retweet count means higher quality of the tweet [49]. We can find that about 91% retweeted tweets of active honeypots are retweeted for less than 100 times. As for random influential accounts 85% retweeted tweets are retweeted for less than 100 times. The retweeted tweets of active honeypots acquire smaller retweet count than random influential accounts. So we

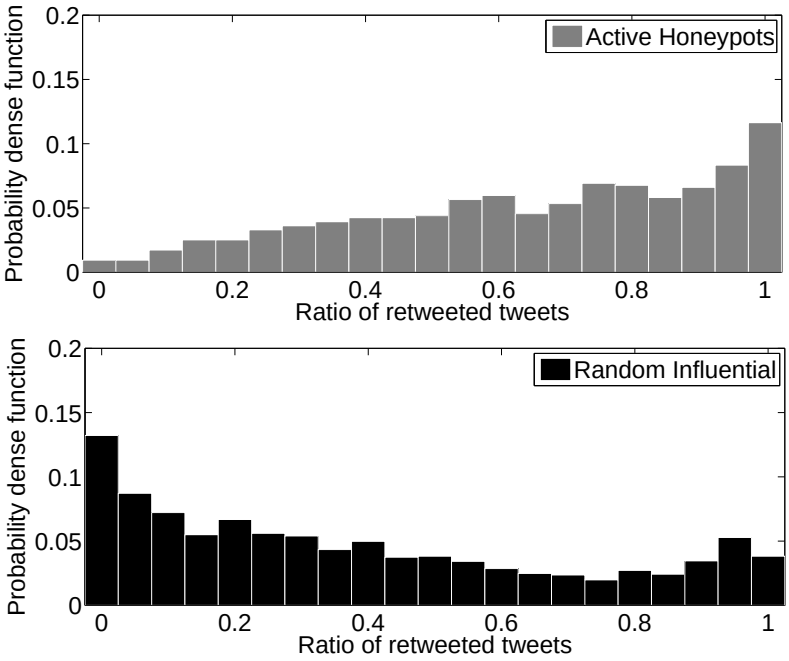

Fig. 3.10 Retweet ratio of active honeypots and random influential accounts

can say, active honeypots retweet more low quality tweets than random influential accounts do. Besides, we checked all the original posters of those retweeted tweets. 13% posters, whose tweets were retweeted by active honeypots, have been suspended by Twitter. As for random influential accounts, this ratio is only 6.7%. Based on all the analysis above, we can say active honeypots are more willing to retweet low quality tweets for spammers than random influential accounts.

3.3.5 Active Honeypots Post Sensitive Keywords in Tweets

Active honeypots post certain sensitive keywords, which are attractive to spammers, in their tweets. According to Sridharan *et al.*'s [12] investigation, spammers will pick their targets based on the content of tweets from Twitter users. For instance, a spam campaign which want to promote a kind of diet pills may target users who have the word "weight lose", "slim" or "fat" in their tweets. It is easy to find such kind of tweets with the help of Twitter search based on keywords. We guess active honeypots also post such kinds of keywords in their tweets.

To prove it, we used the term frequency-inverse document frequency (TF-IDF [50])

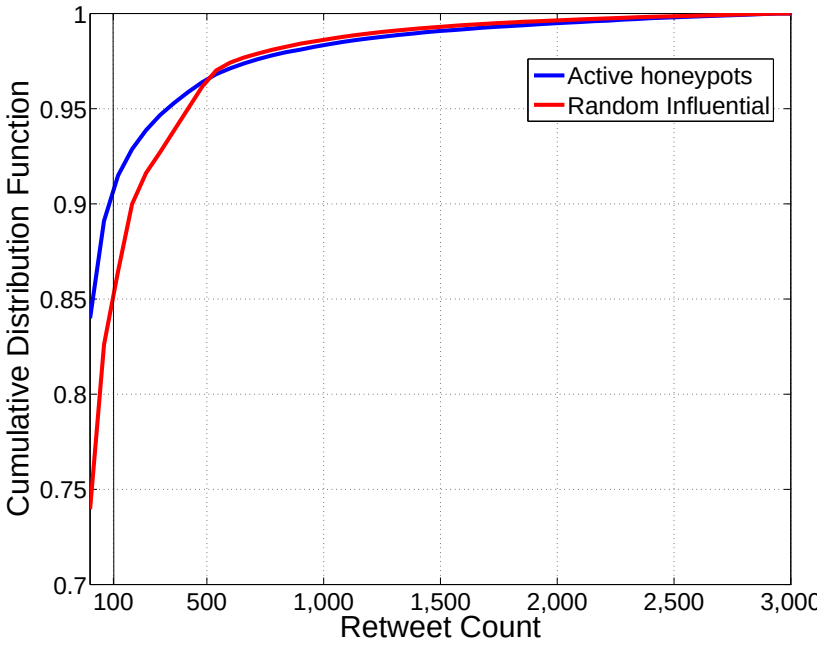

Fig. 3.11 Retweet count of active honeypots and random influential accounts

statistic which reflects how important a word is to a document in a collection to extract the most important keywords in tweets of active honeypots as well as random influential accounts. We eliminated those pronouns, prepositions and modals in the result and present part of the keywords we extract in Fig. 3.12. We can find that "follow" and "retweet" are important keywords in tweets for both active honeypots and random influential accounts. This is because acquiring more followers and more retweeting are common need for most accounts on Twitter. Twitter allow users to get more followers and retweeting only if they do not violate Twitter rules [51]. However there are some obvious difference between the keywords of active honeypots' tweets and random influential accounts' tweets. The keywords posted by random influential accounts are mainly related to "retweet" and "follow". Besides, there are a few keywords expressing the feeling such as "hate", "fucking" and "thank". In contrast, fewer keywords posted by active honeypots are related to "retweet" and "follow". Some keywords of active honeypots are related to porn such as "pussy", "sex", "bitch" and "cock". Some other keywords such as "slim", "iPhone", "ante" and "Tweetbot" are related to those most common promotion topics on Twitter [2] including: losing weight, Apple products and gambling. According to Sridharan's [12] experiment,

these keywords are usually used by spammers to find targets. So we can see the keywords contained in the tweets of active honeypots are more temping or promotion related while the keywords in the tweets of random influential accounts only focus on getting followers and retweeting.

(a) Keywords in tweets of active honeypots

(b) Keywords in tweets of random influential accounts

Fig. 3.12 Keywords in tweets of active honeypots and random influential accounts

3.3.6 Potential Reward Mechanisms behind Attractiveness

To analyse the potential factors influencing the attractiveness of active honeypots, we assume that the behaviors of social spammers are profit motivated. To put it in another way, active honeypots reward spammers with their behaviors so that spammers are willing to interact with them. Based on this assumption, we infer that social spammers are attracted by active honeypots because of the following 5 types of rewards offered by active honeypots.

Rewarding by following back: The first type of reward that attracts social spammers is that about 21% active honeypots follow back more than 50% of their followers according to Fig. 3.7. This type of active honeypots are also analysed in two recent works [10, 11]. Both of them consider this type of active honeypots as the major supporters for spammers to increase the influence of their spam content. Our studies show that a large portion of active honeypots no longer use following back to attract social spammers, which is probably because some techniques have been proposed to fight against this behavior [10, 11].

Rewarding by mentioning: The second type of reward that attracts social spammers is that 38% active honeypots frequently mention social spammers in their tweets. Since mention tweets will be visible to all the followers of active honeypots, this type of reward can also help expand the influence of social spammers. Fig. 3.13 shows the FB-ration and the ratio of mentioned spammers for active honeypots. Active honeypots mainly gather in the region of red circle, which represents a small FB-ratio but high ratio of mentioned spammers. In other words, active honeypots in this circle prefer mention spammers in tweets rather than follow them back. So we can see, mentioning users is now a major reward used by active honeypots to attract social spammers, as 45% active honeypots have more than 20% mentioned users as spammers.

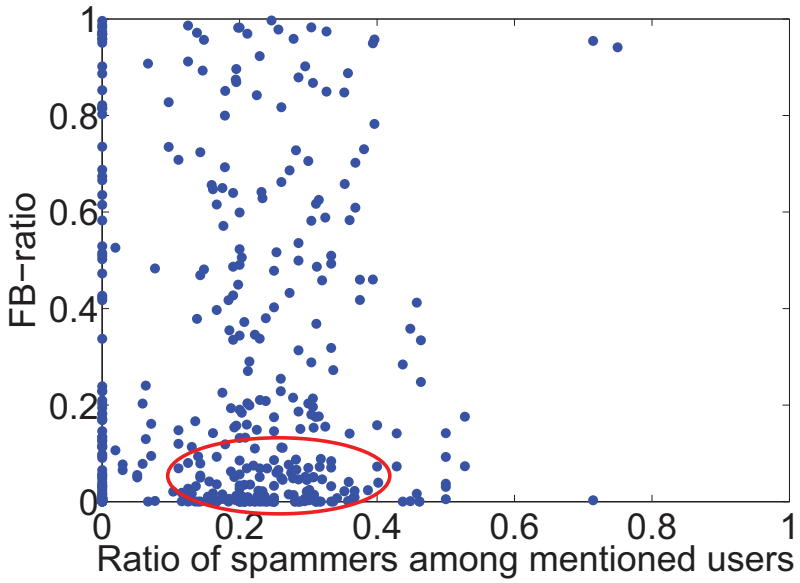

Fig. 3.13 FB-ratio vs. spammer ratio of mentioned users for active honeypots

Rewarding by retweeting: The third type of reward that attracts social spammers is that about 18% active honeypots retweet for spammers frequently. Here "retweeting for spammers frequently" means more than 20% of the original posters of their retweeted tweets are spammers suspended by Twitter. According to [49], a retweeted tweet can reach an average of 1,000 users and once retweeted, a tweet gets retweeted almost instantly on next hops. Active honeypots are all influential accounts which have at least 2,000 followers and some of them even have over one million followers. Retweeting tweets will be displayed in the timeline of active honeypots and can be seen by all the followers of active honeypots. The retweeted tweets can be further retweeted by the followers of active honeypots again. So retweeting can spread those spam content efficiently. Though random influential accounts will also retweet for other accounts, our investigation reveals that active honeypots are prone to retweet low quality tweets for spammers.

Rewarding by posting tweets containing keywords: The fourth type of reward is that active honeypots post tweets containing sensitive keywords. This type of reward is a little different from the above three reward mechanisms. The above three rewards are all directly given to spammers by active honeypots. The fourth type, rewarding by posting tweets containing sensitive keywords, is an indirect way of reward. Spammers may find the tweets containing certain keywords which are posted by active honeypots actively and reply to these tweets with spam message. We extracted some of these tweets and observe all the reply of them, we found many spammers directly replied or retweeted these tweets. By replying to these tweets with spam content, spammers can expose the spam message to other legitimate users who also reply to these tweets. For example, some active honeypots are porn accounts and they will post tweets containing porn content. Among the reply of these tweets we can find many spammers who post spam link of selling Viagra. Those legitimate users who also pay attention to these porn tweets will have a high probability to buy this product.

Rewarding by offline benefits: Apart from the above four types of rewards, We also find that as high as 15% active honeypots which follow back less than 5% of its followers, never mention spammers, seldom retweet for spammers nor post tweets with sensitive keywords. We guess these active honeypots paid to get followers in order to make themselves look more influential. To prove this, we investigated the business of buying Twitter followers and present our results in Table 3.3. We present 5 popular sellers who sell Twitter followers. Each seller claims that they can offer hundreds of thousands of

followers. It is cheap to get a large number of followers since each follower only cost about 1 cent and can be even cheaper if you buy a large number of followers. A seller only offers 300 to 1600 followers for a buyer everyday to avoid the detection of Twitter. In order to learn about the quality of these followers, we buy 1000 followers from each seller and found that these followers have complete profiles which contain profile images and self descriptions. But most of them only post a few tweets and have quite few followers which make them easy to be distinguished from legitimate followers. The popularity of buying Twitter followers shows that there must be many spam and fake followers on Twitter. A report [52] shows that even the president of United States − Barack Obama bought millions of fake accounts to make his Twitter account look more influential.

Table 3.3 Twitter follower selling business

Seller	Controlled accounts	Price (cent /follower)	daily offered followers
Devumi	3×10^5	0.8	1000
Fastfollowerz	10×10^5	1.0	1200
Buy real marketing	1×10^5	0.12	1600
GetMore Followers	3×10^5	1.4	300
Follower sale	5×10^5	1.2	500

3.4 Imitate Behaviors of Active Honeypots

Based on our previous observation of active honeypots we have identified several user behaviors and the corresponding reward mechanisms which probably make active honeypots attractive to spammers. The potential attractive behaviors include: following back, retweeting, posting tweets with sensitive keywords and mentioning unrelated accounts. Now we have a question: can an account become attractive to spammers by imitating these behaviors? If the answer is yes, it can further prove that these behaviors and their corresponding reward mechanisms make active honeypots attractive to spammers. If the answer is no, then what else factors are necessary to make active honeypots attractive. To figure out the problem, we conducted an experiment in which we created some Twitter accounts to imitate these behaviors and measured the effectiveness of these behaviors in attracting social spammers.

3.4.1 Ethical Considerations

Our imitation experiment in this section may violate the Twitters rules [51]. To minimize the risk we may cause to Twitter and its users, we present our ethical considerations and guidelines in the following.

In our experiment we focused on whether these potential attractive behaviors make accounts attractive to spammers. In order to control accounts to do specified behaviors on Twitter as well as meet the requirement of sample number and controlled experiment, we had to build a group of fake accounts on Twitter which are controlled by us. We only use these fake accounts for research purpose and won't use them for any spamming activities to earn any benefits. We create these fake accounts by ourselves rather than buy compromised accounts from underground account market. We stop using these accounts after experiment and won't sell them to anyone else. Some of our behaviors may violate Twitter rules [51] such as mentioning unrelated accounts, massively following and posting sensitive keywords in tweets. In order to lower the influence of these behaviors to Twitter or its users. We will only target one specific user once and won't target on certain users massively. We won't post spam URLs in tweets nor post duplicate content over multiple accounts. We won't promote any service, links, topics or products in our experiment. We lower the frequency of our behaviors to a tolerable threshold of Twitter which means the frequency of behaviors won't cause large number of accounts being suspended by Twitter. We buy as few fake followers as possible from underground account market to meet the requirement of our experiment.

3.4.2 Imitation Experiment Setup

In our experiment, we use 140 Twitter accounts which were created by ourselves one year ago. All these accounts look like legitimate accounts which means they have unique profile images, detailed profile settings, a number of normal tweets and more than 200 followers. Without careful discrimination, it is hard to distinguish these accounts from legitimate accounts on Twitter. We divide these accounts into 7 groups averagely. Each group of accounts have unique behavior or account characteristic which may be attractive to spammers. We compare the number of accounts and spammers attracted by each group to measure the effectiveness of different behaviors in attracting accounts and spammers. In table 3.4, we present the group setting in our experiment.

Table 3.4 Group setting for imitation experiment

Group Name	Group Description
Group A	do nothing
Group B	mention unrelated accounts
Group C	retweet for unrelated accounts
Group D	post tweets with sensitive keywords
Group E	follow unrelated accounts actively & follow back
Group F	have all the attractive behaviors
Group G	influential accounts and have all the attractive behaviors

Group A is the control group and all the accounts in this group do nothing. Accounts in group A are similar to passive honeypots. They do not take any active actions to attract spammers and just wait for unsolicited following or mentioning from other accounts. We set this group as a comparison to other groups as well as to check whether passives active honeypots are still working.

Group B contains accounts which mention unrelated accounts. We chose mention targets randomly from Twitter public stream. The mention we used include active mention and reply mention. The active mention means we directly mention the screen name of a user in our tweet. The reply mention means we reply a tweet posted by a user and Twitter will automatically mention the poster. We set this group to check the attractiveness of mentioning behavior. According to Sridharan's investigation [12] few legitimate accounts on Twitter will mention or reply to spammers. So we conjecture if an account mention spammers actively it will be regarded as being friendly to spammers.

Group C contains accounts which retweet for unrelated accounts. We chose target tweets for retweeting from Twitter public stream. This group is set to check if retweeting service can make accounts attractive to spammers. Retweeting service is a win-win strategy for retweeters and spammers. It can help retweeters to attract spammers as well as help spammers to spread spam messages. So we guess if we offer this service we may also get the reward from spam followers.

Group D contains accounts which post tweets with sensitive keywords. We use the keywords we extract in section 3.3.5 as our keywords library and post tweets with these keywords. In order to post related tweet content with the keywords, we use Twitter search

to find related tweet content as our content library. When we post tweet, we combine the keywords with corresponding content templates. We set this group to check the attractiveness of sensitive keywords in Tweets. As we mentioned in section 3.3.5, searching tweets based on certain keywords is a main approach for spammers to pick targets.

Group E contains accounts which follow lots of unrelated accounts actively. We choose targets from Twitter's public stream randomly and follow them. Since we cannot realize following back behavior directly, we use active following instead. This group is used to check the attractiveness of following behavior. According to our above analysis, most spammers can only get fewer than 10 followers so they are easy to be detected. We guess if we follow spammers actively, we may become an important resource for spam campaigns to get followers and spammers will be willing to interact with us.

Group F contains accounts which do all the behaviors we listed above. We set this group to check whether combination of these attractive behaviors will enhance the attractiveness to spammers.

Group G contains accounts which also do all the behaviors above, but the difference to group F is that accounts in this group are all influential accounts which have more than 2000 followers. We buy these followers from Twitter follower market and all these followers look like real accounts. We set this group to check whether influence of an account is related to its attractiveness to spammers. According to our analysis in section 3.2.2, more than 96% active honeypots have more than 2000 followers. We guess spammers may be more willing to follow influential accounts since retweeting or mentioning service offered by these accounts are more influential.

Our experiment last for 2 months. We built a Twitter bot to control these accounts and made all the groups keep doing their behaviors during the period. In order to lower the probability of being suspended by Twitter, we lowered the frequency of attractive actions to a relative safe limit. However, some accounts were still suspended by Twitter. When an account was suspended, we would use a similar new account to replace it. During our experiment we collected all the accounts interacted with each group and found the spammers among them. We labelled spammers with Twitter suspension service the same as we did in section 3.1.

3.4.3 Imitation Experiment Result

Before we evaluate the results of the experiment we first introduce some definitions. First, we define mentioning a user, retweeting a tweet, posting a tweet or following a user as an action. Second, we regard all the accounts which mention or follow our experiment accounts as being attracted by our accounts and we regard every following or mentioning action from other accounts as an interaction. The following interaction refer to following action from other accounts. The mention interaction refer to mentioning action from other accounts.

We first evaluate the attractiveness power of different behaviors. For each group, we calculate the number of accounts/spammers attracted by per action with the following equations:

$$A_{apa} = \frac{M_{accounts}}{N_{action}}, \quad A_{spa} = \frac{M_{spammers}}{N_{action}} \tag{3.2}$$

A_{apa} (A_{spa}) is the number of accounts (spammers) attracted by per action. $M_{accounts}$ ($M_{spammers}$) is the total number of accounts (spammers) attracted by each group. N_{action} is the total number of actions of a group. The result is presented in Fig. 3.14. We can find that following others actively (group E) is the most effective approach to attract interactions from other accounts. Each following action can attract 0.15 account. However, following action cannot attract spammers effectively since every following action can only attract 0.0016 spammer. The same as following action, mentioning (group B) unrelated accounts is efficient in attracting accounts (0.1175 accounts per action) but not efficient in attracting spammers (0.0073 per action). In contrast, posting tweets with sensitive keywords (group D) can attract both accounts and spammers efficiently, each of its action can attract 0.1213 accounts and 0.0276 spammers. Retweeting (group C) is the least efficient way to attract accounts since each retweeting action can only attract 0.029 accounts and can attract few spammers. As a summary, most attractive behaviors can attract interactions from other accounts efficiently. However, only posting keywords in tweets can attract spammers efficiently. As for the control group (group A), which did nothing and only wait for unsolicited interactions from other accounts, it only attracted 6 accounts in total. While even the least efficient behavior − retweeting can attract more than 300 accounts and 10 spammers in total. The traditional honeypots on Twitter are just like group A, so we can see they are no longer useful for attracting spammers.

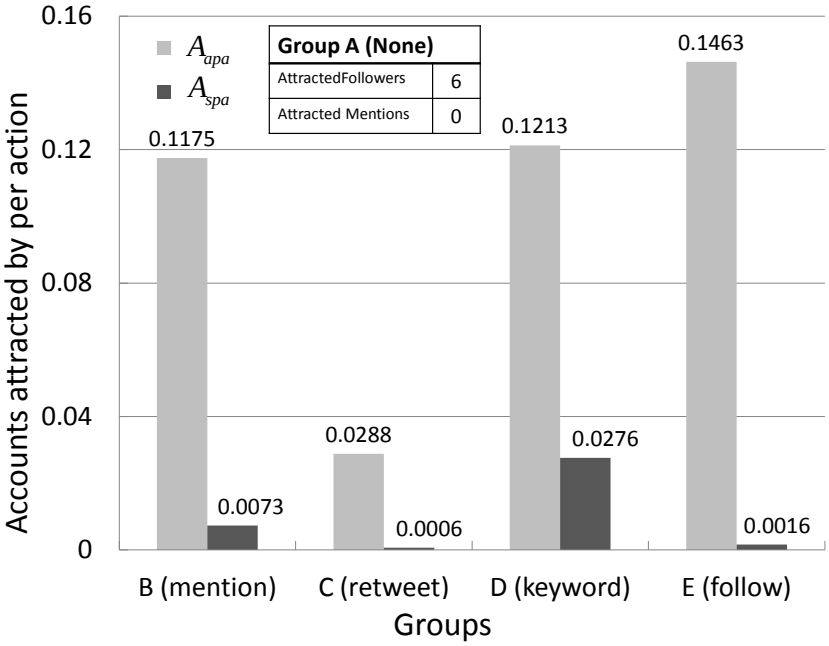

Fig. 3.14 Accounts/Spammers attracted by per action

Secondly, we measure the ratio of accounts attracted by actions directly for each group. If an account attracted by our groups is a target of an action, we say this account is attracted by the action directly. Here a target of an action means a user mentioned by an account in group B, a user retweeted by an account in group C or a user followed by an account in group D. In Fig. 3.15 we present how many accounts are attracted by the actions directly. We can find that 86% accounts attracted by group D are attracted by the following actions directly. For retweeting action the ratio is 50%, while for mention action this ratio is only 32%. This is because following action can only be noticed by the target account itself and can only lead to following back from target account. If you mention a target account, however, the tweet which containing this mention will appear on the timeline of the target account and can be seen by all his followers. And if you retweet a tweet, you profile image will be displayed under the original tweet and can be seen by all the followers of the poster. So the accounts attracted by the mention action and retweeting action indirectly can be the followers of the targets.

Thirdly, we measure the ratio of two kinds of interactions, following interaction and mention interaction, received by each group. We present the result in Fig. 3.16. For group D, about 75% interactions it received are mention interactions. While for group

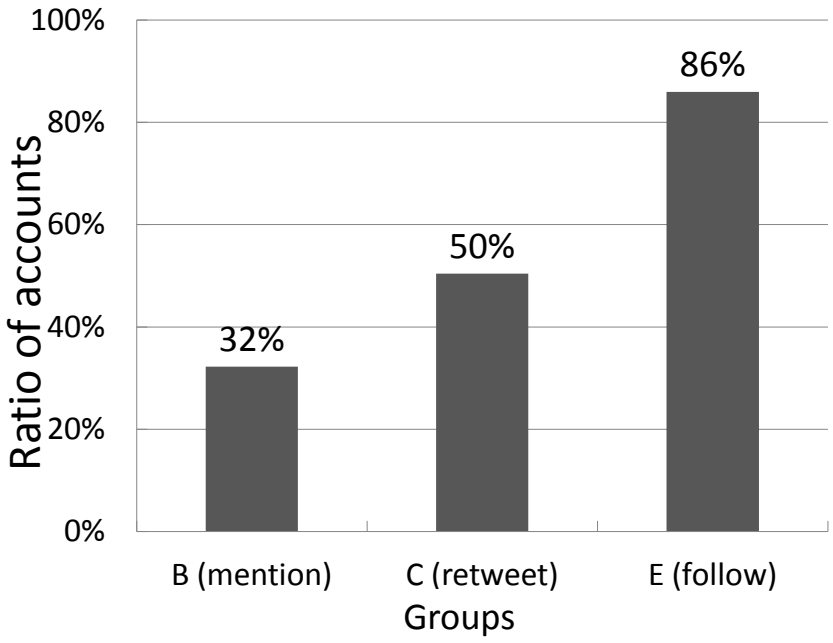

Fig. 3.15 Ratio of accounts attracted by action directly

E, fewer than 5% interactions are mention interactions. This is because spammers use Twitter search to find tweets which contain certain keywords and directly reply to these tweets with spam content rather than following the posters. But for an active following from other accounts the straight forward reaction is following back rather than replying to a tweet. For group B, there are about 40% mention interactions and 60% following interactions, this demonstrates that mention action can lead to both reply or following back from other accounts. And for retweeting action (group C), the target account prefer following back rather than mentioning our accounts.

To measure whether the number of followers of an account will influence its attractiveness, we compare the number of accounts attracted by group G and F. The result is shown in Fig. 3.17. According to the result, influential accounts (group G) can be more attractive than normal accounts under the same condition. Especially in attracting followers. This is because an user will have a judgement before they follow back a strange follower. Obviously, influential accounts are prone to be thought as high reputation users and more likely to be followed back. This further proves our assumption in section 3.3.2. That is though many random influential accounts have high FB-ratio, they avoid following back spam-like accounts. However, group G and F do not have much difference on the mention

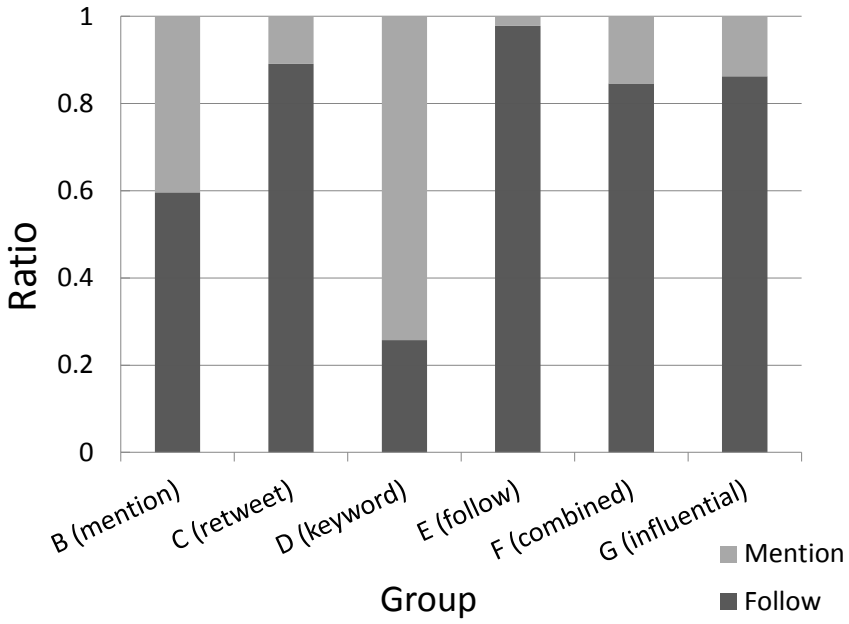

Fig. 3.16 Ratio of two types of interactions for each group

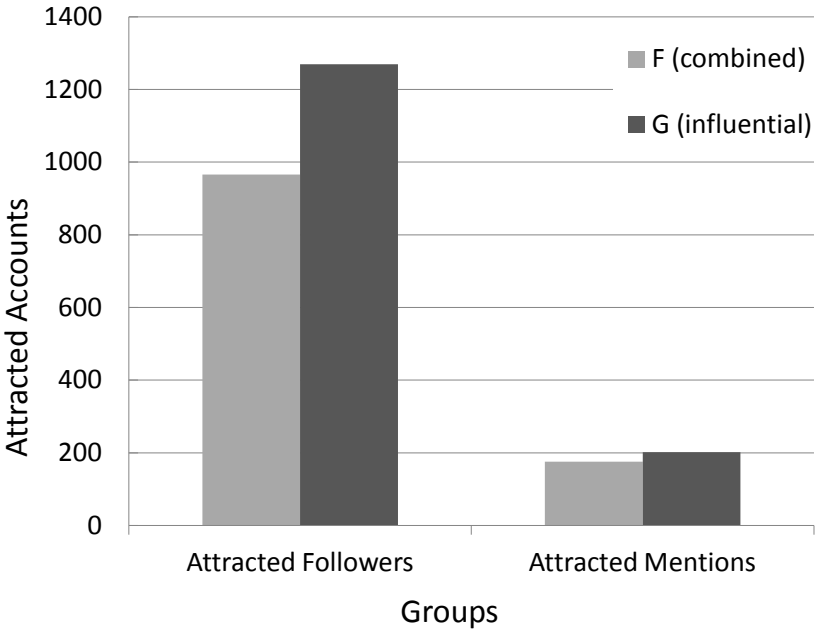

Fig. 3.17 Accounts attracted by group F and G

interactions they attracted. This may be because a user directly reply to an unsolicited mention rather than browse the home page of the sender and have a judgement first.

In our experiment, many of our experiment accounts were suspended by Twitter. In table 3.5 we list the number of suspended accounts in each group. The suspended accounts mainly belong to group F and G. Accounts in these two groups do all the potential attractive behaviors in our experiment. So we can say combining different behaviors to attract accounts on Twitter will lead to account suspension. However, fewer accounts were suspended in group G than in group F. This demonstrates that influential accounts have a lower probability to be suspended by Twitter than normal accounts under the same condition. Besides group F and G, group B also has 6 accounts being suspended by Twitter. Mentioning other unrelated accounts leads to account suspension can be due to spam report from those accounts who received unsolicited mentions. As for 3 suspended accounts in group D, we reviewed the tweets posted by these accounts, we found that there were spam content among them. This is because we used Twitter search to search related tweet which contain spam-sensitive keywords as our textual templates. There was spam content in the search results.

Table 3.5 Number of suspended accounts in each group

Group Name	Suspended Accounts
Group A	0
Group B	6
Group C	0
Group D	3
Group E	0
Group F	28
Group G	20

3.4.4 Conclusion

Based on the experiment results above, we can answer the question we raised at the beginning of this section: can an account become attractive to spammers by imitating these behaviors? According to our result, it is not necessary. Some behaviors such as following others actively and mentioning unrelated accounts can help accounts to attract following

back and mentioning from other accounts. However, these behaviors can not make our accounts attractive to spammers. We guess there can be three reasons for the low efficiency in attracting spammers for these two behaviors: (i) the experiment period is not long enough, only a few spammers found our accounts offered such kind of service; (ii) our accounts are not influential enough to attract spammers; (iii) there are offline money transactions between spammers and those active honeypots. Posting tweets with sensitive keywords, which are tempting to spammers, is the most effective approach to attract spammers in our experiment. It can attract about 3 spammers with every 100 tweets. According to the Twitter rules [51] , an account can at most post 2,400 tweets a day. So we estimate that it can attract at most 72 spammers a day. The spam ratio among all the account attracted by this behavior is 0.23. About 75% interactions attracted by this behavior is mention interaction as well as one third of the attracted accounts are our direct targets. But posting tweets with sensitive keywords has the potential danger of being suspended by Twitter. Becoming a influential account can lower the probability of being suspended and enforce the attractiveness to spammers.

Chapter 4

Identify Active Honeypots

In this chapter, we first introduce the system overview of our active honeypot based spammer detection system. Then we give an detailed presentation about the design of active honeypot identifier which is used to identify effective active honeypots from billions of Twitter accounts. The identifier is composed of three stages of ranking: graph based ranking, feature based ranking and history based ranking. At last we evaluate the performance of the active honeypot identifier.

4.1 System Overview

Fig. 4.1 shows the structure of our active honeypot based spammer detection system. The system iterates between two components. *The first one* is active honeypot identifier (step 1-4 in Fig. 4.1). This component can be further divided into three sub-components: graph based ranking (step 1-2 in Fig. 4.1), feature based ranking (step 3 in Fig. 4.1), and history based ranking (step 4 in Fig. 4.1). The inputs for active honeypot identifier are the spamming accounts collected in the past. For each spamming account, we collect its friends, followers and mentioned users. Initially, we use the accounts suspended by Twitter as the spamming accounts. Then we build three relation graphs based on the collected data. Graph based ranking algorithm is applied to the three relation graphs to rank all the accounts interacting with social spammers. In addition, we use feature based ranking and history based ranking to refine the ranking scores of graph based ranking, to more consistently put active honeypots to the top ranks.

The second component is the spammer detector (step 5-7 in Fig. 4.1). The spammer detector finds spamming accounts among all the ones interacting with active honeypots, by using an enforced spam classifier based on active honeypot based features. The active honeypot based features and spammer detector will be discussed in the chapter 5.

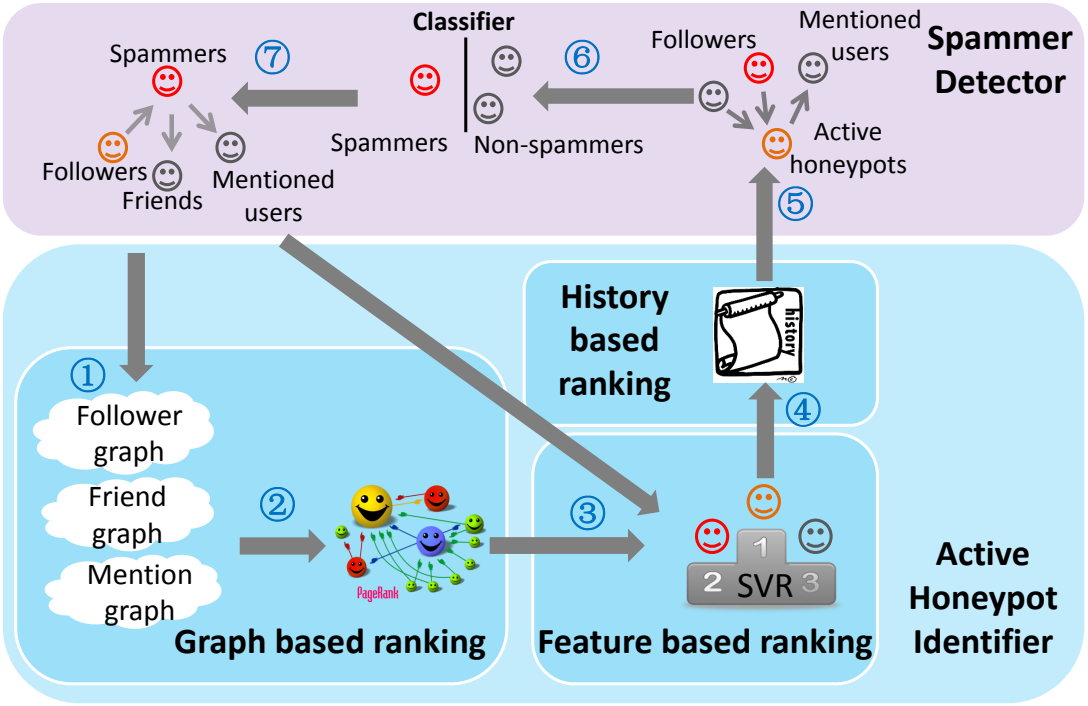

Fig. 4.1 Structure of active honeypot based spammer detection system

4.2 Active Honeypot Identification

The goal of active honeypot identifier is to rank all the accounts interacting with spammers by their attractiveness to spammers. We define the metric of attractiveness as:

$$spam\_num * spam\_ratio^{\beta} \tag{4.1}$$

The intuition of this equation is only if an active honeypot can attract a large number of spammers as well as has a high spammer ratio, it is really attractive to social spammers. where β is a configurable parameter controlling the trade-off between the number of spammers ($spam\_num$) and the ratio of spammers ($spam\_ratio$) that can be trapped

by an active honeypot. In our design, we set $\beta = 0.5$ which is empirically configured to ensure that the active honeypots we obtained can not only trap a large number of social spammers but also with a high spammer ratio.

4.2.1 Graph Based Ranking

As in our observation in section 3.3, spammers interact with active honeypots mainly in the following three ways: (1) spammers follow active honeypots; (2) spammers are followed back by some active honeypots; (3) spammers *like to mention (reply mention)* some active honeypots. Based on these three types of interaction, we construct (1) a *friend graph* ($\mathbf{G_{fr}}$), based on the friend relationship; (2) a *follower graph* ($\mathbf{G_{fo}}$), based on the follower relationship; and (3) a *mention graph* ($\mathbf{G_m}$), based on the mention relationship. The construction of the three relation graphs are similar, where the nodes represent the users, and the directed edges represent the existence of the corresponding relation. In all three relation graphs, we regard all the links with spammers as outbound links, which means no matter an account follow, followed by or mentioned by a spammer we regard there is an outbound link from spammer to this account.

After constructing the graphs, we apply TrustRank [53] on the graphs to calculate the ranking score for each user, as detailed in Algorithm 1. TrustRank is a robust ranking algorithm, which is similar to Google's PageRank algorithm [30]. One primary difference between PageRank and TrustRank is that TrustRank needs to select some nodes as seeding nodes which are given a certain credits. In our implementation, we take all the spamming accounts collected in the past as the seeding nodes (i.e. *Spam* in Algorithm 1). Each seeding account is initialized to 1 while other accounts to 0 (Line 1 in Algorithm 1). In the procedure of TrustRank (line 7-13 in Algorithm 1), the ranking score of a node n is calculated similarly to the ranking score computation in PageRank (the damping factor α is set to the commonly used value, 0.85). As we can see, the ranking score $c(n)$ of node n is mainly determined by the score of its incoming neighboring nodes (i.e., $incoming(n)$). Therefore, the more node n interact with spamming accounts, the higher it will be ranked. In order to combine the ranking scores obtained on the three different relation graphs, we first normalize the ranking scores on each graph by its percentile. For example, if an account is ranked as the second best among 1000 accounts, its ranking score will be normalized to $(1 - 2/1000)$. After ranking score normalization, the final ranking score of each node is

the highest value among the three ranking scores obtained from the three relation graphs respectively.

Algorithm 1: Graph Based Ranking

Input : relation graphs $\mathbf{G_{fr}}, \mathbf{G_{fo}}, \mathbf{G_m}$, spammer set **Spam**, decay factor of TrustRank α ;
Output: ranking score: c

initialize score vector d for all nodes n in relation graphs $d(n) = \begin{cases} 1 & \text{if } n \in \textbf{Spam} \\ 0 & \text{otherwise} \end{cases}$
$c_{fr} = \texttt{TrustRank}(\mathbf{G_{fr}}, d, \alpha)$;
$c_{fo} = \texttt{TrustRank}(\mathbf{G_{fo}}, d, \alpha)$;
$c_m = \texttt{TrustRank}(\mathbf{G_{fr}}, d, \alpha)$;
$c = \texttt{Maximum}(c_1, c_2, c_3)$;

procedure `TrustRank`$(\mathbf{G}, d, \alpha)$
  $c \leftarrow d$;
  while *c not converged* **do**
    for *all nodes n in* **G do**
      $tmp = \sum_{nbr \in incoming(n)} \frac{c(nbr)}{\|outgoing(nbr)\|}$;
      $c(n) = \alpha \cdot tmp + (1 - \alpha) \cdot d(n)$;
  $c \leftarrow$`Normalize` (c);
  return c;

4.2.2 Feature Based Ranking

As shown in Fig. 4.2, although graph based ranking ranks a large portion of active honeypots to the top, it still ranks a significant fraction of active honeypot accounts to the second half of the ranking. For instance, some legitimate users who are targeted by accidental spammers' aggressive friending will be ranked to the top. To further refine the ranking, we exploit additional features pertaining to each account. Based on the comparative studies of section 3.3, we consider some additional features extracted for each account. These additional features combined with the final ranking score of the graph based ranking are the input features for our feature based ranking. All the features we used in feature based ranking are listed in Table 4.1.

We use Support Vector Regression (SVR) [54] to learn a ranking function for predicting accounts' attractiveness to social spammers. To train this ranking function, we randomly sample $1,500$ accounts and extract their features. These features are used as training data. The training labels are obtained as follows. We first calculate the attractiveness for all the

training accounts by checking the average daily spammer number and average spammer ratio among new followers for three days. Then we rank all the accounts in the training set and normalize their ranking scores. The normalized ranking scores are used as training labels. Based on the training data and training labels, we apply SVR to learn the function for attractiveness prediction.

Table 4.1 Features used in feature based ranking

Feature name	Description
FF-ratio	Friend number vs. follower number
FN	Follower number
UM	ratio of unrelated users in all mentioned users
SM	ratio of suspended users among all mentioned users
R-Score	Ranking score in TrustRank
DPF	Ratio of default profile image setting among followers
TF	Average tweets number of followers

4.2.3 History Based Ranking

To further improve the ranking performance, we exploit historical information of each account in trapping spammers. The reason is that the interaction intensity between active honeypots and spammers might fluctuate. For example, an active honeypot might stop interacting with active honeypots for several days after participating in a spam campaign. This is in particular true for those active honeypots who buy a large number of followers from spamming organizations in a specific time period. In order to avoid attention, this time period usually takes several weeks or even several months. After such a time period, these accounts may not interact with new spammers anymore, which means that they are unable to trap new spammers afterwards, even though they had a large interactions with spammers.

In our history based ranking, we combine the historical attractiveness scores with the predicted scores obtained in feature based ranking. We normalize the historical attractive scores in the same way as normalizing the ranking scores in TrustRank, and the historical attractiveness score of an account is simply taken as the average of its attractiveness scores in the last three days. Then we combine historical attractive scores (c_h) with the predicted

attractiveness scores (c_f) of feature based ranking as follows:

$$(1 - \gamma) * c_h + \gamma * c_f \tag{4.2}$$

where γ is a configurable parameter controlling the trade-off between feature based attractiveness scores and historical attractiveness scores. In our implementation we empirically set $\gamma = 0.7$.

4.3 Evaluation of Active Honeypot Identifier

We randomly select 10,000 accounts from all the suspended accounts we collected as seeding accounts. 1,024,856 related accounts which either follow, or are followed by, or are mentioned by the seeding accounts. Note that our active honeypot identifier only uses the suspended account information in the the initial stage. To evaluate the effectiveness of this component, we first record the number and ratio of suspended accounts which interact with these 1,024,856 accounts each day for a period of 9 days. Then we calculate the attractiveness scores using equation (4.1). The attractiveness scores are used as the ground truth for ranking all the 1,024,856 accounts. Fig. 4.2 shows the ranking performances using different kinds of information.

Friend graph based ranking: Using only friend graph, 24% top attractive accounts are ranked within the top 0.1% positions and about half of them stay in the last 80%. The reason for unsatisfactory performance is that popular accounts which are followed by many spamming accounts are inclined to be ranked to the top even though the ratio of social spammers in their new followers is very low.

Mention graph based ranking: Using only mention graph, 11% top attractive accounts are ranked within top 0.1% positions and 30% top attractive account stay in the last 80% positions. Mention graph based ranking can only push those active honeypots which frequently being mentioned by spammers in tweets to the top positions.

Follower graph based ranking: Using only follower graph, 38% top attractive accounts are ranked within the top 0.1% positions, which is better than friend graph and mention graph. The reason for improved performance is that spammers are mostly followed by either other spammers or active honeypots. Legal accounts seldom follow spammers. In this case, many active honeypots are ranked to the top positions. However, there are

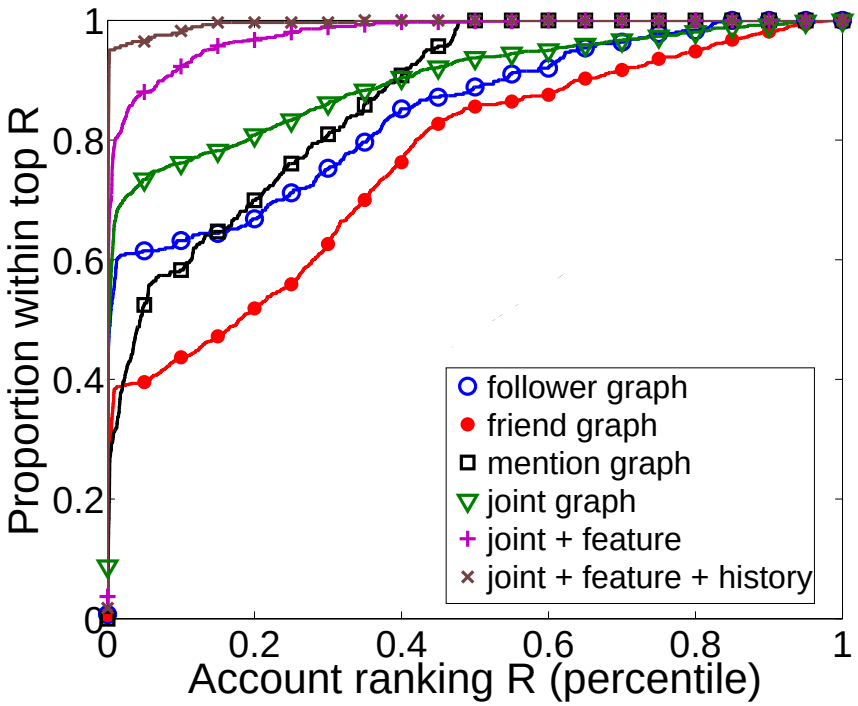

Fig. 4.2 Ranking of attractive accounts

still 33% top attractive accounts rank in the last 80% positions. This is because some attractive active honeypots do not follow back spammers following them. In this scenario, these active honeypots can be hardly ranked to the top positions.

Joint graph based ranking: In this configuration, we use the combined ranking scores based on the 3 relation graphs. 42% top attractive accounts are ranked within top 0.1% positions, which is better than any individual relation graph. By combining results from the 3 relation graphs, active honeypots using different ways to contact with spammers will be ranked to the top positions. However, due to the limited number of spammers in the seeding account set, some top attractive active honeypots have only few related spammers to improve their ranking.

Joint + feature based ranking: By combining graph based ranking scores and some additional features, about 62% top attractive accounts are ranked within the top 0.1%, and only 5% of them stay in the last 80%. The reason for such great improvements is that additional features can increase the graph based ranking scores for accounts similar to attractive honeypots while decrease the graph based ranking scores for accounts dissimilar to attractive honeypots.

Joint + feature + history based ranking: By further introducing the historical information about each account's attractiveness, our active honeypot identifier can rank 80% top attractive accounts within top 0.1% positions and 95% within the top 0.27% positions.

After three stages of ranking, we regard top 0.1% accounts in the final ranking results as active honeypots and add these accounts into an active honeypot pool. We use active honeypots in the pool to trap spammers. We will record the number of spammers and ratio of spammers trapped by each active honeypot in the pool. According to the number of spammers and spam ratio of each account in history, we will eliminate those ineffective active honeypots from the pool. Besides, we will use the new spammers trapped by our system as seeding accounts to identify new active honeypots with our identifier periodically. Then we will add these new identified active honeypots into the pool.

Chapter 5

Detection of Spammers with Active Honeypots

In this chapter, we introduce a new kind of features named active honeypot based features. We build an enhanced spammer detector with this new kind of features. We evaluate the contribution of this new kind of features to performance and the discrimination power of the features. In addition, we will discuss the performance of the classifier with different machine learning algorithms and the influence of unbalanced dataset on performance. At last, we will evaluate the overall performance of our active honeypot based spammer detection system.

5.1 Active Honeypot Based Features

A new kind of features named active honeypot based features is introduced in our spammer detector. The active honeypot based features are listed in table 5.1. The intuition behind active honeypot based feature is quite simple. If an account interacts with an active honeypot which attracts a high proportion of spammers, this account is probably a spamming account. If an account interacts with many active honeypots, then this account is also probably a spamming account, If the new followers of an active honeypot are densely created within a short time period, these followers are probably spamming accounts. If an active honeypot follow back an account, the account is probably not a fake follower. Based on the intuitions, five properties are used in active honeypot based features for accounts interacting with active honeypots. The first one is the ratio of spamming accounts

interacting with an active honeypot in the last time period (ASR). The second one is the average daily new follower number of an active honeypot (DFN), this can be extremely high for whom buy fake followers. The third one is how many active honeypots an account interacts with (AIN). The forth one is the number of common active honeypot accounts created in the same time window (CIW). The last one is whether an active honeypot follow back an account (AFB). We calculate the first, second and forth properties for each active honeypot and the third, fifth one for each account interacting with active honeypots.

Table 5.1 Active honeypot based features (AFeat)

Abbreviation	Feature Description
ASR	Ratio of spamming accounts interacting with an active honeypot
DFN	Average daily new followers of an active honeypot
AIN	Number of active honeypots a user interact with
CIW	Number of common active honeypot accounts created in the same time window
AFB	Is active honeypot follow back this account?

5.2 Experiment Setup

From the accounts trapped by our active honeypots in a day, we randomly selected 8,000 accounts. With URL blacklisting, Twitter suspension service and manual labelling we finally identified 2679 spammers and 4410 legitimate users. The other 911 accounts were difficult to judge or lack of complete profile information so we left them out. For the total 7,089 spammers and legitimate users, we extracted traditional features and active honeypot based features for each of them. The traditional features refer to other profile or tweet based features which were used in previous works. We list all the traditional features we adopted in table 5.2. The detailed definition and calculation of traditional features is listed in Appendix A. We used weka [55] to train the classifier and evaluate the result with 10-fold cross validation.

Table 5.2 Traditional features (TFeat)

Abbreviation	Feature Description
FLN	Number of followers of an account
FF-ratio	Friend Number / follower Number
AA	Age of an account in Days
TWN	Number of tweets posted by a user
DPI	Use default profile image?
FAM	Follower fame of an account
MR	Ratio of mentions in tweets
HR	Ratio of hashtags in tweets
UR	Ratio URLs in tweets
TWI	Average time interval of tweets
RT	Average retweet count
FT	Average favorite count
SN	Different sources of tweets
TS	Average tweets similarity

5.3 Performance Evaluation

We first evaluate whether active honeypots based features can improve the performance of spam detector. In Fig. 5.1 we present the performance of our spammer detector trained with traditional features, active honeypot based features and two sets of features together, respectively. We can find that traditional features are no longer powerful. The accuracy is only a little better than 0.8 and the false positive rate (FP rate) is as high as 0.13. In spam detection the FP rate is the most important metric, since we cannot misclassified an legitimate user as an spammer. So the performance with traditional feature set cannot meet the requirement of practical using. If we only use active honeypot based features, it achieves lower FP rate but get low recall. The low recall means our detector can only cover a small portion of the total spammers. After we combine traditional feature set with active honeypot based feature set, we can achieve an accuracy of 0.93, a recall of 0.94 and a false positive rate of 0.07. The accuracy and recall are much better than simply using the other two feature sets independently. Though the FP rate is higher than simply using active honeypot based feature set, we can modify the threshold to make a trade-off between FP rate and recall. We will do this threshold tuning in the following part of this section. All

in all, active honeypot based features can improve the performance of feature based spam detector obviously.

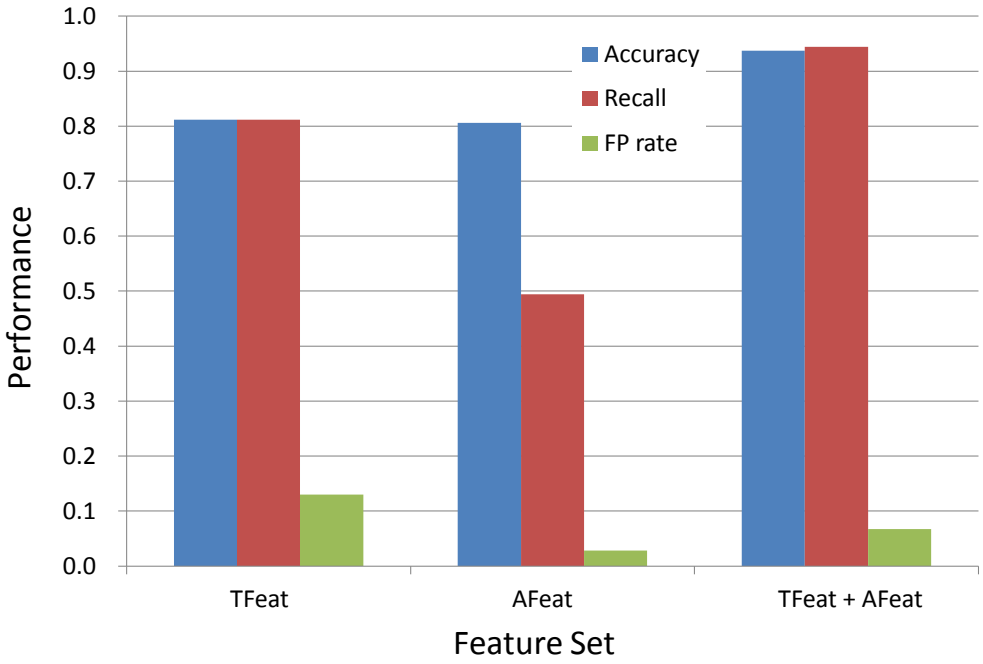

Fig. 5.1 Spammer detector performance with different feature set

To measure the discrimination power of each feature, we plot the Receiver Operating Characteristic (ROC) Curve for each feature. The X axis of ROC curve is FP rate, the Y axis of ROC curve is true positive rate (TP rate). The closer the ROC curve is to the upper left corner, the more discrimination power the feature has. The result is presented in Fig. 5.2. We can find that the feature CIW, AIN and DFN, which we introduce in active honeypot based feature set, are all strong discriminative features. Many traditional features which once thought to be effective in distinguish spammers from legitimate users such as TS, FAM and TWI are no longer powerful. This is because spammers are keeping adjusting their strategies to fight against anti-spam system. However, some traditional features which are difficult to fabricate are still powerful such as FF-Ratio. After all, it is really hard and expensive for spammers to get many followers.

In table 5.3, we present the performance of our spammer detector with different machine learning algorithms. Because Twitter does not reveal all the information of a user and Twitter suspension service may reply on some non-public information to identify spammers.

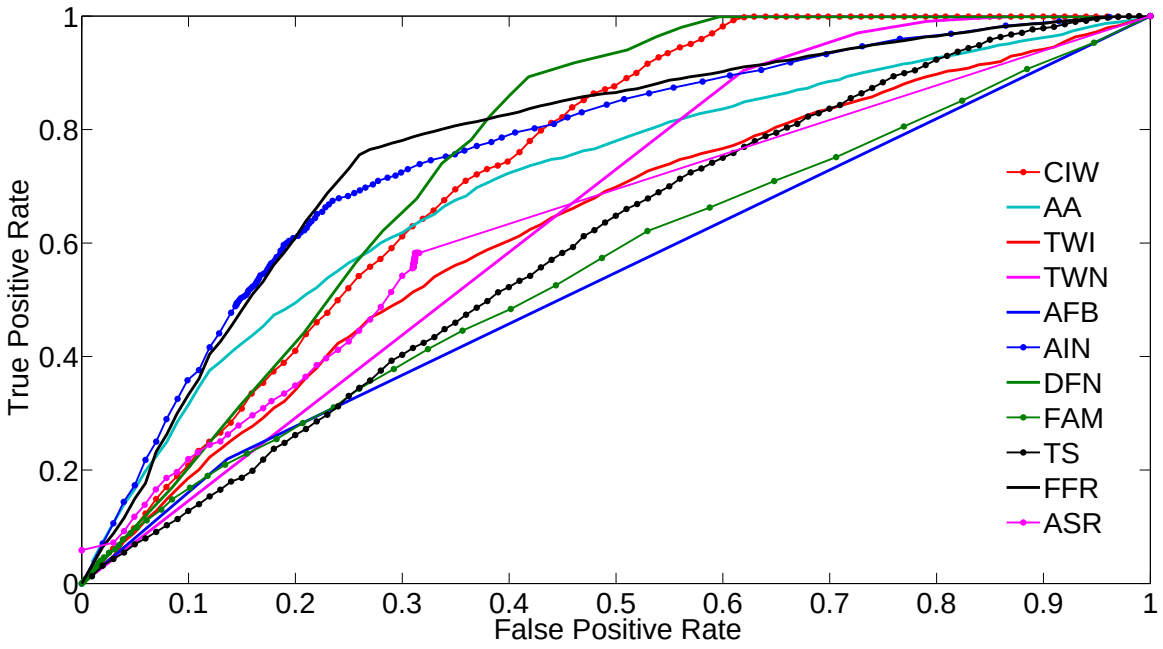

Fig. 5.2 ROC curve of each feature

So some spammers cannot be separated from legal users with the information we can get from Twitter. So our dataset is actually a inseparable dataset. According to table 5.3, ensemble learning and decision tree work better than linear classifier for an inseparable dataset. J48 tree works the best among all these algorithms.

Table 5.3 Performance of spammer detector using different algorithms

Classifier	FP Rate	Recall	Accuracy
SVM(SMO)	0.058	0.821	0.821
Logistic Regression	0.090	0.822	0.822
J48 Tree	0.030	0.966	0.966
Bagging	0.037	0.970	0.970
AdaBoost M1	0.067	0.937	0.937

Our spammer detector needs to detect spammers from all the accounts trapped by our active honeypots, so the spam/legitimate ratio of our dataset can be quite different from the dataset that crawled from Twitter's public stream. Generally, the spam/legitimate ratio of Twitter's public stream is smaller than 1:5, while among the accounts trapped by

our active honeypots this ratio can be as high as 1:2. We present the performance of the spammer detector under different spam/legitimate ratio in Fig. 5.3. We can find that high spam/legitimate ratio of a dataset will cause FP rate to improve. As a trade-off, the recall also improve. As we have mentioned, for a spammer detector the FP rate is more important than the recall. In other words, we would rather miss some spammers than misclassified a legitimate user as a spammer. So we need to tune the threshold for our classifier. In Fig. 5.4 we present the ROC curve for our classifier. The original optimized result of weka is point A which has a FP rate of 0.049 and a recall of 0.913. After adjusting the threshold, we get the optimized result in point B which has a FP rate of 0.013 and a recall of 0.842. Now, only one among one hundred Twitter users will be misclassified as spammer by our spammer detector which is about 4 times better than the unoptimized detector.

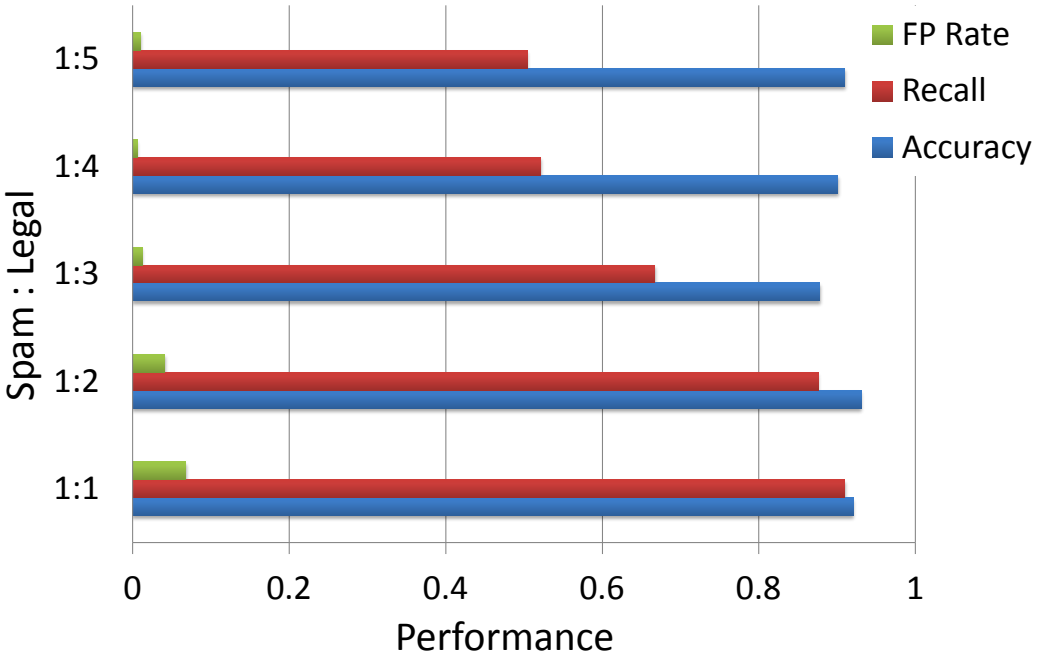

Fig. 5.3 Performance of spammer detector with unbalanced dataset

5.4 Error Analysis

In table 5.4 we list the confusion matrix of our spammer detector. There are 55 legitimate users misclassified as spammers and 423 spammers misclassified as legitimate users. In

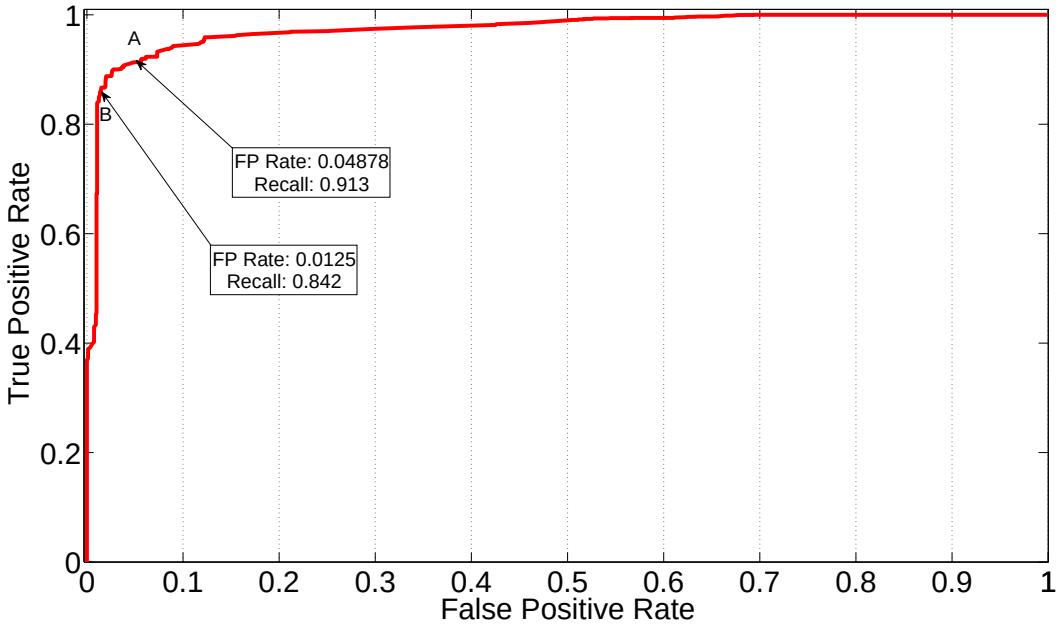

Fig. 5.4 ROC curve of spammer detector with 1:2 spam/legitimate ratio unbalanced dataset

order to figure out why our spammer detector cannot classify these accounts correctly, we checked the profiles and tweets of these accounts manually. According to our observation these errors may be caused by the following reasons:

There are two main reasons leading to legitimate users misclassified as spammers. The first one is some legitimate users are inactive on Twitter, they have few tweets and few followers. They are quite similar with those fake accounts. The second reason is that some legitimate users also follow many active honeypots, they may expect active honeypots to follow back or retweet for them. We also find some active honeypots are porn accounts, these legitimate users may be attracted by the porn content posted by these active honeypots.

There are three possible reason leading to spammers misclassified as legitimate users. The first reason is that some spamming accounts are created elaborately, they have complete account profile, a number of fake followers, post spam content mixed with legitimate content and had been stockpiled for a long time before taking part in spamming activities. These elaborately created spammers are really difficult to detect. However, due to the high cost to build and maintain these accounts, the number of these kind of spammers is quite small.

In addition, we found some accounts were suspended by Twitter, but we could not find any spam characteristics from their profiles as well as tweets. We believe Twitter rely on some non-public information to identify them as spammers such as their login IP address, register pattern and login pattern. The last reason is that some accounts were suspended by Twitter after we crawl their account profiles and tweets. They may post spam content in the time gap between our crawling and Twitter suspension. So we didn't catch their spam behaviors.

Table 5.4 Confusion matrix of spammer detector

		Actual	
		Spam	Legitimate
Predict	Spam	2256	55
	Legitimate	423	4355

5.5 Overall Performance of the Active Honeypot Based Spammer Detection System

To evaluate the overall performance of our active honeypot based spammer detection system, we use the optimized spammer detector we trained above to detect spamming accounts which interact with the active honeypots for one week.

Due to the rate limitation on Twitter API usage, we are unable to obtain complete information for more than $160,000$ accounts interacting with the $1,819$ active honeypots that we identify simultaneously. Instead, we only obtain friend lists, follower lists, and 200 recent tweets for about $21,000$ accounts interacting with 200 randomly sampled active honeypots everyday.

We apply our spammer detector to the $21,000$ accounts we obtained each day. Since it is too expensive to manually label all the $21,000$ accounts, we use the following strategies to measure the classification performance. To estimate the FP rate, we randomly sampled 400 accounts which are predicted as spammers by our detector and then check how many of them are real spamming accounts according to Twitter rules [51]. To estimate how many spamming accounts can be covered by our detector (recall), we check how many

accounts suspended by Twitter can be detected by our detector. We wait one month to check whether the accounts are suspended by Twitter.

As shown in table 5.5, our spammer detector can achieve a FP rate of 0.019 and a recall of 0.6 in average. There are two reasons why our spammer detector can only cover 60% of all the spamming accounts suspended by Twitter. First, according to [16], the spam detection system of Twitter uses some private information such as the registration IP and logging patterns, which are unavailable for our spammer detector. By examining those accounts missed by our spammer detector, we found that some of these accounts suspended by Twitter even still had not posted any tweets. Second, it is also possible that the spamming activities of these accounts had not been caught by our system because we crawled each account only once everyday. Though our spam detector misses about 40% spammers which were suspended by Twitter, we want to emphasize that we can detect 3 *more times of spammers which escape from the spam detection system of Twitter in early age*. According to our statistics, on the first day these accounts interacting with active honeypots, less than 10% of them can be identified by Twitter as social spammers. After one month, about 25% of them are suspended. While our spammer detector can identify about 26.6% accounts as spammers with low FP rate on the first day they interact with active honeypots. If there is no API limitation set by Twitter, we can identify about 40,000 spammers with our 1,819 active honeypots every day. According to a report [13] in 2012, the spammers we identified possess about 4% of Twitters' daily new registers. So our system can be a strong compliment to current Twitter spam detection system. By examining those accounts missed by the detection system of Twitter, we found that most of these spamming accounts demonstrate very obvious spamming behaviors. For example, many of these accounts are created in a very short period (e.g., 10 minutes) and share similar user name, profiles and follow the same active honeypots. Fairly a portion of them shared at least one active honeypots and posted the same URLs linking to pharmacy advertisements or pornographic websites.

Table 5.5 Over all performance of active honeypot based spammer detector system

Day	1	2	3	4	5	6	7
FP Rate	0.020	0.018	0.021	0.019	0.018	0.022	0.019
Recall	0.60	0.56	0.62	0.61	0.57	0.64	0.58

Chapter 6

Conclusions

6.1 Conclusion

In this thesis, we reveal that there exists a specific type of accounts on Twitter, termed as active honeypots, which attract about $8,000$ times more social spammers per day than manually created honeypot accounts proposed in previous works.

First, we reviewed the spam problem on social network, we focused on the motivation and basis of spamming activities on OSNs. Then we discussed six categories of spam detection strategies on OSNs. For each kind of strategy, we listed the main related works as well as discussed their advantages and limitations.

Then we proposed our active honeypot based spam detection strategy. Before designing the detection system, we first collected active honeypots from Twitter and analyse the properties of active honeypots. We found that active honeypots were efficient in trapping spammers especially compared with passive honeypots in previous works. Active honeypot were all influential accounts on Twitter and they did not reveal any spam behaviors so that they could hide behind other influential accounts to prevent being suspended by Twitter. After analysis of properties, we further investigated the attractiveness of active honeypots. According to our investigation the accounts interacting with active honeypots were mainly fake or low reputation accounts. We identified four potential attractive behaviors of active honeypots. Based on these behaviors, we proposed five potential rewarding mechanisms which made active honeypots attractive to social spammers. To further learn about the potential attractive behaviors of active honeypots, we created accounts to imitate these behaviors. According to the experiment results, these behaviors themselves are not nec-

essarily able to make accounts attractive to spammers. Some other factors may also be needed for becoming an active honeypot.

After the analysis of active honeypots, we designed our active honeypot based spammer detection system. We first designed an active honeypot identifier to identify effective active honeypots from billions of Twitter accounts. Our identifier contained three stages of ranking: graph based ranking, feature based ranking and history based ranking. After three stages of ranking, we could rank 80% top attractive accounts within top 0.1% positions and 95% within the top 0.27% positions. We successfully identified 1000 active honeypot from about 1 million accounts.

Based on active honeypots, we proposed a new kind of features named active honeypot based features. We built an enhanced spammer detector with active honeypot based features and compared it with traditional feature based spammer detector. The results showed that active honeypot based features did improve the performance of spammer detector. We also evaluated the discrimination power of each feature as well as optimized the spammer detector under an unbalanced dataset. At last we evaluated the overall performance of our active honeypot based spammer detection system. Our system can achieve a FP rate of 1.9%. With 1,819 active honeypots we can trap about 40,000 social spammers on Twitter every day which is about 4% of the daily new registered Twitter users.

6.2 Future Works

Two aspects of our current work can be improved in the future.

First, as we have mentioned in section 3.2.1, many followers of active honeypots are suspicious to be fraudulent accounts. These accounts are closely created in time and quite similar in profile settings so they may belong to the same spam campaign. In addition, in section 3.3.6 we analyse that some active honeypots can be follower buyers, their followers may be bought from the same spam campaign. Based on the above two points, we think we may use cluster based strategy to indentify spam campaigns from accounts interacting with active honeypots. Especially, previous spam campaign detection approaches were all realized on tweet level and only adopted URL contained in tweets and text similarity to define the similarity of two tweets. With the help of active honeypots, we may realize spam campaign detection on account level. We may introduce account create time and the time of following a certain active honeypot and the active honeypots shared by two accounts to

define the similarity of two accounts.

Cooperation with Twitter can be another potential improvement for this thesis. As we mentioned in section 5.5. The low recall of our spam detection system is mainly due to the lack of complete account information such as login IP address, register pattern and login frequency. If we can cooperate with Twitter and extract new features from these information, we believe the performance of our system can be improved significantly.

Appendix A

Traditional Features for Feature Based Spam Detection

In this section, we list the detailed definition and calculation of traditional features which are used in features based spam detection on Twitter by previous works.

Follower number (FLN): The number of followers of an account.

FF-ratio: The ratio of friend number divides follower number of an account. If the follower number of an account is 0, we set his FF-ratio as 100.

Account age (AA): The age of an account, count in days from the created time of an account to the date we extract this feature.

Tweet Number (TWN): The number of tweets posted by an account till the date we crawl its profile.

Default profile image (DPI): If an account do not upload a specified image for its profile, Twitter will set a default profile image for it. If an account use default profile image we set its DPI as 1, otherwise we set it as 0.

Follower fame (FAM): We regard the follower number of an account as its fame [11]. The follower fame of an account is the average fame of its followers.

Mention ratio (MR): The ratio of mentioned users in tweets among all the tweets of an account. MR can be calculated by the following equation:

$$MR = \frac{M}{C_{tw}} \tag{A.1}$$

M is the total number of mentioned users in tweets. C_{tw} is the total number of tweets.

Hashtag ratio (HR): The ratio of hashtags in tweets among all the tweets of an account. MR can be calculated by the following equation:

$$HR = \frac{H}{C_{tw}} \tag{A.2}$$

H is the total number of hashtags in tweets.

URL ratio (UR): The ratio of URLs in tweets among all the tweets of an account. UR can be calculated by the following equation:

$$UR = \frac{U}{C_{tw}} \tag{A.3}$$

U is the total number of URLs in tweets.

Tweets interval (TWI): The average time interval between two consecutive tweets. TWI can be calculated by the following equation:

$$TWI = \frac{\sum_{i=1}^{C_{tw}-1} (t_{i+1} - t_i)}{C_{tw} - 1} \tag{A.4}$$

t_i is the posted time of the i-th tweet of an account. We calculate the time in minutes.

Retweet count (RT): The average retweet count of all the tweets of an account. RT calculated by the following equation:

$$RT = \frac{\sum_{i=1}^{C_{tw}} R_i}{C_{tw}} \tag{A.5}$$

R_i is the retweet count of the i-th tweet of an account.

Favorite count (FT): The average favorite count of all the tweets of an account.

$$FT = \frac{\sum_{i=1}^{C_{tw}} F_i}{C_{tw}} \tag{A.6}$$

F_i is the favorite count of the i-th tweet of account.

Source count (SN): The tweets on Twitter can be posted with different source such as: web, mobile phone, twitter bot and so on. The source count count the number of

different sources for all the tweets of an account.

Tweet similarity (TS): The tweet similarity measure the similarity among the tweets posted by an account. It can be calculated with the following equation introduced in [9]

$$TS = \frac{\sum_{p \in P} c\,(p)}{l_a l_p} \tag{A.7}$$

P is the set of possible tweet-to-tweet combinations among any two tweets posted by a certain account, p is a single pair, $c(p)$ is a function calculating the number of words two tweets share, l_a is the average length of tweets posted by that account, and l_p is the number of tweet combinations.

References

[1] "Justin bieber twitter followers: 50% are fake, says report." http://www.digitalspy.ca/music/news/a471915/justin-bieber-twitter-followers-50-percent-are-fake-says-report.html#~oBOjVq7yQVfhdx.

[2] K. Thomas, "The role of the underground economy in social network spam and abuse." 2013.

[3] K. Thomas, C. Grier, D. Song, and V. Paxson, "Suspended accounts in retrospect: an analysis of twitter spam," in *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference*, pp. 243–258, ACM, 2011.

[4] N. Provos, "A virtual honeypot framework," in *Proc. 13th USENIX security symposium*, 2004.

[5] F. Pouget, M. Dacier, and V. H. Pham, "Vh: Leurre. com: on the advantages of deploying a large scale distributed honeypot platform," in *E-Crime and Computer Conference*, Citeseer, 2005.

[6] M. Andreolini, A. Bulgarelli, M. Colajanni, and F. Mazzoni, "Honeyspam: Honeypots fighting spam at the source," in *Proc. Workshop on Steps to Reducing Unwanted Traffic*, 2005.

[7] S. Webb, J. Caverlee, and C. Pu, "Social honeypots: Making friends with a spammer near you.," in *CEAS*, 2008.

[8] K. Lee, J. Caverlee, and S. Webb, "Uncovering social spammers: social honeypots+ machine learning," in *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pp. 435–442, ACM, 2010.

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

[10] S. Ghosh, B. Viswanath, F. Kooti, N. K. Sharma, G. Korlam, F. Benevenuto, N. Ganguly, and K. P. Gummadi, "Understanding and combating link farming in the twitter social network," in *Proceedings of the 21st international conference on World Wide Web*, pp. 61–70, ACM, 2012.

[11] C. Yang, R. Harkreader, J. Zhang, S. Shin, and G. Gu, "Analyzing spammers' social networks for fun and profit: a case study of cyber criminal ecosystem on twitter," in *Proceedings of the 21st international conference on World Wide Web*, pp. 71–80, ACM, 2012.

[12] V. Sridharan, V. Shankar, and M. Gupta, "Twitter games: How successful spammers pick targets," in *Proceedings of the 28th Annual Computer Security Applications Conference*, pp. 389–398, ACM, 2012.

[13] "The state of the twitterverse 2012." http://www.briansolis.com/2012/02/the-state-of-the-twitterverse-2012/.

[14] K. Thomas, C. Grier, and V. Paxson, "Adapting social spam infrastructure for political censorship," in *Proceedings of the 5th USENIX conference on Large-Scale Exploits and Emergent Threats*, pp. 13–13, USENIX Association, 2012.

[15] C. Kanich, C. Kreibich, K. Levchenko, B. Enright, G. M. Voelker, V. Paxson, and S. Savage, "Spamalytics: An empirical analysis of spam marketing conversion," in *Proceedings of the 15th ACM conference on Computer and communications security*, pp. 3–14, ACM, 2008.

[16] K. Thomas, D. McCoy, C. Grier, A. Kolcz, and V. Paxson, "Trafficking fraudulent accounts: the role of the underground market in twitter spam and abuse," in *USENIX Security Symposium*, 2013.

[17] G. Stringhini, G. Wang, M. Egele, C. Kruegel, G. Vigna, H. Zheng, and B. Y. Zhao, "Follow the green: growth and dynamics in twitter follower markets," in *Proceedings of the 2013 conference on Internet measurement conference*, pp. 163–176, ACM, 2013.

[18] M. Egele, G. Stringhini, C. Kruegel, and G. Vigna, "Compa: Detecting compromised accounts on social networks," in *Symposium on Network and Distributed System Security (NDSS)*, 2013.

[19] "Gmail fires back in the war on spam." http://gadgetwise.blogs.nytimes.com/2012/04/11/gmail-fires-back-in-the-war-on-spam/?_php=true&_type=blogs&_r=0.

[20] H. Nguyen, "Research report: 2013 state of social media spam," tech. rep., Nexgate, 2013.

[21] P. Heymann, G. Koutrika, and H. Garcia-Molina, "Fighting spam on social web sites: A survey of approaches and future challenges," *Internet Computing, IEEE*, vol. 11, no. 6, pp. 36–45, 2007.

[22] B. Krause, C. Schmitz, A. Hotho, and G. Stumme, "The anti-social tagger: detecting spam in social bookmarking systems," in *Proceedings of the 4th international workshop on Adversarial information retrieval on the web*, pp. 61–68, ACM, 2008.

[23] F. Benevenuto, G. Magno, T. Rodrigues, and V. Almeida, "Detecting spammers on twitter," in *Collaboration, electronic messaging, anti-abuse and spam conference (CEAS)*, vol. 6, p. 12, 2010.

[24] T.-S. Moh and A. J. Murmann, "Can you judge a man by his friends?-enhancing spammer detection on the twitter microblogging platform using friends and followers," in *Information Systems, Technology and Management*, pp. 210–220, Springer, 2010.

[25] C. Yang, R. C. Harkreader, and G. Gu, "Die free or live hard? empirical evaluation and new design for fighting evolving twitter spammers," in *Recent Advances in Intrusion Detection*, pp. 318–337, Springer, 2011.

[26] J. Song, S. Lee, and J. Kim, "Spam filtering in twitter using sender-receiver relationship," in *Recent Advances in Intrusion Detection*, pp. 301–317, Springer, 2011.

[27] Z. Chu, S. Gianvecchio, H. Wang, and S. Jajodia, "Who is tweeting on twitter: human, bot, or cyborg?," in *Proceedings of the 26th annual computer security applications conference*, pp. 21–30, ACM, 2010.

[28] M. G. Noll, C.-m. Au Yeung, N. Gibbins, C. Meinel, and N. Shadbolt, "Telling experts from spammers: expertise ranking in folksonomies," in *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pp. 612–619, ACM, 2009.

[29] Y. Yamaguchi, T. Takahashi, T. Amagasa, and H. Kitagawa, "Turank: Twitter user ranking based on user-tweet graph analysis," in *Web Information Systems Engineering–WISE 2010*, pp. 240–253, Springer, 2010.

[30] L. Page, S. Brin, R. Motwani, and T. Winograd, "The pagerank citation ranking: Bringing order to the web.," 1999.

[31] Y. Duan, L. Jiang, T. Qin, M. Zhou, and H.-Y. Shum, "An empirical study on learning to rank of tweets," in *Proceedings of the 23rd International Conference on Computational Linguistics*, pp. 295–303, Association for Computational Linguistics, 2010.

[32] I. Uysal and W. B. Croft, "User oriented tweet ranking: a filtering approach to microblogs," in *Proceedings of the 20th ACM international conference on Information and knowledge management*, pp. 2261–2264, ACM, 2011.

[33] H. Gao, J. Hu, C. Wilson, Z. Li, Y. Chen, and B. Y. Zhao, "Detecting and characterizing social spam campaigns," in *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement*, pp. 35–47, ACM, 2010.

[34] K. Thomas, C. Grier, J. Ma, V. Paxson, and D. Song, "Design and evaluation of a real-time url spam filtering service," in *Security and Privacy (SP), 2011 IEEE Symposium on*, pp. 447–462, IEEE, 2011.

[35] S. Lee and J. Kim, "Warningbird: Detecting suspicious urls in twitter stream," in *Symposium on Network and Distributed System Security (NDSS)*, 2012.

[36] C. Grier, K. Thomas, V. Paxson, and M. Zhang, "@ spam: the underground on 140 characters or less," in *Proceedings of the 17th ACM conference on Computer and communications security*, pp. 27–37, ACM, 2010.

[37] D. Wang, S. B. Navathe, L. Liu, D. Irani, A. Tamersoy, and C. Pu, "Click traffic analysis of short url spam on twitter," in *Collaborative Computing: Networking, Applications and Worksharing (Collaboratecom), 2013 9th International Conference Conference on*, pp. 250–259, IEEE, 2013.

[38] E. Tan, L. Guo, S. Chen, X. Zhang, and Y. Zhao, "Spammer behavior analysis and detection in user generated content on social networks," in *Distributed Computing Systems (ICDCS), 2012 IEEE 32nd International Conference on*, pp. 305–314, IEEE, 2012.

[39] A. Ramachandran, N. Feamster, and S. Vempala, "Filtering spam with behavioral blacklisting," in *Proceedings of the 14th ACM conference on Computer and communications security*, pp. 342–351, ACM, 2007.

[40] K. Lee, J. Caverlee, Z. Cheng, and D. Z. Sui, "Content-driven detection of campaigns in social media," in *Proceedings of the 20th ACM international conference on Information and knowledge management*, pp. 551–556, ACM, 2011.

[41] X. Zhang, S. Zhu, and W. Liang, "Detecting spam and promoting campaigns in the twitter social network," in *Proceedings of the 2012 IEEE 12th International Conference on Data Mining*, pp. 1194–1199, IEEE Computer Society, 2012.

[42] H. Gao, Y. Chen, K. Lee, D. Palsetia, and A. Choudhary, "Towards online spam filtering in social networks," in *Symposium on Network and Distributed System Security (NDSS)*, 2012.

[43] M. Flores and A. Kuzmanovic, "Searching for spam: detecting fraudulent accounts via web search," in *Passive and Active Measurement*, pp. 208–217, Springer, 2013.

[44] "Google safe browsing api." https://developers.google.com/safe-browsing/.

[45] "Uribl.com - realtime uri blacklist." http://www.uribl.com/.

[46] "joewein.de llc - fighting spam and scams on the internet." http://www.joewein.net/.

[47] "Surbl." http://www.surbl.org/.

[48] "Mcafee siteadvisor live." http://home.mcafee.com/store/siteadvisor-live.

[49] H. Kwak, C. Lee, H. Park, and S. Moon, "What is twitter, a social network or a news media?," in *Proceedings of the 19th international conference on World wide web*, pp. 591–600, ACM, 2010.

[50] K. S. Jones, "A statistical interpretation of term specificity and its application in retrieval," *Journal of documentation*, vol. 28, no. 1, pp. 11–21, 1972.

[51] "The twitter rules." https://support.twitter.com/articles/18311-the-twitter-rules.

[52] "Obama has millions of fake twitter followers." http://content.usatoday.com/communities/theoval/post/2012/08/obama-has-millions-of-fake-twitter-followers/1#.Uk5N5Rs3s50.

[53] Z. Gyöngyi, H. Garcia-Molina, and J. Pedersen, "Combating web spam with trustrank," in *Proceedings of the Thirtieth international conference on Very large data bases-Volume 30*, pp. 576–587, VLDB Endowment, 2004.

[54] A. J. Smola and B. Schölkopf, "A tutorial on support vector regression," *Statistics and computing*, vol. 14, no. 3, pp. 199–222, 2004.

[55] "Weka 3: Data mining software in java." http://www.cs.waikato.ac.nz/ml/weka/.