J Intell Inf Syst (2018) 51:503–527
https://doi.org/10.1007/s10844-017-0494-z

Discovering spammer communities in twitter

P. V. Bindu[1] · **Rahul Mishra**[1] · **P. Santhi Thilagam**[1]

Received: 31 May 2017 / Revised: 23 December 2017 / Accepted: 26 December 2017 /
Published online: 10 January 2018

Abstract Online social networks have become immensely popular in recent years and have become the major sources for tracking the reverberation of events and news throughout the world. However, the diversity and popularity of online social networks attract malicious users to inject new forms of spam. Spamming is a malicious activity where a fake user spreads unsolicited messages in the form of bulk message, fraudulent review, malware/virus, hate speech, profanity, or advertising for marketing scam. In addition, it is found that spammers usually form a connected community of spam accounts and use them to spread spam to a large set of legitimate users. Consequently, it is highly desirable to detect such spammer communities existing in social networks. Even though a significant amount of work has been done in the field of detecting spam messages and accounts, not much research has been done in detecting spammer communities and hidden spam accounts. In this work, an unsupervised approach called **SpamCom** is proposed for detecting spammer communities in Twitter. We model the Twitter network as a multilayer social network and exploit the existence of overlapping community-based features of users represented in the form of Hypergraphs to identify spammers based on their structural behavior and URL characteristics. The use of community-based features, graph and URL characteristics of user accounts, and content similarity among users make our technique very robust and efficient.

Keywords Spammer detection · Anomaly detection · Spammer community · Twitter · Online social networks · Multilayer social networks

✉ P. V. Bindu
bindupv007@gmail.com

Rahul Mishra
mishrahul11@gmail.com

P. Santhi Thilagam
santhi@nitk.ac.in

[1] Department of Computer Science & Engineering, National Institute of Technology Karnataka, Surathkal, India

1 Introduction

The emerging technology of online social networks has led to the development of various platforms through which millions of social entities collaborate and communicate with each other. For example, Facebook boasts of 1.5 billion active users (Facebook 2016) while Twitter has 320 million active users (Twitter 2016) in the year 2016, allowing their users to stay in touch with friends, share ideas and information, publish their personal information (profile), and make work-related communications.

Twitter is one of the most popular and fastest growing online social networks. Founded in 2006, Twitter has emerged as most popular microblogging platform where users can share news, media, meme, views, and updates in the form of tweet. A tweet is a post containing text and HTTP URLs limited to 140 characters. Many popular search engines like Yahoo, Microsoft Bing, and Google use Twitter stream to track the live updates of happenings around the world to provide information practically without any delay.

The users of online social networks often find social networks more secure and threat-free due to their popularity and ease of access (Swamynathan et al. 2008). Although the ease of spreading news and discussing views sounds appealing, it also opens door to develop new opportunities and variants of anomalous and malicious activities in social networks (Bindu and Thilagam 2016; Bindu et al. 2017). Spamming is the most ubiquitous form of malicious activity in social networks. Spamming involves undesirable users sending tweets consisting of text and HTTP URLs to large number of legitimate users as possible. The motivations for spammers to spread spam messages is with an aim for promotional marketing by capturing trending topics, spreading views, and generating revenues based on URL clicks. It leads to uncontrolled dissemination of content, virus/malware, scams, pornography, and advertisements leading to huge wastage of network bandwidth and revenue losses of organization. It can lead to psychological, financial, or physical harassment of legitimate users by these malicious users leading to dissatisfaction with the service and environment provided by social network platforms.

The most widely recognized type of spamming in Twitter is to capture the trending topics (Martinez-Romo and Araujo 2013). Whenever a noteworthy event occurs, users try to express their opinion or share information on the event using hashtags. If the topic is most tweeted-about in the day, it is visible to all the Twitter users in Twitter homepage as trending topic. The spammers use the same hashtags to be visible to a large user base following the particular trending event but with unsolicited URL's leading to unrelated websites. Due to the 140 character limitation in twitter, the users usually share URL's using URL shortening service. Moreover, the spammers take advantage of URL shortening service to make the identification of spam related URLs difficult for users. A study shows that 45% of users in social networking platforms readily click any URL posted by a friend. Thus, spammers are attracted to use social networking platforms to send unsolicited messages and malicious links to legitimate users, and hijack trending topics. It has been reported that more than 11% of tweets in Twitter are spams (SciTechBlog 2016).

Currently, Twitter uses its "Follow Limit Policy" to filter possible spam accounts. According to Twitter Rules[1],"a Twitter account can be considered to be spam account, and thus can be suspended by Twitter, if it has comparatively a small number of followers compared to the amount of accounts that it follows." However, different from other social networks, microblogging platform such as Twitter allows the user to follow any account

[1] http://help.twitter.com/entries/18311-the-twitter-rules

without their consent. This unidirectional binding allows the spammers to follow a large base of random accounts. Many legitimate users, also called supporters or social capitalists, blindly follow back the accounts for the sake of courtesy, after they are being followed by someone. A recent study on microblogging websites proves that a large fraction of such supporters follow back the spammers helping them to break through the Twitter "Follow Limit Policy" thereby increasing the accounts' popularity and credibility (Ghosh et al. 2012). The following of these accounts also helps the spammers to increase their influence on their followers along with avoidance of suspicion or detection. Additionally, the spammers can purchase followers from websites (Yang et al. 2013). These websites have a large base of bot accounts that follow their customers once the payment is done.

Spammers usually mimic the patterns of legitimate user behavior to avoid being detected by spam detection techniques. Spammers develop tools and techniques to evade the existing techniques for detection. Additionally, the current research trends on spam detection have complexity constraints or have some caveats that can be bypassed by the spammers. In this regard, it is highly desirable to detect and block/remove spammers from social networks such as Twitter to save resources and human efforts from unwanted users. Including more robust features that are harder to mimic and using the interaction of users within and outside the community structures can be used to build spam classification models making it difficult for spammers. Spammers primarily form a bunch of fake accounts, and collaborate with each other forming a closely-knit community to increase their credibility. Thus, spammer accounts tend to be socially well-connected with high clustering coefficient (Yang et al. 2012). The accounts sitting at the center of such communities are generally referred to as spam hubs and are inclined to follow large base of spamming accounts. These well-connected communities target a large base of random accounts by spamming them with shortened URLs.

Even though a substantial amount of research work has been carried out in the field of detecting spam messages and social spammers, not much work has been done in detecting spammer communities. Hence, in this paper, we aim to detect spam communities residing in Twitter by analyzing the spam accounts' community features and robust characteristics of these accounts that are difficult to evade by spammers. Like legitimate users, spammers also participate in many overlapping communities and can send different or same spam messages in different communities. Consequently, the overlapping community based features existing in Twitter network, the structural characteristics, URL (content) based characteristics, user behavior, and user account characteristics are employed to detect spammer communities in Twitter. In order to represent the content, behavioral, and structural characteristics of the users of Twitter network for detecting spammer communities, the network is modeled as a *directed and attributed multilayer social network*. The goal is to classify the accounts as spammers and legitimate users, and find social connections between the spammers. To best of our knowledge, this is the first in-depth effort to detect spammer communities existing in Twitter. It is found that the spammers collude with each other and have a small world network. In summary, the major contributions of our study are as follows:

- Modeling the topological, tweet content, and behavioral characteristics of the users of Twitter network by using a *directed and attributed multi-layer social network* with two layers - *Follower* and *Tweet* network layers
- Proposing a novel and efficient, unsupervised approach called **SpamCom** to detect spammer communities by employing community-based features, robust structural and user-behavior characteristics, URL (content) based characteristics, and user profile characteristics that are difficult to evade by spammers

- Capturing the hidden spammers that try to hide in communities and spread malicious information through other spammers using the proposed framework
- Evaluating the performance of the proposed approach based on the communities detected by the algorithm. The experimental results show that the spammer communities have very high clustering coefficients and target users collectively. The spammer detection algorithm is found to be 89% precise.

The rest of the paper is organized as follows. Section 2 consists of related work of the spam detection in social networks. Section 3 presents how the Twitter network is modeled as a multilayer social network. Section 4 deals with problem description which includes the formal definition of the problem. In Section 5, the solution methodology for the problem is explained. The experimental results are presented in Section 6 and conclusion is presented in the last section.

2 Related work

This section presents a brief review of the existing techniques and characteristics used to detect spammers and an in-depth taxonomy of the parameters and techniques used. The role of community-based features in identifying spammers is also presented. Additionally, the research gaps that help us to propose the spammer community detection approach are presented.

2.1 Spam detection

Most of the research on spam detection in recent years has been to detect spam or spam accounts individually (Benevenuto et al. 2008, 2010; Lee et al. 2010; Stringhini et al. 2010; Yang et al. 2013; Wang 2010). Less amount of research has been performed to understand the social relationship existing among the spam accounts in Twitter. The spam content detection usually includes content-based filtering (Benevenuto et al. 2010), URL blacklists (Gao et al. 2010), and spam traps known as honeypots (Lee et al. 2010, 2011; Stringhini et al. 2010) to build classifier algorithms.

Initial works on spam detection in Twitter majorly focused on analyzing the social behavior and network characteristics of spam accounts by studying a spam campaign (Yardi et al. 2009; Mustafaraj and Metaxas 2010). Similarly, a study of the spread of Astroturf memes for a political campaign in Twitter is analyzed by Ratkiewicz et al. (2011) and Ratkiewicz et al. (2011). These works mainly focus on information diffusion in Twitter based on content using a supervised algorithm (Ratkiewicz et al. 2011) and use of network information using a clustering algorithm (Ratkiewicz et al. 2011). Gao et al. (2010) presented the quantification and characterization of spam campaigns in social networks by detecting spam clusters using the content and user behavioral characteristics. Spam clusters are initially detected based on similarity of URLs posted by the users to form correlated subsets of posts. Using the dual behavioral hints of burstiness and distributive communication within subsets, the identification of malicious spam campaigns is done. The distributive property focuses on the number of users within a community sending the same set of URLs and the bursty nature depicts the short time span within which the messages were posted. Thomas et al. (2011) analyzed the suspended accounts by Twitter to learn about the tools, techniques, and support infrastructure used by spammers.

A multitude of spammer detection techniques based on machine learning classification algorithms have been developed by researchers. Such classification models use machine

learning techniques from training instances to learn and develop a spam signature (Benevenuto et al. 2010; Lee et al. 2010; Wang 2010; Chu et al. 2010; Song et al. 2011). It includes network information (in-degree, out-degree, bi-directional links, etc.), user profile information (about me, address, etc.), content information, and user behavior (interactions with other users, clustering coefficient, etc.). Stringhini et al. (2010) have developed a machine learning algorithm that uses textual features of spammer profile and their interactions in the network to develop spam signatures. Initially, it involved human classification for building the training set. The process of human inspection to build classifiers is a costly process involving a lot of human efforts to build training data. Spammers constantly can adapt to the classification algorithms strategies/tactics and make their feature sets match to the feature sets of legitimate users to avoid being detected by spam detection classifiers. The spam classifiers can go stale quickly by the adaptation of spammers. It is based on the assumption that spammers follow a pattern in their profile description and use a set of distinguished keywords and URLs while interacting with other users. However, this assumption has been found to be evaded by copy-profiling (imitation of the profile of legitimate user) and content obfuscation by spammers (Song et al. 2011; Yang et al. 2013). DeBarr and Wechsler (2009), and Wang (2010) have used more robust characteristics such as graph-based metrics and degree centrality based metrics to detect spammers. Benevenuto et al. (2008, 2010) used video rating, user behavioral characteristics, and topological characteristics to detect spammers in video sharing online social network.

Another method proposed by researchers to detect spammers is content-based analysis known as keyword-based filtering (Grier et al. 2010). The drawback of content-based analysis is that it involves a huge amount of computation and usually has a big delay in identifying malicious links. Secondly, spammers use non-dictionary words or images to counter the keyword-based filtering. Tools have been developed that post the same tweet with the same meaning but different words, being posted to a large random base of users. Moreover, there has been a change in Twitter Policy in allowing the content access due to the user privacy protection issue. The use of user-content for detecting spammers is often being reported as a violation of privacy by many users.

Finally, many existing studies depend on using social honeypots to attract and detect spammers (Lee et al. 2010, 2011; Yang et al. 2014). Social honeypots are administered bot accounts that monitors and logs spammer behavior and features. Any unusual activity by a user is automatically logged by the bots. These spammers are later manually classified and further analyzed by the researchers to develop spam signatures. Finally, the information collected in logs and the spam signatures are used to develop classification algorithms based on machine learning approaches. Yang et al. (2014) has used tweet content and user behavioral characteristics using social honeypots to identify the taste of spammers. The tastes identified from the machine learning algorithm is used to further detect spammers. However, honeypot classification is not much efficient in terms of entire Twitter scope involving a huge number of spam accounts. These techniques require passively waiting for spammers and thus does not include all spammers. Additionally, the spammer can evade honeypot detection by copy-profiling. Honeypot classification is not scalable and requires manual efforts. As discussed above, the manual classification is a tedious, time-consuming and heavy-weight process. Given the restricted time and resource constraints, relatively a much simpler and automated process is desired to detect spammers from such a large base of Twitter universe.

A summary and comparison of the features and methods used by popular spammer detection techniques are given in Table 1. The methods popularly used by researchers include supervised learning, URL blacklisting, clustering, and use of social honeypots to trap spammers. The advantages and disadvantages of these methods have already been discussed

Table 1 Summary and comparison of articles on spammer detection in social networks

		Features			Methods			
Research Article	Year	Content	Network	Behavior	Supervised	URL Blacklist	Clustering	Social Honeypot
Benevenuto et al. (2008)	2008		✓	✓	✓			
Yardi et al. (2009)	2009		✓	✓	✓			
DeBarr and Wechsler (2009)	2009		✓		✓			
Lee et al. (2010)	2010	✓			✓			✓
Benevenuto et al. (2010)	2010	✓		✓	✓			
Gao et al. (2010)	2010	✓		✓			✓	
Grier et al. (2010)	2010	✓				✓		
Stringhini et al. (2010)	2010	✓	✓		✓			✓
Wang (2010)	2010	✓	✓		✓			
Mustafaraj and Metaxas (2010)	2010		✓	✓	✓			
Chu et al. (2010)	2010	✓	✓	✓	✓			
Ratkiewicz et al. (2011)	2011	✓			✓			
Ratkiewicz et al. (2011)	2011		✓				✓	
Song et al. (2011)	2011		✓		✓			
Fire et al. (2012)	2012		✓	✓			✓	
Yang et al. (2012)	2012		✓				✓	
Hu et al. (2013)	2013	✓	✓		✓			
Bhat and Abulaish (2013)	2013		✓				✓	
Yang et al. (2014)	2014	✓		✓	✓			✓
Zheng et al. (2015)	2015	✓		✓	✓			

above. The spammer detection algorithms require certain features to detect spammers. It can be contextual features such as tweet content, URL information, length of profile description, username, etc. to detect spammers. Content information gives most accuracy in detecting spammers but involves lot of computation to recognize the credibility of content. Next, the spammer detection techniques use more generic network or topological information. The network information consists of number of followers, followings, bidirectional links, clustering coefficient, mean degree, etc. Network information is easy to compute and has more availability. Based on the topological characteristic of known spammers, a spam signature can be created to detect future spammers. However, spammers usually are successful in evading most of the network signature detection techniques by mimicking legitimate users. Finally, there are some behavioral features that are extracted by researchers to detect spam accounts. The behavioral features depict the general behavior of an account in a social network. It includes features such as ratio of URLs in tweet, fraction of hashtags in tweet, number of re-tweets, ratio of username in tweet, burstiness in tweet, etc. The behavioral features are robust features that the spammers find difficult to evade. The spammer needs to behave like legitimate users to avoid detection which is harder as compared to mimicking topological characteristics. In this paper, all the three kinds of features, viz., content, network, and behavior are employed to detect spam accounts. The network and behavioral features introduced are most robust and very hard to mimic for spammers. Additionally, the community-based features used to detect spammers make the proposed approach novel and robust compared to previous works.

In social networks, users belong to multiple overlapping communities. The overlapping community structure exists even for spammers in social networks. Spammers are known to form a close-knit community among themselves with high clustering coefficient. Additionally, these spammers send a large number of spam messages to a large base of random legitimate users. These randomly selected users, are generally socially unconnected and does not show community structure among themselves. This kind of spam attack is called Random Link Attack (RLA) (Shrivastava et al. 2008). Generally, the clustering coefficient is a good feature that can be exploited to detect RLA attacks. Hence, the authors used clustering coefficient and neighborhood independence to tackle with RLA from spammers in networks. Spammers usually form connections among themselves and with supporters (users that readily follow back) to obtain a high clustering coefficient similar to legitimate users to evade RLA detection schemes. Ying et al. (2011) exploit graph spectral analysis that deals with the analysis of the eigenvalues and eigenvector components of the graph adjacency matrix rather than traditional topology-based approach to detect RLAs faster.

Fire et al. (2012) incorporated the idea of using community detection to detect spammers. Each community detected by them was analyzed based on the interactions of the user, in-degree and out-degree of the user, the number of communities the users belongs to, and the number of links between the friends of the user. Bhat and Abulaish (2013) proposed the detection of dynamic overlapping communities, and exploited the role and interaction of nodes within the communities to classify them as spammers or legitimate users.

2.2 Research gaps

Most of the existing works have been based on learning content or user based features to detect spammers. Commonly, the features used to detect spammers include number of follow, followers, malicious URLs, follower to follow ratio, reputation, number of retweets, etc. Still improvements can be made by addressing some unexplored areas and techniques that are mentioned as follows:

- Even though the signatures that use user and content-based features to detect spammers are useful, they are not robust and can get stale because spammers use various tools and techniques to evade detection and conceal their fake identities. The focus must be on identifying the behavioral characteristics of spammers to help behavior-driven suspicious signatures in detecting them.
- Spammers usually operate as a group within a same locality and time period. There is a lack of research in the direction of detecting spammers based on intention, environment, and temporal information of spammers.
- Most of the existing techniques for spammer detection employ spammer scores or thresholds based on their signature. If any user is crossing the threshold, it is marked as a spammer. Quantification of spammer score to essentially classify the user as a spammer or legitimate user is still an open issue.
- The in-depth analysis of the community structure of spammers existing in social networks is a significant open issue.

The works of Yardi et al. (2009), Gao et al. (2010), Yang et al. (2013), and Thomas et al. (2011) provide us with deep and valuable insight with the tools, techniques, and characteristics that describe the spammers. The taste of the spammers and the strategies that can be used to effectively detect spammers have been addressed by many recent researchers. However, the existing techniques involving machine learning approaches, URL blacklisting, and social honeypots have limitations as described above. Additionally, significantly less amount of work has been carried out in the direction of analysis of community structure among spammers. The motivation for the proposed work comes from RLA (Shrivastava et al. 2008) prevalent in social networks including Twitter and the existence of spammer community ecosystem (Yang et al. 2012) in social network. Compared to the existing literature, our work primarily focuses on detection of spammer community ecosystem by investigating the overlapping community structures existing in the social network along with URL similarity, uniqueness, user topological features, and user profile features to classify users as spammers. The characteristics used in the proposed work have been found to be most robust and have best discriminative power compared to other features for detecting spammers.

3 Twitter multilayer social network

In order to represent the tweet content, behavioral, and structural characteristics of the users of Twitter network for detecting spammer communities, the Twitter network is modeled as a *multilayer social network*. In this section, we describe what is a multilayer social network and how the Twitter network is modeled as a multilayer social network.

3.1 Multilayer social networks and media multiplexity

A multilayer network is a complex network consisting of several modes of interactions among the same set of entities. Many complex real-world systems can be modeled as multilayer networks. Examples include social networks where multiple types of interactions exist among individuals, transportation systems where multiple types of travel exist between places, and biological systems where multiple types of interactions exist among biological entities such as genes.

In a social setting, media multiplexity (Haythornthwaite 2005) is the term used to refer to the multiple means of communication among individuals. The more the number of communication means among the users, the more is the relationship strength. In other words, media multiplexity indicates stronger tie among individuals in any social setting. People can interact in several ways including phone calls, instant messages, unscheduled and scheduled meetings, emails, online social networks, etc. It is observed that there exists high social influence among strongly tied individuals. In addition, if a medium of communication fails, the people with strong ties will be less affected, as they are connected through multiple means of communication. Moreover, if a new means of communication is introduced, strong ties are more likely to adopt it if it suits their needs and is useful for maintaining the relationship among them.

Media multiplexity is modeled using multilayer networks. In online social networks, users can interact in multiple ways simultaneously leading to the formation of multilayer social networks among the same set of users (Bródka and Kazienko 2014; Bindu et al. 2017). For example, in Facebook, users can interact with each other through private messages, post contents on each other's walls, like posts of other users, tag other users in posts, etc. Similarly in Twitter, users can follow each other, tweet and re-tweet messages, reply to tweets, and mention other users. Each type of interaction is viewed as a distinct network layer in a multilayer network. In the proposed work, for discovering spammer communities, we consider the follow and tweet interactions of the users and model Twitter as a multilayer social network by representing the two interactions as two separate layers.

A social network is usually represented as a graph $G(V, E)$, where V is the set of nodes or users and E is the set of edges showing the interactions among the users in the network. This representation is highly useful in modeling many social phenomena. However, it can represent only one type of relationship among users in the network. In other words, conventional graph representation can model a singe-layer network effectively. However, it can not represent the multiple ties existing among the users of a social network. Hence, we model the simultaneous interactions among users as a multilayer network M, or a set of m graphs, each representing the interactions in a distinct layer, as $M = \{G^1, G^2, ..., G^m\}$. Each layer in the multilayer network can be considered as a network on its own, and the i^{th} layer of the multilayer network M is denoted as $G^i(V^i, E^i)$, where V^i and E^i are respectively the nodes and edges of the layer i. A node can be present in one or more layers. If all the nodes are not present in every layer of the multilayer network, a union of the nodes in the network layers is taken as the shared node set, i.e. $V = \bigcup_{i=1}^{m} V^i$ and $n = |V|$, number of nodes in V.

3.2 Twitter network layers

In order to represent the tweet content, behavioral, and structural characteristics of the users of Twitter network for detecting spammer communities, the network is modeled as a *directed and attributed multilayer social network* with two layers, $M = \{G^F, G^T\}$, where G^F is the *Follower* network layer and G^T is the *Tweet* network layer respectively. The Twitter multilayer social network considered in our study is a heterogeneous network; the *Follower* layer is an attributed network with the nodes labeled with profile features, whereas the *Tweet* layer is an attributed network with the edges labeled with the tweet URLs posted by the users. The *Follower* and *Tweet* layers can also be modeled as attributed network layers with both node attributes and edge attributes. For instance, in *Follower* layer, we can include edge attributes showing when the relationships have been created. This auxiliary information may be used to detect spammers when comparing with the time of

tweets. However, in this work, we have considered *Follower* layer as node-attributed and *Tweet* layer as edge-attributed. The layers are explained in detail as follows:

1. **Follower network layer:** This layer represents the Follower/Followee relationship in Twitter. The layer is modeled as $G^F(V, E^F, A)$, where V is the set of users in the network, $E^F = \{< i, j > | i, j \in V\}$ is the set of Follower/Followee relationships among the users, and A is the set of profile attributes. The directed edge (i, j) indicates that user i is following user j. User i is said to follow j and is called a *Follower* of user j. Hence, the number of followers of a user is the set of incoming links or the in-degree of the node. It can be represented as N_{fer}. In the case of the edge (i, j), user j is said to be the *Follow* of user i. Hence, *Follow* is the set of outgoing edges of a node. The total number of *Follow* is the out-degree of the node and is represented as N_{fing}. The nodes of the *Follower* layer are labeled with profile characteristics such as node ID and the time-stamp when the user account has been created. The time-stamp information of the node is used to find the age of the corresponding user account.
2. **Tweet network layer:** The second layer of Twitter network considered for our study is the *Tweet* network layer, G^T. As the spam in twitter mainly comprises of URLs, we have a set of URLs tweeted by the users. Hence, *Tweet* layer models the tweets of URLs posted by the users in the network. This layer is attributed with tweet contents or tweet URLs associated with the edges of the layer. It is represented as $G^T(V, E^T, U)$, where V is the set of users , $E^T = \{< i, j > | i, j \in V\}$ is the set of edges, and U is the set of URLs posted by the users. Each edge $< i, j >$ is associated with a set of URLs posted by the user i to user j. The tweet URL information of the edges is used to determine the uniqueness and similarity of the URLs tweeted by the users.

4 Problem description

Given the directed and attributed Twitter multilayer social network $M = \{G^F, G^T\}$, where $G^F(V, E^F, A)$ is the *Follower* network layer and $G^T(V, E^T, U)$ is the *Tweet* network layer, our aim is to develop an unsupervised approach that extracts the overlapping community structure existing in the social network and analyzes the user's clustering coefficient, neighborhood, behavior, and content information to detect spammer communities in Twitter. The output is multiple connected components from the Twitter network that represent the set of socially connected spammer communities. The set of all the symbols used in our work is defined in Table 2.

To explain the working of the proposed methodology, the following terms are defined:

- Spammer community: A group of highly connected spammers in the Twitter ecosystem to increase their credibility and spread. The higher the number of followers, the more credibility it obtains. Additionally, this community acts as a medium to interact to other spammers.
- Hidden spammer: A hidden spammer is a spam account having connections with multiple spam accounts, but not with legitimate accounts. Even though the hidden spam account can act as spam hub and operate the functioning of other spam accounts, it does not perform spamming of legitimate accounts to prevent ban from Twitter. This is done to increase the importance of the account by increasing the number of followers.
- Local mining: Local mining uses the features that are local to a community to detect spammers.

Table 2 Symbols

Symbols	Definitions
M	Twitter multilayer social network
G^F	Follower network layer of M
G^T	Tweet network layer of M
n	Number of nodes in M
i, j	Node indices $1 \leq i, j \leq n$
U	Set of all URLs posted by the users in the dataset
V	Set of nodes in G^F and G^T
E^F	Set of edges representing following relationship in G^F
E^T	Set of edges representing tweet relationships in G^T
A	Set of all profile attributes of the users
H	Hypergraph of overlapping communities detected from G^F
$N_{fer}(v)$	Number of followers of user v
$N_{fing}(v)$	Number of users followed by user v
U_v	The URL posted in tweet by user v

– Global mining: Global mining uses the features that are globally the same throughout the communities.

A toy example of possible social network ecosystem is provided in Fig. 1. The spammers are shown as shaded nodes in the figure. The approach intends to use the community structure in social networks to cluster the users. Later, each community is analyzed in parallel to detect spammers. It can be seen that there are four legitimate users and seven spammers

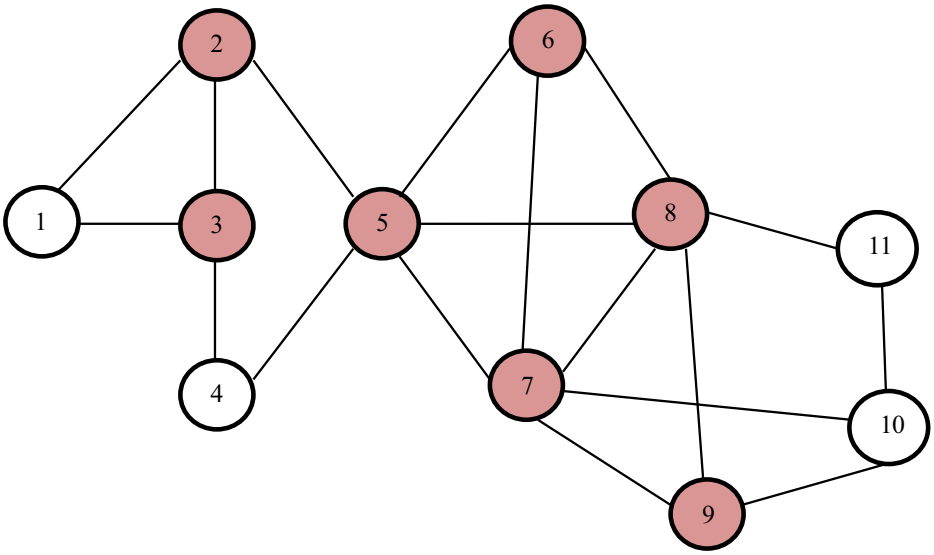


Fig. 1 Toy example for a social network

in the ecosystem. Suppose, three communities are obtained after applying the overlapping community algorithm as shown in the figure. Overlapping communities are detected because in online social networks it is likely that a user belongs to multiple communities and hence, the communities naturally overlap. The number of communities an individual can belong to is essentially unlimited because the individual can simultaneously associate with as many groups as he wishes based on his interests. Like legitimate users, a spammer also can participate in many communities and can send the same or different spam messages in different communities.

The overlapping communities detected are shown in Fig. 2. Our main aim is to detect all the possible spammers in the network. There is a highly connected network of spammers in Community 2, which includes users 5, 6, 7, and 8, and is a spammer community. Additionally, let us assume that user 6 is hidden spammer who acts as single point to spread malicious URLs to other accounts. In Community 1, users 2, 3, and 5 can be detected as spammers based on their behavioral features. The spammers will particularly post a large number of same URLs in tweets. The "large" and "same" URLs act as our behavioral feature to detect spammers. This behavioral feature will be locally mined for that particular community.

In Community 2, users 5, 6, 7, and 8 can be detected as spammers based on their content similarity. Based on the assumption that the spam accounts are related, the URLs posted by these accounts will be similar. This content similarity is a local feature and other accounts connected to spammers will be ignored. Additionally, the quality of accounts who follow them, i.e., mainly spam accounts, will be poor. The quality or credibility of accounts can be quantitatively evaluated based on the number of followers of an account. This is a global feature, that will help to find the hidden spam accounts. It can be noted that, user 6 is a hidden spam account that does not interact with any legitimate account and will not be detected by any of the previous works. We intend to analyze the strong connections with the spammers

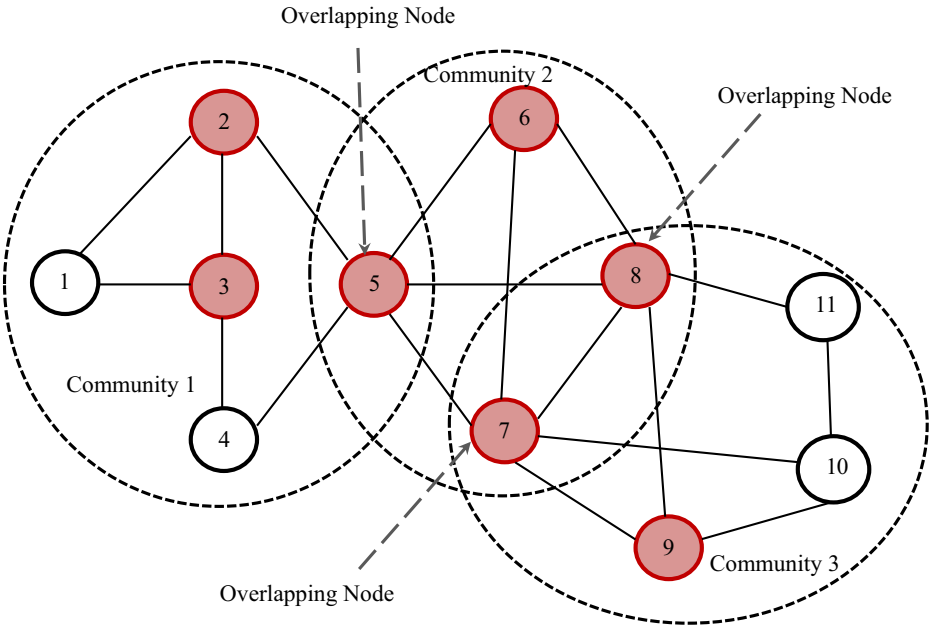


Fig. 2 Communities in the toy example

(clique formation or high local clustering coefficient) and the quality of neighborhood as a major factor to detect user 6 as spammer. In case of Community 3, user accounts 7, 8, and 9 can be detected as spammers using its local connection with spammers, topological, behavioral, and content similarity. These spammers are connected to each other (spammer clusters) and will show content similarity among each other. The large number of same URLs posted by these accounts will also help to mark these accounts as suspects. Consequently, the proposed approach will give three clusters viz., one cluster having users 2, 3, and 5, other cluster having users 5, 6, 7, and 8, and another cluster having accounts 7, 8, and 9. The spammer community containing the user accounts 5, 6, 7, and 8 is a root spammer community that spreads malicious links to users in other communities.

Using the overlapping community structure in Twitter, our aim is to identify spam accounts acting individually as well as in a community based on its content similarity, topological, behavioral, and account features. This framework helps to unearth hidden communities existing in social networks and to study the social relationships between the spammers.

5 Proposed methodology

The unsupervised approach named **SpamCom** is proposed to identify spammer communities in the network. As a first step, the efficient Link Aggregate (LA) and Improved Iterative Scan (IS^2) algorithms (Baumes et al. 2005) are used to identify the overlapping communities in the network. Then the behavioral, structural, and contextual features are used to identify certain accounts as benign or malicious. In this section, we describe **SpamCom** through which we cluster, identify, and group potential spammers into a well-formed community. Figure 3 shows the flow description of **SpamCom**. The description of each step is given below:

5.1 Identifying base spammers

As a first step towards finding the spammer communities, a set of suspect nodes that will be at the base of the attack cluster are identified. Each user in the Tweet network layer $G^T(V, E^T, U)$ is tested for a behavioral characteristic, and if it does not satisfy the minimum threshold, the user is marked as a base spammer. This behavioral property of *Unique URL ratio* is intuitively derived from the findings of related work by researchers (Lee et al. 2010). It is a fact that spammers post same URL multiple times to increase their click ratio. The spammer would want the legitimate users to visit the particular site, and would post it numerous times to get more visits. The lower the *Unique URL ratio*, the higher the chances of it being a spam account. This property is used to prune out the set of suspect nodes. We define the *Unique URL ratio* property as follows:

$$Unique\_URL\_Ratio(v) = \frac{Number\ of\ unique\ URLs(v)}{Total\ number\ of\ URLs(v)} \tag{1}$$

The set of suspect nodes that will be at the base of attack cluster is identified using Algorithm 1. The algorithm initially takes an empty set of base spammers and checks for the *Unique URL ratio* property with each user in the Tweet network layer. The ratio is compared with a *threshold*, and all users not satisfying the threshold are added to the set of base spammers. A *threshold* of 0.05, has been tested with the Honeypot dataset and found to achieve 90% precision in detecting base spammers.

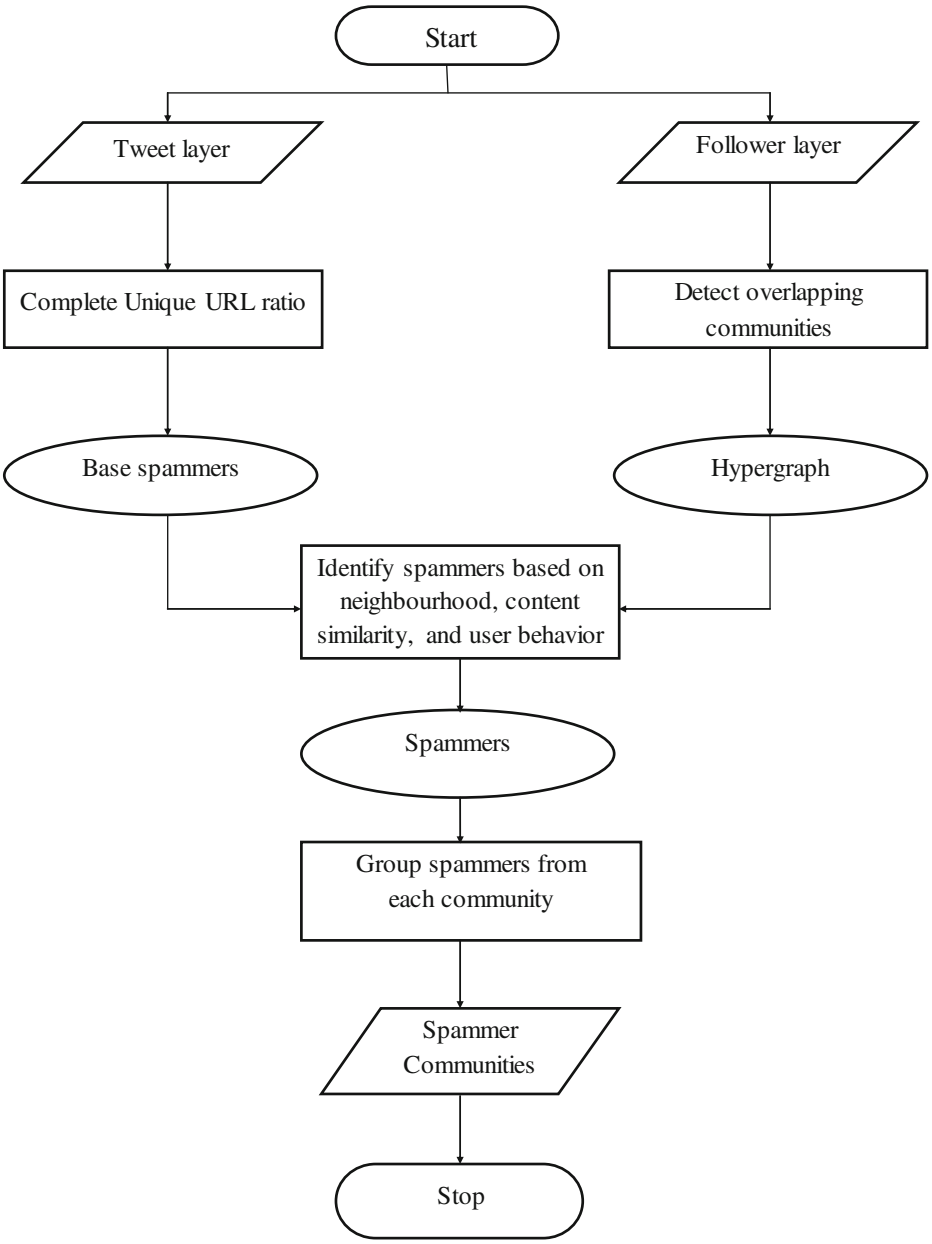


Fig. 3 Flow description of SpamCom

5.2 Detecting overlapping communities

This step involves detecting node level overlapping communities in Twitter from the Follower network layer $G^F(V, E^F, A)$ using the efficient LA and IS^2 algorithm (Baumes et al. 2005). The Follower layer involves the *following* relationship and the LA and IS^2 algorithm

does not rely on contents of the message and uses only the communication graph. Unlike the traditional community detection methods, LA and IS^2 algorithm is an overlapping community detection method which tries to discover a group of users that hide their communication, possibly for malicious reasons. Users in social networks tend to form groups and associate with people that reflect their interests. Thus, users in social networks belong to many such groups or communities. Hence, we intend to extract such groups in Follower network layer using the LA and IS^2 algorithm with primary motivation to filter out hidden malicious communities existing in the social network based on the work of Baumes et al. (2004). The LA and IS^2 algorithm handles sparse networks efficiently and identifies high quality overlapping communities in networks. The running time of LA and IS^2 algorithm is significantly less for sparse networks compared to dense networks.

The output of this step is represented as a Hypergraph. A Hypergraph is a graph where multiple nodes belong to one community or edge known as Hyperedge. It is a graph with edges containing nonempty subset of nodes. The formal definition of Hypergraph is as follows.

Hypergraph Let $H = (V, E^h)$ be a hypergraph, where V represents a finite set of nodes and E$^{\text{h}}$ the set of Hyperedges such that for any $e_i \in E, e_i \subset V$. Let H_i be a hypergraph incidence matrix with $h(v, e) = 1$, if vertex v is in edge e.

Algorithm 1 BaseSpammers(G^T)

Input: Tweet graph $G^T(V, E^T, U)$
Output: Set of base spammers

```
Base_Spammers ← φ;
for all v ∈ V in G^T do
    U ← Unique_URL_Ratio(v);
    if U ≤ threshold then
        Base_Spammers ← Base_Spammers ∪ {v};
    end if
end for
return Base_Spammers;
```

5.3 Identifying spammers in each community

In order to avoid detection by spammer detection techniques, a spammer will connect to many other spammers in the social network. As a set of base spammers have been identified, the malicious hidden communities existing in the network are to be discovered. Thus, the *FindSpammer* algorithm is introduced in Algorithm 2 to identify spammers in each community. In order to speed up the overall computation, the spammers in each community are identified in parallel by distributing the tasks to different cores of the machine. To identify spammers in each community, first the spammer suspects in the community are discovered. The intuition behind this step is that the spammers will have high local clustering coefficient with other spammers. The Hypergraph formed in the previous step $H(V, E^h)$ and the Follower network layer $G(V, E^F, A)$ are the inputs to this step. For each vertex in the hyperedge, we check if it exists in the identified set of base spammers and mark it as a suspect node. We find S as the maximum clique formed by the suspect node in the Follower layer. The neighborhood (N_S) of the maximum clique identified will consist of victims,

spammers, and legitimate users. The spammers attack in a random way to any legitimate user. Hence, the clustering coefficient of a legitimate user will be very less with a group of spammers. However, the spammers will have a high clustering coefficient among themselves. Consequently, all the nodes in N_S that have high connectivity with the identified clique S are added to suspect set.

Local clustering coefficient The local clustering coefficient for a vertex is defined as the ratio of a number of nodes it forms within its neighborhood to the number of links that can possibly exist between them. We consider the bi-directionality of links in the Follow network layer and hence the number of possible links is multiplied by a factor of 2. This metric will be used to identify how close is the vertex to the clique S. The local clustering coefficient can be defined by as:

$$LC(v, G) = \frac{2.|e^v|}{N_v.(N_v - 1)} \tag{2}$$

where, N_v is the sum of N_{fer} and N_{fing} of vertex v in graph G^F and $|e^v|$ is total number of edges built by all the neighbors of v.

Algorithm 2 FindSpammers($H, G^F, Base\_Spammers$)

Input: Hypergraph H(V,E^h), G^F(V, E^F,A), Base_Spammers
Output: Set of spammers in each community

```
1:  Spam_accounts ← φ;
2:  for all E^h ∈ H do
3:      Suspect_set ← φ;
4:      for all nodes v ∈ E^h do
5:          if v ∈ Base_Spammers then
6:              S ← MAX − CLIQUE(v);
7:              N_S ← neighborhood of S;
8:              for all nodes u ∈ N_S do
9:                  if LC(u, S) ≥ support then
10:                     S ← S ∪ {u};
11:                 end if
12:             end for
13:             Suspect_set ← Suspect_set ∪ S;
14:             spams ← Spammers(Suspect_set, v);
15:             Spam_accounts ← Spam_accounts ∪ spams
16:         end if
17:     end for
18: end for
19: return Spam_accounts;
```

Each node in N_S is checked with the local clustering coefficient. If it has a good connectivity above a threshold called *support*, the node is added to the suspect set. The spammers in the community are then identified from the spammer suspects (Algorithm 3) by using some robust features that are difficult for the spammers to evade. These feature sets comprise of content similarity, topology-based features, user behavior, and user account features. They express the role and similarity of the nodes with the identified spammers, i.e., whether the suspect sends the same set of URLs, follows the users randomly, etc. These features are taken from the attributes associated with the *Tweet* and *Follow* network layers. Each account

in the suspect set is checked with these features to extract its role in spam activity. The various features used in this paper are described as follows:

Jaccard's similarity coefficient for URLs The Jaccard index is used to compare the similarity and diversity between the suspect and spam accounts. It is known that the spammers in a community are related or use Sybil accounts to post a large amount of legitimate users with a small set of URLs. Using this intuition, the similarity and diversity between the URLs posted by spammer and suspect accounts are compared. Jaccard similarity coefficient is defined as the ratio of the size of intersection to the size of union of the sets. Henceforth, let U_{base} and U_{sus} be the URLs posted by base spammer and suspect accounts respectively. The Jaccard index for URL similarity is thus defined as:

$$J(U_{base}, U_{sus}) = \frac{|U_{base} \cap U_{sus}|}{|U_{base} \cup U_{sus}|} \tag{3}$$

Algorithm 3 Spammers(*Suspect_set, base_spammer*)

Input: Suspect_set, base_spammer, $G^F(V,E^F,A)$, $H(V,E^h)$, $G^T(V,E^T,U)$
Output: Set of spammers identified from the set of suspects

```
spammers ← ϕ;
for all v ∈ Suspect_set do
    J ← J(v, base_spammer);
    A ← ANF(v);
    U ← URL_Tweet_Ratio(v);
    age ← Age_of_Account(v);
    spam_score ← GetSpamScore(J, A, U, age);
    if spam_score > spam_threshold then
        spammers ← spammers ∪ {v};
    end if
end for
return spammers;
```

Average Neighbors' Followers *Average Neighbors' Followers* (Yang et al. 2013) is a neighbor-based feature to distinguish spammer and legitimate accounts based on account's quality of choice of friends. Let N_{fer} and N_{fing} denote the followers and followings of suspect account. The number of followers of an account usually reflects the reputation of the accounts; the more the number of followers, the better the accounts credibility. Spammers usually increase their credibility by forming a community among themselves to increase the followers. Still, the quality of accounts followed by legitimate users obviously is better compared to spammers. Additionally, this feature is found to be highly robust to evade by spammers (Yang et al. 2013). The *Average Neighbors' Followers* is defined as:

$$ANF(v) = \frac{1}{N_{fer}(v)} \cdot \sum_{u \in N_{fing}(v)} N_{fer}(u) \tag{4}$$

URL to Tweet Ratio Spammers post a large amount of URLs as compared to legitimate users. Based on this impression, we take the ratio of number of URLs posted by the suspect to the number of tweets posted by suspect. Spammers usually evade content blacklisting or keyword based filtering by content obfuscation. However, they additionally post shortened URLs to dupe the legitimate users into clicking it. If U_v is the total number of URLs posted

and $Tweet_v$ is the total number of tweets by user v, then the *URL to Tweet ratio* is defined as:

$$URL\_Tweet\_Ratio(v) = \frac{U_v}{Tweet_v} \tag{5}$$

Age of Account It has been found that the spam accounts are usually newly created compared to legitimate users. The age of an account has best discriminating power to detect spammers. Additionally, this feature cannot be evaded at all by spammers. If t_{oldest}, t_{newest}, and t_v are time-stamps for creation of oldest, newest, and suspect account, the age of account is calculated as:

$$Age\_of\_Account(v) = \frac{t_v - t_{oldest}}{t_{newest} - t_{oldest}} \tag{6}$$

Based of the above mentioned features, a spam score is calculated based on a weighted average function. The *GetSpamScore* function takes the *Jaccard's similarity coefficient for URLs*, *Average Neighbors' Followers*, *URL to Tweet Ratio*, and *Age of Account* to return a spam score. The accounts are then ranked according to the spam score. The top spammers can be highlighted using this approach. This approach for spammer detection is computationally expensive and does not scale well for large graphs as it involves computing the maximum clique of each node in the hyperedge and finding neighborhood of each spammer suspect using the topological features of the network.

5.4 Identifying connections of spammers

The main objective of this step is to find the connections of spammers between communities and to identify the nature of relationships. This will be useful to identify if spammers really have community structure, and can be used to detect the accounts that interconnect two or more communities. The Hypergraph H described above is converted to a reduced representation in the form of a line graph.

Let *L(H)* be the line graph of the Hypergraph, H. The line graph L is defined as $L(H) = (V', E')$, where $V' = E(H)$ and $E' = \{(e_1, e_2) | e_1, e_2 \in E(H), e_1 \cap e_2 \neq \phi\}$. The line graph representation helps us to identify the connections among the communities. We mark each Hyperedge E^h as corrupt if it contains a single spammer. Later, as the Hypergraph is converted to line graph, the hyperedges in Hypergraph will be converted to nodes in line graph. The resulting line graph will have nodes marked as corrupt. We find the connected subgraph component based on the marked property to identify the spread of spammers. This representation of spammers connectivity via line graph is the global behavior of spammers. Additionally, the local behavior of spammer connectivity is captured in Hyperedges.

Finally, all the connected components are identified to detect spammer communities. Every spam account behavior can be analyzed based on its local and global connectivity. Accounts having high internal and external connections with spammers need to be targeted as they try to hide in Twitter but spread malicious information through other accounts.

6 Experimental results

In this section, the experimental results of SpamCom are presented. We implemented the algorithms in R language and evaluated their accuracy and behavior for detecting spammers. The experiments were carried out on a Linux machine with a 3.40 GHz Intel Core i7

processor and 8 GB RAM. To speed up the overall computation, the tasks are distributed to multiple cores of the processor using the R *parallel* package.

The Twitter Honeypot dataset (Lee et al. 2011) is used to classify the users as spammers or legitimate users using the community, content, behavioral, and topological features. The users in the Honeypot dataset have already been classified as spammers and legitimate users. The main goal is to identify the potential effectiveness of the proposed approach in identifying spammers.

Further, the evaluation metrics used to evaluate the experimental results are presented. Then the effectiveness of features in detecting spammers is studied. Further, the results obtained by the approach when applied to the experimental setup is evaluated. Finally, the community relationships and characteristics of spammers are studied.

6.1 Twitter honeypot dataset

Twitter Honeypot dataset (Lee et al. 2011) contains tweets that were captured during the eight month period of 2010. The dataset consists of the tweets posted by users which is classified as legitimate users and content polluters or spammers by Lee et al. (2011). The dataset consists of 41,499 user accounts, with pre-classified accounts of 22,223 spammers and 19,276 legitimate users. It consists of 5,643,297 tweets in total posted by all the users in that period. As the spammers mainly spam the users by adding URLs in tweets, a script is developed to extract all the URLs existing in tweets. Totally 2,292,339 URLs from the tweets were extracted. To extract the follower relationship among the users, a web crawler was developed based on Twitter API to extract 58,750,578 social relationships among users. The basic characteristics of the dataset are shown in Table 3. The Twitter multilayer social network consisting of the *Follower* and *Tweet* layers is constructed from the Honeypot dataset.

6.2 Evaluation metrics

In order to evaluate the effectiveness of our algorithm, the following metrics are used:

- Recall, R= $\frac{d}{c+d}$
- Precision, P= $\frac{d}{d+b}$
- F-measure = 2. $\frac{P.R}{P+R}$
- Accuracy = $\frac{a+d}{a+b+c+d}$

Table 3 Characteristics of the dataset

Feature	Value
Twitter accounts	41,499
Legitimate users	19,276
Malicious users	22,223
Tweets of legitimate users	3,263,238
Tweets of malicious users	2,380,059
Total number of Tweets	5,643,297
URLs extracted	2,292,339
Links in follower layer	58,750,578

where *a* represents true positives which are the number of legitimate users correctly classified as non-spammers, *b* represents false negatives which are the number of legitimate users falsely categorized as spammers, *c* represents the number of false positives which are the spammers falsely classified as legitimate and *d* is the true negative variable representing the total number of spammers correctly classified as spammers.

Here the recall, *R*, of spammer class will show the ratio of number of users correctly classified to the number of spammers. Precision, *P*, of spam class is the ratio of number of users correctly classified correctly to the total predicted spammers by our algorithm. F-measure is the standard way to summarize the precision and recall, and it varies from 0 to 1. The accuracy provides the rate at which the algorithm classifies the results correctly. The value of 1 depicts that the entire prediction was perfect.

6.3 Evaluation of features

As mentioned in previous sections, various robust features have been used to identify the spam accounts. Apart from Jaccard index, all the features are independent of the neighborhood and community characteristics. The importance of these features in identifying the spammers is illustrated by plotting the cumulative distribution function (CDF) to depict the differences between spammers and legitimate users. The following four attributes are considered: *Age of Account*, *Average Neighbors' Followers*, *Unique URL Ratio*, and *URL to tweet Ratio*. Next, the CDFs of these attributes are shown in Fig. 4. It can be clearly noted from Fig. 4a that the age of spam accounts have low values compared to legitimate users. Spam accounts are usually newly created compared to legitimate users probably because they are constantly being blocked by other users and Twitter. Figure 4b shows that average number of followers of non-spammers is much higher as compared to spammers as they follow a good quality of accounts usually. Figure 4c shows the Unique URLs in Tweets between spammers and legitimate users. It is clearly visible that spammers have very low value of unique URLs as they repeatedly post the same URLs to their victims. Finally, Fig. 4d shows the CDF of URL to Tweet ratio between legitimate users and spammers with high discriminative power. Legitimate users have very less URL to Tweet Ratio while

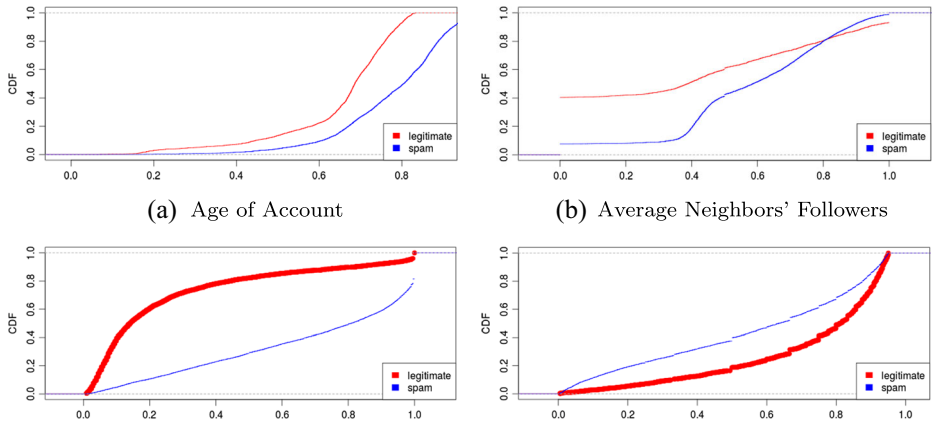


Fig. 4 Cumulative Distribution Functions of attributes for Honeypot dataset

Table 4 Performance on Twitter Honeypot Dataset

Classifier	TP Rate	FP Rate	Precision	Recall	F-Measure
ADTree	0.857	0.194	0.856	0.857	0.856
J48	0.853	0.196	0.852	0.853	0.853
IBk	0.842	0.213	0.841	0.842	0.841
SVM	0.824	0.209	0.827	0.824	0.825
Naive Bayes	0.805	0.199	0.819	0.805	0.809
SpamCom	0.867	0.132	0.895	0.867	0.880

spammers post a large amount of URLs in their tweets. In general, the analysis of these behavioral, content, and topological characteristics shows that they have the potential to differentiate spammers and legitimate users effectively.

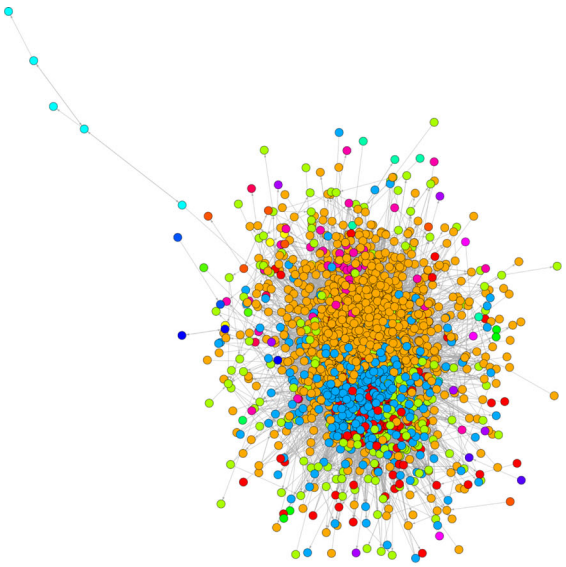
6.4 Spammer classification

In order to demonstrate the effectiveness of the proposed approach, standard machine learning classification algorithms are applied on the Social Honeypot dataset. The classification is performed based on the features calculated and described in the previous sections. The performances of five classifiers including two decision tree based (ADTree (Kohavi and Quinlan 2002), J48 (Freund and Mason 1999)), one k-nearest neighbor based (IBk (Aha et al. 1991) using k=5 nearest neighbors), Support Vector Machine based, and Naive Bayes Algorithm (John and Langley 1995) are compared with that of the proposed approach. We use 10-fold cross validation for each classification algorithm on the Honeypot dataset. The evaluation metrics obtained for the classifiers are compared with the results obtained from SpamCom in Table 4. It can be observed that the proposed approach gives better precision and recall compared to all the algorithms. The false positive rate is also the best, showing the low rate of legitimate users being classified as spammers. The F-measure is not that high due to the classification of many spammers as legitimate users. The F-measure can be further improved by lowering the threshold values.It can be concluded from the classification

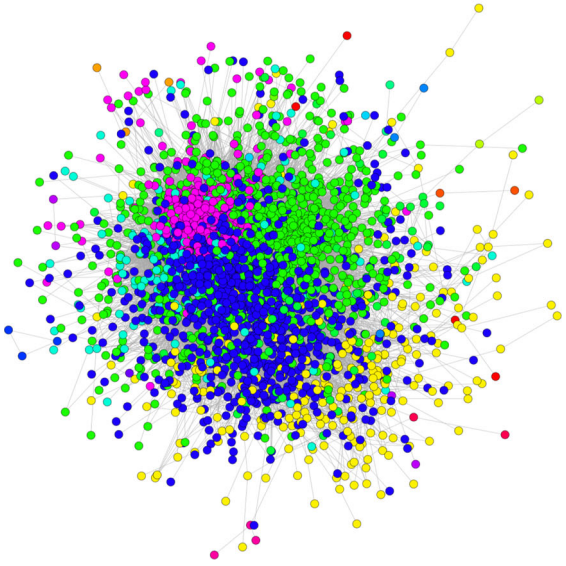
Table 5 Spammer community statistics

Feature	Value
Nodes	4047
Edges	339359
Nodes in largest WCC	3993
Edges in largest WCC	339354
Nodes in largest SCC	3495
Edges in largest SCC	85454
Average clustering coefficient	0.156007
Number of triangles	10704978
Diameter (largest shortest path)	9
Size of largest cliques in graph	35

Fig. 5 Spammer Communities in Twitter Honeypot dataset



(a) Communities in weakly connected spammers



(b) Communities in strongly connected spammers

results that the proposed approach yields better performance using the community-based features and other robust features compared to other machine learning algorithms.

6.5 Community structure

The experimental results are concluded by analyzing the community structure of spammers. Initially, a subgraph of spammers from the *Follower* network layer $G^F(V, E^F, A)$ is constructed. The spammer graph is denoted as S. The graph is decomposed into clusters based on strong and weak connections. The weak connections form 51 clusters with a single cluster of size 3993, while the remaining clusters consist of only one or two spammers. Similarly, the strong connections form a total of 421 clusters with a single cluster of size 3495, whereas other clusters consist of only one or two spammers. The statistics of these strong and weak components of spammer network is described in Table 5. The table shows the nodes and edges in weakly connected components (WCC) and strongly connected components (SCC) in spammer network. The average clustering coefficient is not significantly high, showing the low number of triangles formed between spammers. There are two large cliques of size 35 in the spammer network showing the large highly connected spammer communities existing in social networks.

The spinglass community algorithm (Reichardt and Bornholdt 2006) is applied to find the community structure existing in networks. We identify 18 and 19 communities existing in spammer network for unidirectional and bidirectional links respectively. The community structure of the spammers is shown in Fig. 5 with weakly connected and highly connected components. The figure shows the density, spread, and communication relationship of spammers.

Based on the experimental results, it is evident that there are communities of spammers working collectively to spread spam and evade spammer detection techniques. Hence, there is an urgent need to detect and curb the formation of such communities to enhance the user experience in social networks.

7 Conclusion

A novel and robust approach called **SpamCom** to detect spammer communities based on overlapping community structure, topological, behavioral, and content attributes in the online social network Twitter is proposed in this paper. After identifying overlapping community structure existing in Twitter, the suspects are identified based on content similarity and connectivity with spammer accounts. Finally, the spammers are identified from the set of suspects based on content, age of account, neighborhood, and behavioral attributes of each user. The dual behavior of spammers to pose as legitimate users and perform malicious activities is overcome using this approach. The identified spammers are clubbed together to identify the core spammer network spread in social networks. Our aim is to identify the hidden communities and to study them in detail to tackle the significant problem of spammers in Twitter. Even though the proposed approach needs evaluation in much finer detail, the preliminary experiments show significant performance in detecting spammers. Additionally, this is the first effort to study the spammer community structure existing in social networks. In future, we aim to provide much detailed and extended study of our approach and its performance in real-world scenario. More specifically, the Honeypot dataset cannot precisely represent the real Twitter ecosystem, and the *Follower* network layer constructed

from Honeypot dataset is not complete. Hence, collecting the real Twitter data from streaming API and crawling user profiles for the finer evaluation of the approach is a future work.

References

Aha, D.W., Kibler, D., Albert, M.K. (1991). Instance-based learning algorithms. *Machine learning*, *6*(1), 37–66.

Baumes, J., Goldberg, M., Magdon-Ismail, M. (2005). Efficient identification of overlapping communities. In *International Conference on Intelligence and Security Informatics* (pp. 27–36). Springer.

Baumes, J., Goldberg, M., Magdon-Ismail, M., Al Wallace, W. (2004). Discovering hidden groups in communication networks. In *International Conference on Intelligence and Security Informatics* (pp. 378–389). Springer.

Benevenuto, F., Magno, G., Rodrigues, T., Almeida, V. (2010). Detecting spammers on twitter. In *Collaboration, electronic messaging, anti-abuse and spam conference (CEAS)* (Vol. 6 p. 12).

Benevenuto, F., Rodrigues, T., Almeida, V., Almeida, J., Zhang, C., Ross, K. (2008). Identifying video spammers in online social networks. In *Proceedings of the 4th international workshop on adversarial information retrieval on the web* (pp. 45–52). ACM.

Bhat, S.Y., & Abulaish, M. (2013). Community-based features for identifying spammers in online social networks. In *Proceedings of the 2013 IEEE/ACM international conference on advances in social networks analysis and mining* (pp. 100–107). ACM.

Bindu, P.V., & Thilagam, P.S. (2016). Mining social networks for anomalies: Methods and challenges. *Journal of Network and Computer Applications*, *68*, 213–229. https://doi.org/10.1016/j.jnca.2016.02.021.

Bindu, P.V., Thilagam, P.S., Ahuja, D. (2017). Discovering suspicious behavior in multilayer social networks. *Computers in Human Behavior*, *73*, 568–582. https://doi.org/10.1016/j.chb.2017.04.001.

Bródka, P., & Kazienko, P. (2014). *Encyclopedia of social network analysis and mining, chap. Multilayered social networks* (pp. 998–1013). New York: Springer. https://doi.org/10.1007/978-1-4614-6170-8_239.

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

DeBarr, D., & Wechsler, H. (2009). Spam detection using clustering, random forests, and active learning. In *6th conference on email and anti-spam*. Mountain view: Citeseer.

Facebook (2016). Facebook company-info. http://newsroom.fb.com/company-info/.

Fire, M., Katz, G., Elovici, Y. (2012). Strangers intrusion detection-detecting spammers and fake proles in social networks based on topology anomalies. *HUMAN*, *1*(1), 26.

Freund, Y., & Mason, L. (1999). The alternating decision tree learning algorithm. In *Proceedings of the sixteenth international conference on machine learning* (Vol. 99 pp. 124–133): Morgan kaufmann.

Gao, H., Hu, J., Wilson, C., Li, Z., Chen, Y., Zhao, B.Y. (2010). Detecting and characterizing social spam campaigns. In *Proceedings of the 10th ACM SIGCOMM conference on internet measurement* (pp. 35–47). ACM.

Ghosh, S., Viswanath, B., Kooti, F., Sharma, N.K., Korlam, G., Benevenuto, F., Ganguly, N., Gummadi, K.P. (2012). 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.

Grier, C., Thomas, K., Paxson, V., Zhang, M. (2010). @ spam: the underground on 140 characters or less. In *Proceedings of the 17th ACM conference on computer and communications security* (pp. 27–37). ACM.

Haythornthwaite, C. (2005). Social networks and internet connectivity effects. *Information, Communication & Society*, *8*(2), 125–147. https://doi.org/10.1080/13691180500146185.

Hu, X., Tang, J., Zhang, Y., Liu, H. (2013). Social spammer detection in microblogging. In *Proceedings of the twenty-third international joint conference on artificial intelligence* (vol. 13 pp. 2633–2639). AAAI Press.

John, G.H., & Langley, P. (1995). Estimating continuous distributions in bayesian classifiers. In *Proceedings of the eleventh conference on uncertainty in artificial intelligence* (pp. 338–345). Morgan kaufmann.

Kohavi, R., & Quinlan, J.R. (2002). Data mining tasks and methods: Classification: decision-tree discovery. In *Handbook of data mining and knowledge discovery* (pp. 267–276). Oxford University Press Inc.

Lee, K., Caverlee, J., Webb, S. (2010). 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.

Lee, K., Eoff, B.D., Caverlee, J. (2011). Seven months with the devils: a long-term study of content polluters on twitter. In *Proceedings of 5th international AAAI conference on weblogs and social media (ICWSM)*.
Martinez-Romo, J., & Araujo, L. (2013). Detecting malicious tweets in trending topics using a statistical analysis of language. *Expert Systems with Applications*, *40*(8), 2992–3000.
Mustafaraj, E., & Metaxas, P.T. (2010). From obscurity to prominence in minutes: Political speech and real-time search. In *In proceedings of the WebSci10: extending the frontiers of society on-line*.
Reichardt, J., & Bornholdt, S. (2006). Statistical mechanics of community detection. *Physical Review E*, *74*(1), 016110.
Ratkiewicz, J., Conover, M., Meiss, M., Gonçalves, B., Flammini, A., Menczer, F. (2011). Detecting and tracking political abuse in social media. In *Proceedings of 5th international AAAI conference on weblogs and social media* (pp. 297–304).
Ratkiewicz, J., Conover, M., Meiss, M., Gonçalves, B., Patil, S., Flammini, A., Menczer, F. (2011). Truthy: mapping the spread of astroturf in microblog streams. In *Proceedings of the 20th international conference companion on world wide web (ICWSM),* (pp. 249–252). ACM.
SciTechBlog (2016). Scitechblog. http://scitech.blogs.cnn.com/.
Shrivastava, N., Majumder, A., Rastogi, R. (2008). Mining (social) network graphs to detect random link attacks. In *IEEE 24th international conference on data engineering, 2008. ICDE 2008.* (pp. 486–495). IEEE.
Song, J., Lee, S., Kim, J. (2011). Spam filtering in twitter using sender-receiver relationship. In *Recent advances in intrusion detection* (pp. 301–317). Springer.
Stringhini, G., Kruegel, C., Vigna, G. (2010). Detecting spammers on social networks. In *Proceedings of the 26th annual computer security applications conference* (pp. 1–9). ACM.
Swamynathan, G., Wilson, C., Boe, B., Almeroth, K., Zhao, B.Y. (2008). Do social networks improve e-commerce?: a study on social marketplaces. In *Proceedings of the 1st workshop on online social networks* (pp. 1–6). ACM.
Thomas, K., Grier, C., Song, D., Paxson, V. (2011). 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.
Twitter (2016). Twitter company-info. https://about.twitter.com/company.
Wang, A.H. (2010). Don't follow me: Spam detection in twitter. In *proceedings of the 2010 international conference on Security and cryptography (SECRYPT)* (pp. 1–10). IEEE.
Yang, C., Harkreader, R., Gu, G. (2013). Empirical evaluation and new design for fighting evolving twitter spammers. *IEEE Transactions on Information Forensics and Security*, *8*(8), 1280–1293. https://doi.org/10.1109/TIFS.2013.2267732.
Yang, C., Harkreader, R., Zhang, J., Shin, S., Gu, G. (2012). 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.
Yang, C., Zhang, J., Gu, G. (2014). A taste of tweets: reverse engineering twitter spammers. In *Proceedings of the 30th annual computer security applications conference* (pp. 86–95). ACM.
Yardi, S., Romero, D., Schoenebeck, G., et al. (2009). Detecting spam in a twitter network. *First Monday* 15(1).
Ying, X., Wu, X., Barbará, D. (2011). Spectrum based fraud detection in social networks. In *Proceedings of the 27th international conference on data engineering, ICDE 2011, April 11–16, 2011, Hannover, Germany, pp. 912–923*, https://doi.org/10.1109/ICDE.2011.5767910.
Zheng, X., Zeng, Z., Chen, Z., Yu, Y., Rong, C. (2015). Detecting spammers on social networks. *Neurocomputing*, *159*, 27–34. https://doi.org/10.1016/j.neucom.2015.02.047.