GRAPH FEATURE BASED MULTI-LAYER SOCIAL NETWORK ANALYSIS FOR ANOMALY DETECTION

Thesis

Submitted in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

by

BINDU P. V.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY KARNATAKA

SURATHKAL, MANGALORE - 575 025

March, 2018

DECLARATION

by the Ph.D. Research Scholar

I hereby *declare* that the Research Thesis entitled **Graph Feature Based Multilayer Social Network Analysis for Anomaly Detection** which is being submitted to the **National Institute of Technology Karnataka, Surathkal** in partial fulfillment of the requirements for the award of the Degree of **Doctor of Philosophy** in Department of Computer Science and Engineering is a *bonafide report of the research work carried out by me*. The material contained in this Research Thesis has not been submitted to any University or Institution for the award of any degree.

Bindu P. V., CS14F01
Department of Computer Science and Engineering

Place: NITK, Surathkal
Date: March 12, 2018

CERTIFICATE

This is to *certify* that the Research Thesis entitled **Graph Feature Based Multilayer Social Network Analysis for Anomaly Detection** submitted by **Bindu P. V.** (Register Number: 145043 CS14F01) as the record of the research work carried out by her, is *accepted as the Research Thesis submission* in partial fulfilment of the requirements for the award of degree of **Doctor of Philosophy**.

Dr. P. Santhi Thilagam
Research Guide

(Signature with Date and Seal)

Chairman - DRPC
(Signature with Date and Seal)

To my family for their constant support
and
unconditional love

Acknowledgements

It is a great pleasure to thank the people who have supported and helped me so much throughout the period of my research. First and foremost, I would like to express my sincere gratitude to my research supervisor Dr. P. Santhi Thilagam, Associate Professor and Head of the Department of Computer Science and Engineering, for constantly supporting me throughout the journey of my research with her knowledge, experience, and patience. I owe my deepest gratitude to her for the valuable guidance, timely suggestions, and advices. Beyond the subject area of this thesis, I have learnt a lot from her that, I am sure, will be useful in the later stages of my career as well as my life.

I would like to extend my sincere thanks to the members of my research progress assessment committee Dr. Alwyn R. Pais and Dr. Jaidhar C. D. for their insightful feedback and constructive suggestions throughout the period of my research work. I also extend my sincere thanks to the entire faculty of the Department of Computer Science and Engineering for rendering their help and support throughout the period of my research. Nevertheless, I am also grateful to the technical and administrative staff of the department for their timely help and co-operation to carry out the research work.

I would like to express my sincere thanks to NITK for providing the necessary infrastructure and facilities required for the successful completion of this research work. I also extend my sincere gratitude to All India Council for Technical Education, Govt. of India, and the Directorate of Technical Education, Govt. of Kerala, for providing me the opportunity to do the research work under the Quality Improvement Programme of Govt. of India.

Special thanks to G. Deepa for being with me throughout the period of my research as a pillar of incredible support. I warmly thank my friends Fathima, Sivakumar, Amit, and Ambikesh for their encouragement and for the good times we spent in the department. I would like to express my gratitude to my other fellow doctoral students and friends for their healthy criticism and valuable co-operation. I also thank Deepesh Ahuja and Rahul Mishra for their technical assistance.

This thesis would not have been possible without the exceptional support of my family. I am indebted to my grandmother, my parents, my siblings, my in-laws, my husband, and my kids for their unconditional love, inspiration, and encouragement over the years.

Finally, I thank the Almighty for granting me the health and strength for carrying out this research work.

Bindu P. V.

Abstract

Online social networks have received a dramatic increase of interest in the last decade due to the growth of the Internet and Web 2.0. They provide convenient platforms for people to share, communicate, and collaborate in real-time regardless of the differences and geographic distances among them. However, with the openness and the diversity of the users of social networks, malicious users turn online social networks into platforms for illicit activities such as spamming, identity theft, cyber-attacks, organized crimes, and even terrorist attack planning. Discovering such suspicious and illicit behavior in social networks is a significant and challenging problem in social network analysis. The unusual behavior of users that cause harm to legitimate users can be identified by using anomaly detection techniques. The major categories of anomalies occurring in social networks are point anomalies and collective or group anomalies. Point anomalies or anomalous nodes signify the unusual behavior of individual users whereas collective anomalies signify the unusual behavior of groups of users. As these two types of anomalies can signify illegal and illicit behavior, they are to be detected to uncover such suspicious behavior. Several techniques and tools have been proposed for detecting point and collective anomalies in social networks. These techniques and tools are developed for single-layer social networks with only one type of interaction among the individuals. However, the social relationships among individuals are more complex and they interact with each other in multiple ways simultaneously leading to multiple networks among the same set of individuals, or a multi-layer social network with each layer representing one type of interaction. The analysis of only one type of interaction for anomaly detection does not provide a complete picture of the relationships among the users of the networks. Therefore, there is an urgency and need for multi-layer analysis of the networks for identifying the anomalies by employing the rich information hidden in the individual network layers. Hence, this work aims at developing approaches for detecting point and collective anomalies in multi-layer social networks.

In social networks, if the neighborhood of a user is a clique/near-clique or a star/near-star pattern, the online behavior of the user can be linked to an anomalous behavior, as only minority of users follow these patterns. In a multi-layer social network, if the neighborhoods of nodes in different layers are close to stars or cliques, they can signify anomalous behavior. Hence, in this work, an unsupervised approach called Anomaly Detection On Multi-layer Social networks (ADOMS) is proposed for detecting these point anomalies in multi-layer social networks, by using graph-theoretic features of the networks and data mining techniques. The online behavior of users is modeled as an unattributed multi-layer social network, and the network structure-based features of the network are extracted to detect anomalies. Anomaly scores are computed for the nodes of the multi-layer network and the nodes are then ranked based on their anomalousness. The nodes with high anomaly scores are the top ranked anomalies. The proposed approach is evaluated using extensive experiments on multiple real-world multi-layer network datasets, and the experimental results substantiate that the approach can effectively detect anomalous nodes in multi-layer social networks.

Spamming is the most predominant form of anomalous activity prevalent in online social networks that involves malicious users sending unsolicited messages to legitimate users with the intention of wasting their time, bandwidth, and money. Being one of the fastest growing online social networks, Twitter has become a cardinal target platform for social spammers. A substantial amount of research work has been carried out in the field of detecting spam messages and social spammers in Twitter. However, one of the important issues in Twitter is that the social spammers collaborate with each other and form collective anomalies or spammer communities to spread spam messages to a large set of legitimate users. Consequently, it is highly desirable to detect such spammer communities prevailing in Twitter. Hence, in this work, an unsupervised approach called Spammer Community detection (SpamCom) is proposed for detecting spammer communities in Twitter by using graph-theoretic features of the network and the network attributes. The Twitter network is modeled as an attributed multi-layer social network, and the overlapping community-based features of the network are exploited to identify spammers based on their structural behavior and URL characteristics. The utilization of community-based features of the network, URL characteristics of user accounts, and content similarity among the tweets makes the approach robust and efficient. The approach is evaluated on real-world dataset, and the experimental results show significant performance in detecting spammers and spammer communities.

***Keywords*:** Social network analysis, Anomaly detection, Outlier detection, Graph mining, Graph-based anomaly detection, Multi-layer networks, Spammer detection, Spammer communities,

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

LIST OF SYMBOLS

Symbol	**Description**
G	Multi-layer network
V	Set of nodes in G
L	Number of layers in G
n	Number of nodes in G
G^l	l^{th} network layer of G
E^l	Set of edges in G^l
A_G	Set of $n \times n$ adjacency matrices corresponding to G
$A^{[l]}$	$n \times n$ adjacency matrix of G^l with elements $a_{ij}^{[l]}$
N_i^l	Number of nodes in the egonet of node i in G^l
E_i^l	Number of edges in the egonet of node i in G^l
$aScore_i^l$	Anomaly score for node i in G^l
LR_i^l	Layer relevance of layer l for node i
$multiScore_i$	Anomaly score of node i in the multi-layer network G
G_A	Aggregated topological network of G
A	Adjacency matrix corresponding to G_A
M	Twitter multi-layer social network
G^F	Follower network layer of M
G^T	Tweet network layer of M
i, j	Node indices
U	Set of all URLs posted by the users in the dataset
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

CHAPTER 1

INTRODUCTION

A social network is a social structure that represents the relationships or associations among social entities such as friends, professionals, or co-authors. The last decade has witnessed a proliferation of social networks supported by variety of information communication technologies and web-based services. Examples of social networks include online social networks (Mislove et al. 2007), mobile call networks (Nanavati et al. 2006), instant messenger networks (Leskovec and Horvitz 2008), email communication networks (Diesner et al. 2005), terrorist networks, and co-authorship networks (Barabsi et al. 2002), to name a few.

Over the last few years, online social networks have become more and more pervasive and have received significant attention and popularity due to the advancement of the Internet and Web 2.0. They are among the most popular sites on the Internet that have become integral part of our day-to-day life. They are being used in almost all areas of life including education, entertainment, medical, and business. Through online social networks, there are plenty of possibilities for individuals to share contents, interact, and collaborate with each other irrespective of the geographical distances among them (Pedrycz and Chen 2013). In addition, they enable users to maintain social relationships, to find users with similar interests, and to access information that have been shared by other users (Mislove et al. 2007). Because of these reasons, in recent years, online social networks have received tremendous attention by both academic and industries.

Social Network Analysis (SNA) (Wasserman and Faust 1994) is the field of study which investigates the properties of social networks. SNA helps us to explore the relationships between individuals who are connected through social networks, and provides understanding about the inherent patterns that are embedded in these networks (Scott 2011). SNA concerns a variety of tasks. Some of the most significant tasks of SNA are centrality analysis (Carrington et al. 2005), community detection (Girvan and Newman 2002; Arab and Afsharchi 2014), information diffusion (Bakshy et al. 2012; Haralabopoulos et al. 2015), influence maximization (Chen et al. 2009), link prediction (Liben-Nowell and Kleinberg 2007), recommender systems (Wang et al. 2015), and anomaly detection.

Social networks are typically modeled as graphs, and SNA investigates the properties of social networks through the use of graph theory. A graph is a powerful representation framework for a complex network (Boccaletti et al. 2006), with nodes or vertices representing entities and edges representing the interactions among them. More formally, a graph is defined as $G(V, E)$, where V denotes the set of nodes and E denotes the set of edges such that $e_k \in E$ and $e_k = \{(v_i, v_j)|v_i, v_j \in V\}$. Social network graphs can be classified into directed/undirected based on the edge direction (Robins et al. 2009), weighted/unweighted based on edge weights (Newman 2004; Das et al. 2010), signed/unsigned based on edge sign (Leskovec et al. 2010), or attributed/unattributed based on edge and/or node attributes.

Each of the online social networks is a huge database of millions of individuals and their activities. Unfortunately, due to the openness of these networks, this enormous amount of information attracts the interest of malicious users (Fire et al. 2014). Consequently, the perversion of these networks has also increased and has opened up the door for numerous malicious and predatory activities such as spamming, identity theft, collaborative fraud, cyber attacks, cyberbullying, organized crimes, terrorist attack planning, and fraudulent information dissemination (Jiang et al. 2016; Keyvanpour et al. 2014; Liu and Chawla 2015; Al-garadi et al. 2016; Asam and Samara 2016; Yu et al. 2015, 2016). As the perpetrators of these unusual and irregular activities often interact in a manner that significantly differs from the common public, they can be discovered by employing anomaly detection techniques. Anomaly detection in social networks refers to the problem of identifying the strange and unexpected behavior of users by analyzing the patterns hidden in the networks. The knowledge about these unusual and irregular patterns leads to the identification of suspicious users and illegal activities in social networks. For example, online fraudsters often interact with many individuals who are otherwise unconnected to form star-like patterns in the network. These anomalous patterns can be identified by analyzing a set of network features. As social networks are mathematically represented as graphs, the abnormal behavior of users in the networks can be identified by using graph-based anomaly detection techniques.

1.1 ANOMALY DETECTION

An anomaly or outlier is a data object that behaves substantially different from the other objects in the dataset (Chandola et al. 2009). The classic definition of an outlier is "an observation which deviates so much from other observations as to arouse suspicions that it was generated by a different mechanism" according to Hawkins (1980), or "an observation (or subset of observations) which appears to be inconsistent with the remainder of that set of data" according to Barnett and Lewis (1994). These abnormal patterns or observations are also known as aberrations, peculiarities, discordant observations, contaminants surprises, or exceptions based on the application areas (Chandola et al. 2009). Anomalies occur because of fraudulent behavior, human error, or faults in systems (Hodge and Austin 2004).

Anomaly detection or outlier detection is a key problem in data mining and is defined as the problem of detecting anomalous patterns in a dataset. Whereas other data mining algorithms such as classification, clustering, and frequent pattern analysis find popular patterns from the dataset, anomaly detection identifies relatively small set of objects that differ from the normal behavior of the larger number of data items in the dataset (Dang et al. 2014).

Anomaly detection is a well-researched topic in various research domains such as data mining, statistics, sensor networks, distributed systems, spatio-temporal mining, environmental science, etc (Gupta et al. 2014a). It has attracted wide recognition in many practical application areas including fraud detection for insurance, credit cards, stock markets, or health care, cyber security and intrusion detection in computer networks, military surveillance for enemy activities, fault diagnosis in safety critical systems, and pharmaceutical research for identifying novel molecular structures.

Some of the challenges that are encountered while detecting anomalies are as follows (Chandola et al. 2009):

- It is often difficult to define a clear separation between normal and anomalous behavior.

- In the case of anomalies that arise from illegal activities, the adversaries often pretend to be legitimate users. It complicates the process of defining the normal behavior.
- Normal behavior of data may change over time which makes the current notion of normal behavior irrelevant in the future.
- The exact definition of anomalous behavior depends on the application domain. Therefore, the anomaly detection techniques depend on the application domain for which it is developed, and hence cannot be applied directly to other domains.
- Due to the unavailability of datasets with ground truth, it is hard to perform the performance evaluation of anomaly detection algorithms.
- In most cases, datasets contain noise that are close to the anomalies which makes it difficult to differentiate and eliminate anomalies.

Due to these challenges, it is difficult to develop a general-purpose anomaly detection system. Hence, application-specific anomaly detection approaches need to be formulated by taking into consideration of various aspects such as the type of data, availability of ground truth, and the type of anomalies to be identified.

1.2 GRAPH-BASED ANOMALY DETECTION

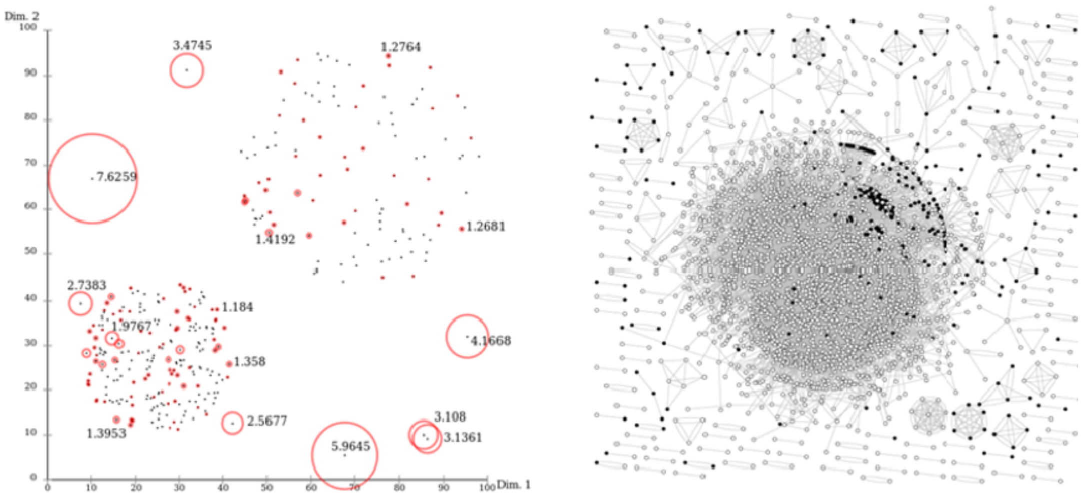
Graph-based anomaly detection is the problem of finding anomalies from data that are represented as graphs, by using graph mining techniques. Graph mining, which has been a popular area of research in recent years, is the novel approach for extracting useful knowledge hidden in graph data by employing techniques from domains such as graph theory, machine learning, data mining, statistics, and pattern recognition (Jiang 2011). Substantial research works have been directed towards graph mining because of its various applications in a multitude of practical domains including bioinformatics, chemical compound analysis, program flow structures, computer networks, and online social networks. In literature, several graph mining approaches have been proposed, including: frequent subgraph mining (Cook and Holder 1994; Kuramochi and Karypis 2005, 2001; Dehaspe et al. 1998), graph pattern summarization (Xin et al. 2006; Chen et al. 2008; Chaoji et al. 2008), approximate graph pattern mining (Chen et al. 2007b; Kelley et al. 2003), graph classification (Saigo et al. 2009; Jin et al. 2009; Deshpande et al. 2005), graph clustering (Newman 2012; Xu et al. 2012; Clauset 2005; Kulis et al. 2009), graph indexing (Wang et al. 2012b; Yuan et al. 2012), graph searching (Chen et al. 2007a; Yan et al. 2005), and graph-based anomaly detection (Eberle and Holder 2007; Akoglu et al. 2010).

Graph-based anomaly detection attempts to find anomalous units such as nodes, edges or subgraphs in graph-structured data. More specifically, graph-based anomaly detection techniques identify abnormal substructures from graphs, the existence of unexpected nodes or edges, the absence of expected nodes or edges, or modifications in the attributes of nodes or edges (Eberle and Holder 2007). As graph based data has become omnipresent, it is essential to detect anomalies from graph data for addressing problems such as activity monitoring in social networks, network intrusion detection, network surveillance, spam filtering, biomedical imaging, gene network analysis, churn analysis in social networks, disease outbreak detection, sensor network monitoring, environmental monitoring, and malware detection (Sharpnack et al. 2013; Sricharan and Das 2014).

Even though the traditional anomaly detection problem has been actively studied in statistics and data mining communities over the last several decades, there has been much less focus on anomaly detection on graph-based data. Unlike the traditional anomaly detection methods that analyze independent and identically distributed data objects or data points, graph-based anomaly detection techniques have to consider the inter-related data objects having long range correlations (See Figure 1.1). Therefore, the algorithms and techniques formulated for traditional anomaly detection problems are not directly applicable to graph data. Moreover, due to the powerful representation capability of graphs, the definitions of anomalies is much more diverse and typically depend on the data and application of interest (Akoglu et al. 2015). In addition, in order to enumerate anomalous units in graphs, the detection algorithms have to search the entire complex graph, due to which the time and space complexities of the detection algorithms are often high. Due to these challenges, it is hard to develop an efficient and scalable technique for graph-based anomaly detection.

1.3 ANOMALY DETECTION IN SOCIAL NETWORKS

Social network anomaly detection is of crucial importance to detect the malicious activities in social networks. In social networks arena, an anomaly is an unexpected behavior



(a) Clouds of points (multi-dimensional)

(b) Inter-linked objects (network)

Figure 1.1: Traditional anomaly detection vs. graph-based anomaly detection (Akoglu et al. 2015)

of a user or group of users whose behavior is unusual compared to the normal behavior of the users in the network. The anomalies occur when the patterns of interactions of certain individuals deviate significantly from their peers. As a consequence of these irregular interactions, the network structure changes, and the parts of the network that are significantly different from the normal structure are deemed as anomalies. More specifically, anomalies in social networks occur when the behavior of users forms unusual network patterns (Savage et al. 2014).

Anomalies prevailing in social networks can be mainly classified into two categories: anomalous nodes or point anomalies and anomalous subgraphs or collective anomalies. Point anomalies are individual nodes or users whose online behavior is abnormal compared to other users, whereas collective anomalies appear as groups of nodes or users in social networks. In a collective anomaly, the individuals in the group collaborate with each other to achieve a malicious goal. These two types of anomalies can signify illegal and irregular behavior of users and need to be detected to unveil such suspicious user behavior. Even though not all anomalies are malicious, anomalies in social networks can be representatives of illicit and fraudulent activities and can harm users in the virtual world as well as in the real world. Therefore, the detection of such anomalies has been used to identify illegal and fraudulent activities in social networks.

Anomaly detection in social networks attempts to identify anomalous entities in the networks such as abnormal nodes, abnormal interactions, or abnormal subgraphs using graph-based anomaly detection techniques. As the entities in a social network are highly inter-related, the anomaly detection methods have to examine the interactions in the network to identify anomalous units. Hence, the traditional anomaly detection techniques developed for multi-dimensional points can not be directly applied to social networks. Moreover, the anomaly detection techniques on social networks have to analyze the whole network data to reveal the anomalies, which makes the detection process complex in terms of both computational time and space requirements.

In social networks, the analysis of the users' local and global views helps in generating the online behavior patterns of the users in the networks. The local view of the users specifies local connectivity patterns of them and global view generalizes the rules for identifying the deviation of the users' online behavior. More specifically, if the local view of a user obeys the rule characterized by the global view of the network, the user's online behavior is categorized as "normal"; otherwise it is categorized as "anomalous".

Anomaly detection has become a significant and challenging problem in SNA. In recent years, anomaly detection in social networks has received remarkable interest from the research community and a multitude of algorithms, techniques, and tools have been proposed. These techniques have been proposed for networks with only one type of interaction among individuals. However, individuals in a social system can interact in multiple ways simultaneously, leading to the formation of *multi-layer social networks* (Bródka and Kazienko 2014). Even though several techniques and tools have been developed for anomaly detection in single-layer social networks, anomaly detection in multi-layer social networks is an unexplored area of research.

1.4 MULTI-LAYER NETWORKS

Over the last few years, complex networks (Boccaletti et al. 2006) have attracted enormous attention in a wide variety of domains including social, biological, physical, information, and engineering sciences. The study of complex networks has been traditionally based on graph theory, representing only single type of interaction among entities. However, real-world systems are more complex and multiple types of interaction can exist among the same set of entities which can not be completely captured by conventional simple graph models. Examples are social systems 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 these systems, each layer is associated with a given relevance or meaning and treating all layers as similar results in the lose of important information present in individual layers. This has led to the development of *multi-layer networks* (Kivelä et al. 2014; De Domenico et al. 2013; Boccaletti et al. 2014). A multi-layer network is an emergent model that can encode much richer and relevant information than a single-layer network.

The standard way of representing a multi-layer network is to aggregate the information present in different layers together to obtain a single-layer network so that the traditional network analysis tools can be directly applied. However, this is a crude approximation and results in the loss of important information hidden in the individual layers. Even though a large body of literature exists for single-layer networks, there is a deficiency of widely-accepted methods and means for multi-layer network analysis. Therefore, it is extremely important to develop network analysis tools for multi-layer networks.

Even though the study of multi-layer networks originated decades ago from social sciences, they were not analyzed at the large scale. Recently, multi-layer networks have become extremely popular in multi-disciplinary research domains and a substantial amount of effort has been dedicated to their mathematical modeling and characterization. The analysis of multi-layer networks is an ongoing research area of network science, and the sudden increase in the study of such networks has resulted in a multitude of terminologies such as multi-layered networks (Bródka and Kazienko 2014), multiplex networks (Battiston et al. 2014; Lee et al. 2015), multi-relational networks (Harrer and Schmidt 2012), multidimensional networks (Berlingerio et al. 2013a), and interconnected networks (Dickison et al. 2012).

1.5 MULTI-LAYER SOCIAL NETWORKS

Multi-layer social networks represent multiple types of interactions occurring in social systems among the same set of individuals (Magnani and Rossi 2011; Bródka and Kazienko 2014). 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. Individuals may interact through phone calls, instant messaging, emails, online social networks, and so on. 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.

Each layer of a multi-layer social network describes one mode of interaction among individuals. In online social media, people can connect through multiple social networking platforms like Twitter, Facebook, Instagram, LinkedIN, YouTube, and Google plus, to name a few. 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. In such cases, a multi-layer social network can be constructed with each layer representing the interactions at one service, and multi-layer network analysis is required for distinguishing between the interactions in different layers. Most classical SNA tools deal with single type of interaction between individuals. Recently, many of the classical SNA metrics and problems have been extended to multi-layer social networks.

A simple graph representation can model a singe-layer network effectively and is highly useful in modeling many social phenomena. However, it can not represent the multiple ties existing among the users of a social network. Hence, the simultaneous interactions among users is modeled as a multi-layer social network M, containing 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 multi-layer social network can be considered as a network on its own, and the i^{th} layer of the multi-layer 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 multi-layer 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. An example for a three-layer multi-layer social network is shown in Figure 1.2.

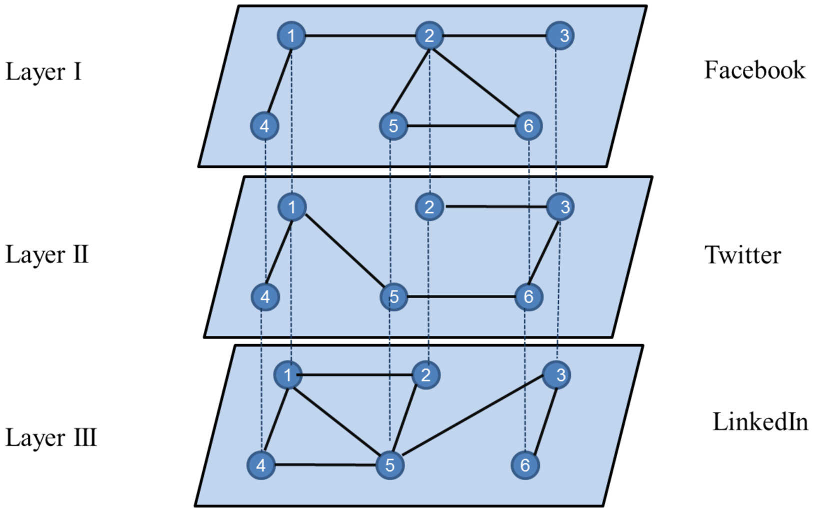


Figure 1.2: Multi-layer social network with three layers

1.6 MOTIVATION

Anomalies in social networks often represent illegal activities and security issues such as spamming, identity theft, cyber attacks, organized crimes, bullying, fraudulent information dissemination, and even terrorist attack planning (Keyvanpour et al. 2014; Liu and Chawla 2015; Yu et al. 2016). Anomalies in social networks could also signify unusual user behavior such as credit card fraud, electronic auction fraud, email spam and phishing (Akoglu et al. 2010), and many others. Therefore, it is extremely important to detect these anomalous behaviors.

The criminal activities prevalent in social networks necessitates the importance of digital forensics in social networks arena. Anomaly detection can provide inherent and invaluable information for detecting criminal activities in social networks for Social Network Forensics (SNF) (Keyvanpour et al. 2014). SNF is an emerging subject that involves the detection, analysis, prevention, and prediction of illegal and criminal activities in social networks. Apart from identifying fraudulent, distrustful, or dangerous behavior, anomaly detection is also helpful in identifying influential individuals and rare events in a network.

The problem of anomaly detection in social networks has been well-researched in the recent years and a plethora of approaches have been proposed. However, these approaches focus only on single-layer networks with one type of interaction among the users. The analysis of only one type of interaction for anomaly detection does not provide a complete picture of the relationships among the users of the networks.

For instance, in a narcotic criminal network, actors interact in multiple ways to exchange tangible resources such as drugs, money, precursor chemicals, equipment, and premises as well as intangible resources such as information, labor, and skills (Bright et al. 2015). Understanding the multi-layer structure of criminal networks helps the law enforcement agencies for identifying the key actors and their interactions in the networks. Using the traditional SNA techniques, the criminal network is aggregated as a single-layer network by interpreting any mode of interaction between two individuals as a connection between them without specifying the nature of the interaction. However, it is possible that some criminals may be central in one or more individual network layers. Hence, the information regarding the key actors involved in the dealing of specific resources may be discarded in such type of analysis. Therefore, the analysis of the multi-layer social network is particularly crucial for identifying key players and for designing disruption strategies for criminal networks. Consequently, there is an urgency and need for multi-layer analysis of the networks for identifying the anomalies by employing the rich information hidden in the individual network layers. Hence, this work aims at developing approaches for detecting point and collective anomalies in multi-layer social networks.

In social networks, the minority of users follows the uncommon topology of either the neighbors of are fully connected to form clique-like structures, or the neighbors are completely disconnected to form star-like structures. The nodes whose neighborhoods follow star/near-star and clique/near-clique patterns can be linked to suspicious behavior and have been established as anomalies, as only a minority of users follows this behavior (Akoglu et al. 2010; Hassanzadeh et al. 2012; Gupta et al. 2013; Hassanzadeh and Nayak 2013a,b; Kaur and Singh 2017). In a multi-layer social network, if the neighborhood of a user in different layers are close to stars or cliques, the online behavior of the user can signify an anomalous behavior. For example, in the narcotic criminal network, the actors who are in the center of stars or cliques in multiple layers are the key players in the criminal network. In social networks, the uncommon friendship patterns of cliques and stars can signify anomalies such as spammers, fraudsters, or sexual predators (Fire et al. 2012). A star topology in online social networks could also signify a celebrity or influential person. Hence, in this work, the detection of these two types of point anomalies in multi-layer social networks is considered and an unsupervised approach is developed by using graph theoretic features and data mining techniques. Even though the proposed approach is applied on multi-layer social networks, it can be applied to any multi-layer network with intra-layer connections. For example, the proposed approach can identify important hub cities in multi-layer transportation networks, and important proteins and genes that have critical roles in biological networks.

Spamming is the most predominant form of anomalous activity prevalent in online social networks. Spamming involves undesirable users sending malicious tweets consisting of text and HTTP URLs to large number of legitimate users as possible (Zheng et al. 2016). The motivations for spammers to spread spam messages is with an intention 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.

Being one of the fastest growing online social networks, Twitter has become a primary target platform for social spammers. A considerable amount of research work has been carried out in the field of detecting spam messages and social spammers in Twitter. However, one of the crucial issues in Twitter is that the social spammers usually form collective anomalies or spammer communities to spread spam messages to a large set of legitimate users. Consequently, it is highly desirable to identify such spammer communities prevailing in Twitter. Hence, in this work, an unsupervised approach is proposed for detecting spammer communities in Twitter by using graph-theoretic features of the network and the network attributes.

1.7 APPLICATION DOMAINS OF ANOMALY DETECTION IN SOCIAL NETWORKS

This section presents some of the specific real-world application domains where the social network anomaly detection methods have been used.

1. Fraud Detection in Online Social Networks: Online social networks have become new targets for cybercrime, and malicious users attempt to perform illegal activities such as spamming, cyber attacks, bullying, fraudulent information, organized crimes, and even terrorist attack planning on these systems (Yu et al. 2015). Furthermore, online social networks are prone to malwares, spam messages and other offensive materials (Rahman et al. 2012; Gao et al. 2012; Hassanzadeh et al. 2012; Hassanzadeh and Nayak 2013a,b; Akoglu et al. 2010; Shrivastava et al. 2008). These fraudulent activities often cause monitory loss and harm to other users of these online platforms. As the patterns of interactions of the perpetrators of these illegal activities often significantly deviate from normal users, they can be detected by using network-based anomaly detection techniques. The fraudulent activities of users in online social networks necessitate the importance of digital forensics in social networks arena. Anomaly detection can provide inherent and invaluable information for detecting criminal activities in social networks for Social Network Forensics (Keyvanpour et al. 2014).

2. Insider Threat Detection: An insider threat is a security threat to an organization from individuals within the organization. The threat can be fraud, theft of sensitive information, and destruction or compromise of hardware and software resources. Network-based anomaly detection measures can be used for identifying insider threats. The work presented in Eberle et al. (2010) uses Graph-Based Anomaly Detection (GBAD) algorithms (Eberle and Holder 2007) to detect insider threats and cybercrime. Chen and collaborators (Chen et al. 2011; Chen and Malin 2011; Chen et al. 2012c,a,b) proposed several effective social network-based methods for identifying insider threats in Collaborative information systems (CIS) such as electronic health record systems. A CIS manages important and sensitive information in collaborative and dynamic environments. Chen and Malin (2011) and Chen et al. (2012b) proposed Community Anomaly Detection System (CADS) to identify anomalous insiders based on the access logs of collaborative environments. Chen et al. (2011, 2012c) formulated a local network approach called Specialized Network Anomaly Detection (SNAD) that leverages social network analysis technologies to detect anomalous access made by a user in dynamic collaborative systems.

3. Review/Opinion Spam Detection: Nowadays, it has become a common practice for customers to read the opinions or reviews written by other customers before deciding to purchase a product. These reviews are also used as feedbacks by manufacturers to identify the problems of their products. However, in review websites such as *amazon.com*, fraudsters write *fake reviews* or *bogus reviews* to defame or boost the reputation of manufacturers and vendors, and to mislead the customers (Jindal et al. 2010). Detecting these fake reviews has become more critical, as these reviews can result in significant financial loss/gain for the manufacturers and vendors. Review or opinion spams can be detected by mining the networks induced by the reviewers, reviews, products or stores. These network-based methods can be used complementary to the traditional methods for reviewer spam detection such as language features, text similarity, and rating patterns. The review spammer detection methods proposed in Wang et al. (2011, 2012a) rely on a heterogeneous review network with three types of nodes to show the relationships among reviewers, reviews, and stores or products that the reviewers have reviewed. Spams are identified by analyzing the interactions among the nodes in the review network. The method proposed by Akoglu et al. (2013) employs a signed network for representing the negative and positive reviews, and uses relational classification to detect opinion spam.

4. Financial Trade Fraud Detection: In financial trade frauds, traders trade themselves to increase the share values and to manipulate the stock market. These illegal trading activities are known as trading rings, and are identified by mining the networks extracted from the trading transactions at consecutive time steps. Li et al. (2010, 2012) developed algorithms to identify the trade ring patterns called blackhole patterns and volcano patterns in large directed financial trading networks.

5. Auction Fraud Detection: Online auction websites such as Ebay and eBid have been receiving much attention in the recent years. The growing popularity of the online auctions has also led to an increase in auction frauds. The most prevalent auction frauds are in the form of non-delivery fraud, where the fraudulent seller receives the payment from the buyer, but fails to deliver the products. The methods proposed in Chau et al. (2006) and Pandit et al. (2007) detect auction frauds by modeling the transactions among users as networks, with nodes denoting users and edges denoting the interactions among them. These methods classify the users of the auction network into three categories: fraudster, accomplice, and honest. Tsang et al. (2014) identify collaborative auction frauds by first generating an initial anomaly score for each user, and then applying belief propagation.

6. Influence Maximization: Apart from identifying fraudulent, distrustful, or dangerous behavior, anomaly detection is also helpful in identifying influential individuals and rare events in a network. For example, an unusual friendship pattern such as star topology in online social networks could signify a celebrity or an influential individual (Shetty and Adibi 2005; Hassanzadeh et al. 2012; Hassanzadeh and Nayak 2013a,b). This information can be used for viral marketing by an e-commerce agency for promoting their products in the influential individual's friendship network.

1.8 THESIS ORGANIZATION

Rest of this thesis is organized as follows. *Chapter 2* discusses a review on the related research works in the areas of anomaly detection and spammer detection in social networks. Additionally, this chapter discusses about the research challenges and open research areas in these fields. This chapter also talks about the significant research areas in multi-layer networks. *Chapter 3* describes the research problem of the thesis. *Chapter 4* presents the proposed approach for detecting anomalous nodes in multi-layer social networks. This chapter also presents the experimental evaluation of the approach in terms of the effectiveness and running time. *Chapter 5* presents the proposed approach for discovering spammer communities in Twitter multi-layer network. This chapter also provides the experimental evaluation of the approach. *Chapter 6* summarizes the contributions of the research presented in this thesis and some future research directions.

CHAPTER 2

LITERATURE REVIEW

In this chapter, a structured review of the various state-of-the-art techniques and related research works for detecting anomalies and spammers in social networks is presented. The first section presents the techniques and tools for detecting anomalies in social networks. In addition, the promising research directions and research challenges associated with anomaly detection in social networks are discussed in this section. After presenting a brief review on the significant research areas and research challenges in multi-layer networks, a review of the literature on spammer detection in social networks is presented in the subsequent section. This section also discusses about the research challenges and open research areas in spammer detection in social networks.

2.1 ANOMALY DETECTION IN SOCIAL NETWORKS

In the past decade, a multitude of techniques have been developed for social network anomaly detection in a wide variety of problem settings. However, this field is still relatively young and rapidly growing. Hence, there is a growing need for an organized study of the research work done in the area of anomaly detection in social networks. This section provides a comprehensive and systematic study of the research works carried out in the field of anomaly detection in social networks. It also brings out the open challenges and research issues in this field. Even though the emphasis of this section is to review the anomaly detection techniques for social networks proposed in the last ten years, few of the earlier works that are formative to this area have also been reviewed. In addition, we have identified the key aspects associated with the problem of anomaly detection in social networks, and have provided a multi-level taxonomy to categorize the existing anomaly detection techniques based on i) nature of input network, ii) types of anomalies, and iii) anomaly detection approach.

2.1.1 Different Aspects of Anomaly Detection in Social Networks

As the problem of anomaly detection is application-specific, it is determined by different aspects of the data to be analyzed such as the type of input data, the availability of ground truth, the type of anomalies being identified, and the restrictions of the specific application domain. This section describes the different characteristics of anomaly detection in social networks such as the nature of input networks and the type of anomalies that are being identified.

2.1.1.1 Nature of Input Networks

A key characteristic of an anomaly detection method is the nature of the input data (Chandola et al. 2009). In the case of social network anomaly detection, the input networks that are used by various methods can be categorized as static/dynamic or attributed/unattributed, based on the type of analysis performed for anomaly detection.

a) Static versus Dynamic networks: In static networks, the number of nodes and the relationships between the nodes do not change over time. More specifically, a static network is able to represent only a single snapshot of the social network data at any instant of time. However, social networks are constantly evolving, leading to dynamic networks. They are subject to discrete changes such as insertions or deletions of nodes or edges, and alterations to attributes associated with nodes or edges. For example, in social networks individuals may lose old acquaintances, make new friends, or move from one place to the other, which leads to new links appearing and existing links disappearing. Last decade has witnessed a growing interest in dynamic networks and a whole body of algorithmic techniques and data structures have been discovered for them.

Dynamic networks from different application domains evolve differently over time which leads to application-specific anomalies and approaches. For slowly evolving networks such as bibliographic networks, snapshot analysis can be employed, by analyzing the state of the networks at two different instants of time (Aggarwal and Subbian 2014). Such snapshot analysis can be performed off-line effectively. For highly evolving or streaming networks, real-time analysis is required. This is more difficult due to the computational complexity involved and the inability to store the whole network on the disk.

b) Unattributed versus Attributed Networks: In unattributed or unlabeled networks, the attributes or labels associated with the individuals or their interactions such as any details about the type of interaction, the age of the individuals involved in the interaction, or the duration of the interaction are not considered for anomaly detection (Savage et al. 2014). The only information considered about an unattributed network is the network topology or the fact that the interaction has occurred. In other words, anomalies in unattributed social networks are determined by analyzing only the interaction between the individuals, because the details about the individuals and their interactions are either ignored or not available.

Networks that have attributes associated with nodes and/or edges are referred to as attributed or labeled networks. For instance, individuals in a social network can have different attributes such as the locations where the individuals live in or work, their educational qualifications, age, etc., whereas the interactions between the individuals or edges may have types, duration, frequency, etc. For detecting anomalies in attributed networks, these attribute values are also considered, in addition to the network structure. Such meta-data associated with nodes and edges can significantly enhance the anomaly detection by providing auxiliary information for distinguishing between normal and abnormal behavior.

2.1.1.2 Types of Anomalies in Social Networks

Another important characteristic of an anomaly detection technique is the type of anomalies and the manner in which the anomalies are reported. The anomalies are categorized based on the output produced by various anomaly detection methods that have been proposed in the literature. The detection methods uncover anomalous units such as nodes, edges, subgraphs, and/or events in the networks. The methods either assign an anomaly score to each unit, or classify each unit as normal or anomalous.

a) Anomalous Nodes: If we need to identify the individual users whose behavior deviates considerably from the usual behavior of users in the social network, we may view a subset of users or nodes as anomalies. Anomalous nodes are also known as point anomalies, as they are scattered in the network. For example, spammers who send unsolicited messages to other users, and malicious users who cause harm to other users in the network are anomalous nodes or point anomalies.

b) Anomalous Edges: If we need to identify unusual or irregular interactions among the users in the network, we may view a subset of edges as anomalous. In other words, an edge can be anomalous if it is an unlikely edge or its weight fluctuates over time. The edge weight can correspond to the number of messages exchanged between individuals in the network. For example, if a social network has changed in some substantial way, then in most contexts this is likely that there are some individuals who are now either communicating more or less frequently than usual, or communicating with different individuals than usual (Heard et al. 2010).

c) Anomalous Subgraphs: Anomalous subgraph detection aims at finding the subnetworks such that the pattern of interaction among the nodes in the subnetwork is irregular compared to the other nodes in the network. These anomalies are also known as collective or group anomalies. For example, if we need to identify groups of fraudulent people collaborating to promote their reputation in an on-line auction system, communities of spammers that sends unsolicited messages to legitimate users, or groups of people colluding to create fake product reviews, then the anomalies will be collective anomalies (Pandit et al. 2007; Yu et al. 2015).

d) Events: Events occur exclusively in dynamic networks, and are the discrete time steps at which the social network is substantially different from the preceding and the succeeding networks in the dynamic network sequence (Wang et al. 2017). Event detection aims at finding the time steps at which the change or event has occurred, and then identifying the nodes, edges and/or subgraphs that have contributed to the event.

2.1.2 Anomaly Detection in Static Social Networks

Even though social networks evolve over time, it is often useful to analyze them as if they were static. An anomaly in static network occurs with respect to the rest of the network. Hence, the anomaly detection task in static networks is to uncover anomalous units such as nodes, edges, or subgraphs given the entire network topology. In this section, the various state-of-the-art methods for mining anomalies from static social networks are discussed. The methods are categorized according to the nature of input networks they analyze, the type of anomalies they discover, and the approaches they follow. Figure 2.1 shows the organization of the study.

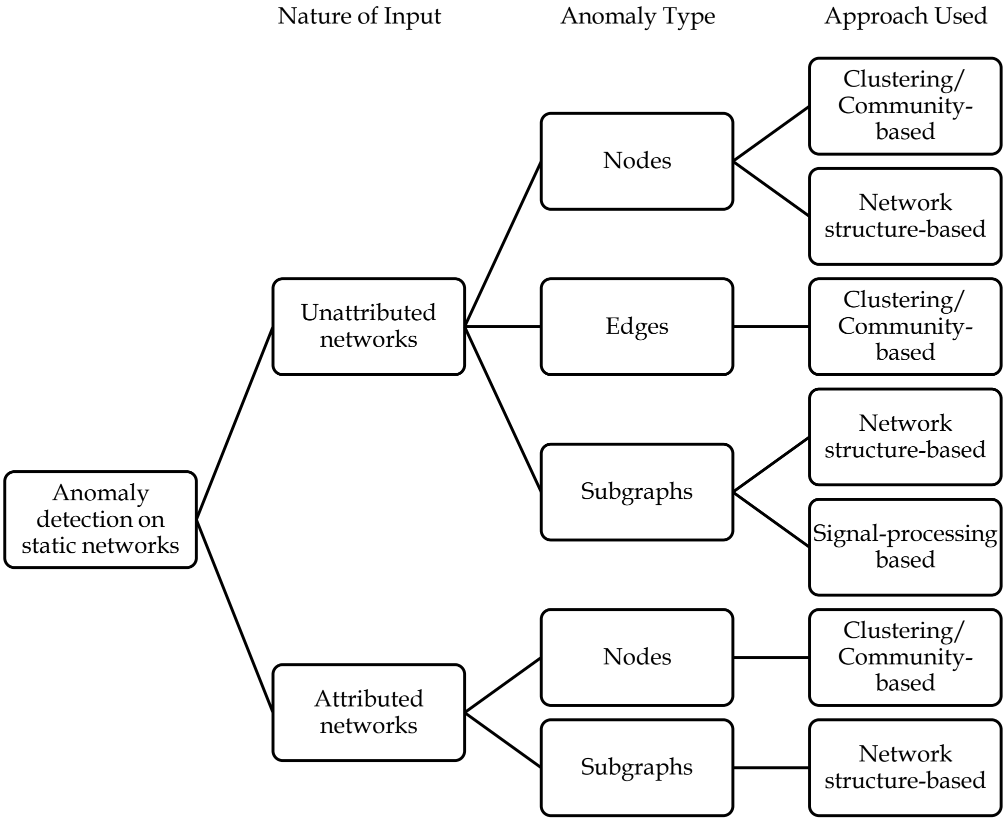


Figure 2.1: Taxonomy of the survey on static social networks. Methods are categorized based on the nature of input network, type of anomalies they identify, and the underlying approach.

2.1.2.1 Static Unattributed Networks

Anomalies in static unattributed social networks are identified by analyzing the interaction between the users or structure of the network at any instant of time, as the details about the individuals as well as their interactions are either ignored or not available. In this section, first the anomaly detection methods are categorized based on the underlying approach, and then a review of the various anomaly detection techniques that are developed for static unattributed networks is presented.

Types of Approaches

The various approaches that have been used for detecting anomalies in static unattributed networks can be categorized into three groups: *clustering / community-based*, *network structure-based* and *signal processing-based approaches.*

a) Clustering/community-based approaches: The clustering/community-based approaches for anomaly detection in static networks find densely connected nodes in the network as clusters or communities, and identify the nodes or edges that interconnect different communities (Sun et al. 2005; Xu et al. 2007; Sun et al. 2010). These approaches define anomaly as nodes or edges that interconnect different communities, but do not belong to any of the communities. Most of the clustering/community-based approaches aim to find anomalies after performing clustering or community detection.

b) Network structure-based approaches: This group of approaches exploit the given network structure or different shapes of topology to compute graph-specific feature space and to detect anomalous nodes (Akoglu et al. 2010; Hassanzadeh et al. 2012; Hassanzadeh and Nayak 2013b,a). These approaches transform the anomaly detection problem on networks into traditional outlier detection problem. Akoglu et al. (2010) specify features of egonet such as degree of the ego, its total weight, number of edges in the egonet, and principal eigenvalue of the weighted adjacency matrix of the egonet. An egonet consists of a central node called *ego* and a set of neighbors to which the ego is directly connected to, plus the edges among them. Henderson et al. (2011) extend the feature base and propose an algorithm called ReFeX (Recursive Feature eXtraction) that recursively aggregates node-based features with neighborhood or egonet-based features and yields regional features by capturing behavioral information. After extracting the feature space, any of the traditional outlier detection methods can be applied to detect anomalous nodes.

c) Signal processing-based approaches: Signal processing on graphs is an emerging field that extends the traditional signal processing for graphs and treat comparatively small anomalous subgraphs as signals embedded in a much bigger network taken as the background noise. In signal processing, if the signal is weaker, or if it resembles the background more, then the signal is more difficult to process. In other words, when the signal-to-noise ratio is low, signals are harder to detect, estimate, and classify. In this setting, anomalous subgraph detection is a special class of the more general graph partitioning problem in which the spectral properties of a graphs Laplacian matrix is used to determine the presence of an anomaly (Miller et al. 2010, 2011, 2012; Bliss 2015; Miller et al. 2015).

Anomalous Node Detection

The state-of-the-art anomalous node detection methods on static unattributed networks are based on *clustering/community-based*, *network structure-based*, and *signal processing - based* approaches. This section reviews the anomaly detection methods that have been developed for spotting anomalous nodes in static unattributed networks.

Clustering/Community-based approach: One of the earliest works on clustering/ community - based anomaly detection by Sun et al. (2005) addresses the problems of neighborhood formation and spotting anomalous nodes in bipartite networks. The real-world scenarios that can be represented with bipartite networks include: users versus files upload/download relationships in point-to-point networks, stocks versus traders network in financial stock markets, and authors versus conferences relationships in publication networks of different research areas. The anomalies in these settings can be: cross-border files in point-to-point networks, cross-sector traders in financial stock networks, and inter-disciplinary authors in publication networks. The authors proposed an algorithm to find the community or neighborhood of each node in the bipartite graph using random walks with restart and graph partitioning, and used this algorithm to detect anomalous nodes in the network.

SCAN (Structural Clustering Algorithm for Networks) (Xu et al. 2007) and GskeletonClu (Graph-skeleton based clustering) (Sun et al. 2010) are density-based network clustering algorithms to identify clusters, hubs, and outliers in large networks. These algorithms use the neighborhood of the nodes as the clustering measure rather than their direct neighbors. Nodes sharing common neighbors are clustered together. Nodes that interconnect several clusters are identified as hubs, and nodes that do not belong to any cluster are designated as outliers. Both SCAN and GskeletonClu depend on a sensitive parameter called minimum similarity threshold for clustering. Whereas SCAN provides no automated way to find the minimum similarity threshold, GskeletonClu automates the issue of selecting the parameter.

Network-structure based approach: Akoglu et al. (2010) proposed a network structure-based technique called OddBall to discover anomalies such as clique/near-clique, star/ near-star, heavy vicinity, and dominant edge patterns from large, weighted networks. Oddball uses the features of egonet such as degree of the ego, its total weight, number of edges in the egonet, and principal eigenvalue of the weighted adjacency matrix of the egonet to detect anomalies. The egonet features are then examined in pairs and different power-law rules are defined among the features. The majority of egonets follow these power-law rules. The set of nodes that do not conform to these rules are declared as outliers.

On similar lines, Hassanzadeh et al. (2012) proposed a framework based on egonet's features for identifying anomalous nodes in social networks. The framework aims to find the common behavior obeyed by majority of the nodes by computing graph theoretic properties of a node's egonet such as number of nodes, number of edges, the average betweenness centrality, and community cohesiveness of the node's super-egonet. It then models the relationships between these metrics by using distribution models such as linear and power laws. The anomaly score for each node is then calculated by determining the distance between the linear and power law fitting line for each node's egonet. Based on the anomaly score, a labeled subset of nodes is obtained. In order to minimize the number of false positives and false negatives, a threshold is computed for the labeled subset of nodes using the scoring function.

The anomalous node detection methods presented in Hassanzadeh and Nayak (2013b,a) consider fuzziness of user behavior in an online social network. The authors propose hybrid methods that combine graph theoretic properties, clustering, and Fuzzy logic for spotting anomalous individuals. In Hassanzadeh and Nayak (2013b), as a first step, an initial anomaly score is computed based on the extend of similarity with an egonet's structure to clique or star, and named them as cliqueness and starness scores respectively. A clustering algorithm based on Expected Maximization (EM) is then used to categorize the users according to their initial anomaly scores. Finally, the method makes use of Fuzzy logic using membership functions to define the degree of anomalousness. In Hassanzadeh and Nayak (2013a), the initial cliqueness and starness scores are computed based on the egonet features such as number of nodes and number of edges in the egonet. It then employs Expected Maximization-Gaussian Mixture Model algorithm, Fuzzy c-means clustering algorithm, and a combination of Gaussian Mixture Model and fuzzy logic to differentiate between normal and anomalous individuals.

Kaur and Singh (2017) propose a network structure-based approach for detecting near-star/near-clique anomalies by extracting features of the network such as egonet node count, egonet edge count, and brokerage, and computing the anomaly score as the deviation from linear and power laws.

Anomalous Edge Detection

The existing anomalous edge detection methods on static unattributed networks are based on *clustering/community-based* approach. This section provides an overview of the different approaches developed for detecting anomalous edges from a static unattributed network.

Clustering/community-based approach: Chakrabarti developed AUTOPART, a parameter - free, scalable, and iterative approach, to detect anomalous edges and node groups. AUTOPART automatically partitions the network into clusters by re-organizing the rows and columns of the adjacency matrix. The edges that do not belong to any cluster as well as the edges that inter-connect different clusters represent anomalies. It also differentiates the nodes that have many cross-cluster connections as anomalies. For clustering the network, the algorithm employs Minimum Description Length (MDL) principle (Rissanen 1999) for re-organizing the adjacency matrix of the graph into homogeneous blocks. These blocks contain nodes that are more thickly connected together compared to the remaining nodes in the network.

Another community-based approach for detecting anomalous edges and nodes is presented in Tong and Lin (2011). The authors propose a non-negative residual matrix factorization framework (NrMF) to identify anomalous nodes and edges. NrMF is based on low-rank approximations on the adjacency matrices of the network. The low-rank approximation of adjacency matrix A is normally formulated in a factorized form, $A = FG + R$, where F and G are factors and R is the residual matrix. The low-rank factors F and G show the community structure in the network, whereas the residual matrix R often indicates anomalies in networks. Unlike the traditional non-negative matrix factorization (NMF) that restricts the factors F and G to be non-negative, NrMF enforces non-negativity constraint on the residual matrix R for finding anomalies.

Anomalous Subgraph Detection

The algorithms developed for detecting anomalous subgraphs from a static unattributed network are based on *network structure-based* and *signal processing-based* approaches. In this section, the different techniques proposed for anomalous subgraph detection in static unattributed networks are reviewed.

Network structure-based approach: In email spam and viral marketing in social networks, the fraudulent user creates a set of fake identities and uses these identities to interact with a large random set of innocent users. Shrivastava et al. (2008) defined Random Link Attacks (RLA) to model such collaborative malicious activities and proposed algorithms to mine subgraphs satisfying the RLA property. RLA is detected in two steps: the first step is identifying the suspect nodes that are possibly part of the attacker cluster by conducting two tests on each individual nodes in the network: clustering test and neighborhood independence test. As the innocent users are chosen at random, they are improbable to be interconnected, forming a star-like pattern in the network. In order to detect the suspect nodes in the network, the triangles in each egonet are counted, with a lower triangle count indicating an attacker. In the next step, the attack set is identified by growing the neighborhood of the suspect nodes.

Signal processing-based approach: The signal processing-based framework proposed in Miller et al. (2010) uses L1 properties of the eigenvectors of the network's modularity matrix to determine the presence of an anomalous subgraph. The framework aims at determining whether an observed network was generated by a given random process or if there is other behavior that deviates from the mode. The objective of the anomalous subgraph detection problem is to solve the binary hypothesis test:
H0 : The network is the large background graph without anomalous subgraphs (noise)
H1 : The network is the large background graph with anomalies (signal+noise)
The framework employs the R-MAT Kronecker graph (Chakrabarti et al. 2004) as the background network in the null hypothesis. This graph model exhibits a heavy tailed power law degree distribution and inherent clustering of nodes often found in real-world networks. In Miller et al. (2011), the detection framework and the related algorithms proposed in Miller et al. (2010) are applied for the specific problem of detecting threat subgraphs embedded in social networks. On similar lines, Miller et al. (2015) proposed a framework based on the principal eigenspace of the network's residuals matrix in which an observed random network is compared to its expected value to find anomalous subgraphs.

2.1.2.2 Static Attributed Networks

Anomalies in static attributed networks are determined by analyzing the network topology as well as the attributes associated with nodes and/or edges. As mentioned in Section 2.1.1.1, the additional information available on nodes and edges of the network can significantly improve anomaly detection.

The anomaly detection methods developed for static attributed networks mainly identify anomalous nodes and subgraphs. In this section, first we categorize the methods based on the underlying approach they follow, and then provide an overview of the methods that have been proposed for detecting anomalies on static attributed networks.

Types of Approaches

The anomaly detection approaches that have been reported for static attributed networks can be categorized into two groups: *clustering/community-based* and *network structure-based* approaches. The *clustering/community-based* methods detect *community outliers* whose properties differ from other members in the community, whereas *network structure-based* methods detect subgraphs that are rare with respect to the structure of the attributed network.

a) Clustering/community-based approaches: This type of anomaly detection methods identify the subset of nodes within the context of communities such that their characteristics differ significantly from other members of the same community. These nodes are termed as community outliers. For instance, a low-income person having many rich people as friends is an example of community outlier (Gao et al. 2010b). The community-based anomaly detection methods proposed in Gao et al. (2010b) and Muller et al. (2013) integrate attribute graph clustering and outlier detection in a single algorithm.

b) Network structure-based approaches: This class of anomaly detection methods aim to spot subgraphs or substructures in the network that are unusual with respect to the structure as well as labels or attributes of the network. Most of these methods make use of the SUBDUE (Holder et al. 1994) system, which is based on MDL principle, for discovering frequent substructures within networks.

Anomalous Node Detection

The detection methods that have been reported for anomalous nodes are based on *clustering/community-based approach.* This section reviews the different methods for identifying anomalous nodes.

Gao et al. (2010b) introduced the concept of community outliers, and developed a unified framework for detecting outliers and for discovering communities. They proposed community outlier detection algorithm (CODA) that combines both community detection and outlier detection in a probabilistic formulation based on Hidden Markov Random Fields (HMRF), by combining information from both network structure and node attributes. The node information is formulated as a mixture of Gaussian distributions or multinomial distributions, whereas the network topology information is encoded as spatial constraints on the hidden variables via the HMRF model. Even though the computational cost of the community outlier detection algorithm is linear in the number of nodes, the success of the algorithm is highly dependent on the good initialization of the clusters.

GoutRank (Muller et al. 2013) is an approach for ranking network nodes according to their degree of deviation in both node attributes and network structure. The approach focuses on the detection of complex outliers that deviate with respect to a subgraph of highly connected nodes. Whereas the individual outlier is highly similar to other nodes in the subgraph, it deviates significantly with respect to a subset of significant attributes called subspaces. GoutRank addresses two problems: 1) the problem of selecting subgraphs with their individual subspaces and 2) the problem of scoring of nodes in these multiple subspaces. It uses graph clustering techniques and subspace analysis as preprocessing steps for generating the outlier rankings.

Yang et al. (2015) developed a framework based on bipartite graph and co-clustering for detecting anomalous users and messages in microblogging. A bipartite graph is constructed to represent homogeneous and heterogeneous interactions between users and messages. Then, a co-clustering algorithm based on non-negative matrix tri-factorization (NMTF) is used to discover anomalous users and messages at once, by considering not only the user attributes and messages but also the homogeneous and heterogeneous interactions. The framework detects both individual and group anomalous users.

Anomalous Subgraph Detection

The anomaly detection methods that aim to identify anomalous subgraphs are based on *network structure-based* approach. This section provides an overview of the methods proposed for anomalous subgraph detection.

Noble and Cook (2003) introduced two methods for identifying unusual pattens in a network with categorical labels using the SUBDUE system. The first method identifies specific, unusual substructures within a network. In the second method, anomalous subgraphs are detected by partitioning the network into distinct, separate subgraphs and then comparing each of them against the other subgraphs for unusual occurrences.

The main idea behind the two methods is that subgraphs containing frequent substructures are generally less anomalous than subgraphs with few frequent substructures. The SUBDUE system discovers frequent substructures and then compresses the network with them. The anomalous subgraphs experience less compression compared to other subgraphs, as they contain only few frequent substructures. The authors also introduced a measure of graph similarity called conditional subgraph entropy, which specifies the number of bits needed to describe a substructure's surroundings. If the conditional subgraph entropy of a substructure is higher, then it is infrequent and hence it is anomalous.

Another method, also based on SUBDUE, that aims to uncover anomalies in attributed networks is by Eberle and Holder (2007). They introduced a suit of three algorithms called GBAD (Graph-Based Anomaly Detection) algorithms for discovering anomalies in three types of possible changes in the network that nearly match non-anomalous activities: node/edge insertions, node/edge label modifications, and node/edge deletions. Each of the three algorithms aims at one of these anomalies and employs the MDL principle to identify those substructures that constitute anomalous nodes and edges. GBAD-P (Probability) algorithm detects anomalous node/edge insertions, GBAD-MDL algorithm detects anomalous node/edge label modifications, and GBAD-MPS (Maximum Partial Substructure) algorithm detects anomalous node/edge deletions. The main idea behind the algorithms is to find anomalies where the anomalous substructure in the network is part of a non-anomalous substructure called *normative pattern*.

The GBAD algorithms (Eberle and Holder 2007) operate on unweighted networks with discrete node and edge labels, and can not incorporate continuous labels. To overcome this problem, (Davis et al. 2011) presented YAGADA (Yet Another Graph-based Anomaly Detection Algorithm), an algorithm to search attributed networks for anomalies using both structural information and numeric labels. If the numeric values in the network are normal, the algorithm discretizes the values with the same constant categorical label. If the values are abnormal, they are assigned an anomalous score. When the network is subsequently searched for frequent substructures, the nodes with the same constant value are incorporated into frequent patterns. The other values are infrequent and therefore substructures that contain them are anomalous. Like the algorithms proposed by Noble and Cook (2003) and Eberle and Holder (2007), YAGADA also employs SUBDUE to find frequent substructures.

In a recent work, Gupta et al. (2014b) developed a query-based outlier subgraph detection mechanism called Subgraph Outlier Detection Algorithm (SODA). Different from the methods discussed so far, it accepts a subgraph as an input query and returns top matching anomalous subgraphs from the original attributed network sorted by their outlier score. An anomalous subgraph has many low-probability or unexpected edges and lacking many high-probability or expected edges within itself, and between itself and its neighborhood. The authors model the anomalous subgraph detection as a linear optimization problem and use one-hop neighborhood information of the query subgraph to deduce the feature weights. Apart from learning the feature weights, linear optimization is also used to calculate the outlier score of the subgraphs.

This section presented a review of the various anomaly detection methods that are proposed for static social networks. As the different methods are developed for different application domains and for detecting different types of anomalies, a quantitative comparison among the methods is not practical. However, a qualitative comparison and summary of the different anomaly detection techniques that are developed for static networks are shown in Table 2.1.

2.1.3 Anomaly Detection in Dynamic Social Networks

As dynamic social networks are constantly undergoing alterations to their structure and/or attributes, the major tasks in identifying anomalies are to detect change-points or events in time at which the majority of the nodes or edges deviate from their normal behavior, and to identify the particular parts of the network that are responsible for the change-point. When considering the dynamic nature of networks, new types of community-based anomalies such as formation of new communities, and splitting up and disappearance of existing communities are also possible (Chen et al. 2012d). This section provides a review of the existing methods for identifying anomalies in dynamic social networks. The taxonomy of the survey is shown in Figure 2.2.

2.1.3.1 Event Detection

Event detection aims at finding the time points at which the change or event has happened, and then identifying the nodes, edges, and/or subgraphs that are responsible for the event. An event is detected by comparing the similarity of the network snapshots at consecutive time points.

Table 2.1: Summary and comparison of articles on anomaly detection in static social networks

	Network		Anomaly				
Research Article	Unattributed	Attributed	Node	Edge	Subgraph	**Approach Used**	**Reporting of anomalies**
Sun et al. (2005)	✓		✓			Clustering/Community-based	Normality scores
Xu et al. (2007)	✓		✓			Clustering/Community-based	Binary labels
Sun et al. (2010)	✓		✓			Clustering/Community-based	Binary labels
Akoglu et al. (2010)	✓		✓			Network structure-based	Anomaly scores
Hassanzadeh et al. (2012)	✓		✓			Network structure-based	Anomaly scores
Hassanzadeh and Nayak (2013b)	✓		✓			Network structure-based	Binary labels
Hassanzadeh and Nayak (2013a)	✓		✓			Network structure-based	Binary labels
Kaur and Singh (2017)	✓		✓			Network-structure based	Anomaly scores
Chakrabarti	✓		✓	✓		Clustering/Community-based	Binary labels
Tong and Lin (2011)	✓		✓	✓		Clustering/Community-based	Binary labels
Shrivastava et al. (2008)	✓				✓	Network structure-based	Node subsets

Continued on next page

Table 2.1 – Continued from previous page

	Network		Anomaly				
Research Article	**Unattributed**	**Attributed**	**Node**	**Edge**	**Subgraph**	**Approach Used**	**Reporting of anomalies**
Miller et al. (2010)	✓				✓	Signal processing-based	Node subsets
Miller et al. (2011)	✓				✓	Signal processing-based	Node subsets
Miller et al. (2015)	✓				✓	Signal processing-based	Node subsets
Gao et al. (2010b)		✓	✓			Clustering/Community-based	Binary labels
Muller et al. (2013)		✓	✓			Clustering/Community-based	Anomaly scores
Yang et al. (2015)		✓	✓			Clustering/Community-based	Node subsets and messages
Noble and Cook (2003)		✓			✓	Network-structure based	Anomaly scores
Eberle and Holder (2007)		✓			✓	Network-structure based	Anomaly scores
Davis et al. (2011)		✓			✓	Network-structure based	Anomaly scores
Gupta et al. (2014b)		✓			✓	Network-structure based	Anomaly scores
Shah et al. (2016)		✓	✓			Network-structure based	Anomaly scores

The similarity of the consecutive networks can be measured by comparing the summaries extracted from the networks. When the similarity is less than a threshold, an event is detected and the corresponding network snapshot is flagged as anomalous. The methods proposed by Sun et al. (2007), Akoglu and Faloutsos (2010), Sun et al. (2006), Kolda and Sun (2008), Aggarwal et al. (2011), Koutra et al. (2012), Huang and Zeng (2006), Heard et al. (2010), Papalexakis et al. (2012), Miller et al. (2012), and Miller et al. (2013) detect events from dynamic networks. As these methods detect anomalous units such as nodes, edges, and/or subgraphs as well, they are discussed in detail in the subsequent subsections.

2.1.3.2 Dynamic Unattributed Networks

For detecting anomalies from dynamic unattributed networks, only the alterations in structure of the network in subsequent time steps are considered; the attributes associated with nodes or edges are not considered. In this section, first we categorize the methods based on the approach they follow, and then we describe the various methods developed for dynamic unattributed networks.

Types of Approaches

The various approaches proposed for anomaly detection in dynamic unattributed networks can be grouped into three categories: *matrix/tensor decomposition-based, community - based*, and *probability-based* approaches

a) Matrix/tensor decomposition-based approaches: The decomposition-based methods identify anomalies by decomposing the adjacency matrix or tensor representation of the dynamic networks and by analyzing the eigenvectors, eigenvalues, or singular values appropriately (Sun et al. 2007; Akoglu and Faloutsos 2010; Miller et al. 2012; Yu et al. 2013; Kolda and Sun 2008). Sun et al. (2006) introduced tensor streams to solve the streaming problems in dynamic networks. Tensors are generalizations of matrices. A dynamic network is represented as a third order tensor in which the first two dimensions denote an adjacency matrix and the third dimension denotes the dynamics of the network. The primary differences among the decomposition-based methods are whether they employ a matrix or a tensor for representing the network, what information is stored in the tensor, and how the matrix or tensor is decomposed.

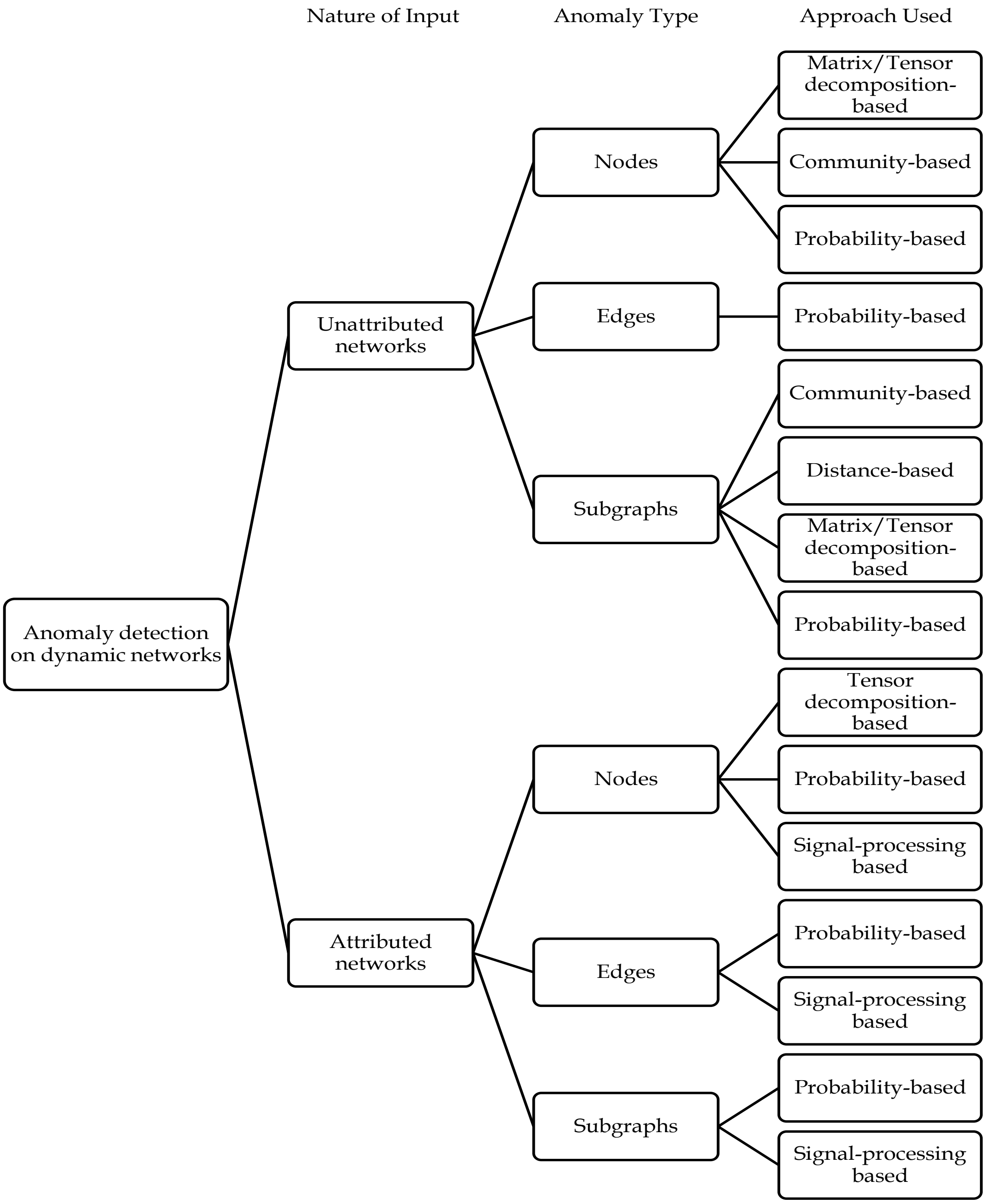


Figure 2.2: Taxonomy of the survey on dynamic social networks. Methods are categorized based on the nature of input network, type of anomalies they identify, and the underlying approach.

b) Community-based approaches: Even though the communities in a network contract, expand, merge, split, appear, vanish, or re-appear after a time period, majority of the nodes within a community follow similar evolution trends which define the evolution trends of the community (Gupta et al. 2012b,a). Community-based anomaly detection approaches spot the nodes that do not follow the common evolution trend of the communities.

c) Probability-based approaches: Probability-based approaches generally build models of normal behavior of the network based on probability theory, probabilistic distributions, and scan statistics (Heard et al. 2010; Priebe et al. 2005; Pandit et al. 2007). Several previous graph snapshots are used for building a model for normal behavior, and every new incoming network instance is compared with this model. Any deviations from this model are reported as anomalies.

Anomalous Node Detection

This section reviews the various anomaly detection methods that have been developed for identifying anomalous nodes in the dynamic unattributed networks. The anomaly detection methods that identify anomalous nodes are based on *matrix/tensor decomposition - based* and *community-based* approaches.

Matrix/Tensor decomposition-based approach: In this section, at first an overview of anomalous node detection methods based on matrix decomposition are presented. Then an overview of the methods based on tensor decomposition are presented.

Sun et al. (2007) employ matrix decomposition for anomaly detection and propose Compact Matrix Decomposition (CMD) to calculate sparse low rank matrix approximations. The low rank decompositions, such as Singular Value Decomposition (SVD) and CUR, reveal hidden variables and associated patterns from high dimensional data. However, they do not consider the sparsity property of the network, and therefore suffer from high computational cost and memory requirements. CMD is computationally efficient and requires less space compared to SVD and CUR. In order to detect anomalies using CMD, the low-rank approximations of input networks are used to summerize the dynamic networks. The reconstruction error of each network is checked over time, and any significant deviation is considered as anomalous and the responsible anomalous nodes are reported.

Anomalous nodes can be detected by comparing their behavior with their previous normal behavior. Akoglu and Faloutsos (2010) proposed an algorithm based on eigenbehavior analysis to detect events and the corresponding anomalous nodes in a dynamic network. The algorithm uses the network structure for extracting various features of each node's egonet such as in-degree, out-degree, out-weight, number of neighbors, reciprocal neighbors, etc. The behavior of each node from each network snapshot is summarized by using these set of features. For each time window, a correlation matrix of node features is created using Pearsons correlation coefficient and the principal eigenvector is computed. The current behavior of each node is obtained by putting the values of the eigenvectors in a vector. The past behavior of each node is computed by SVD decomposition and is compared with the current eigen-behavior vector. If the current behavior is found to be significantly different from recent past, current time window is flagged as anomalous and reported as an event has occurred. The anomalous nodes are identified by examining the singular vectors.

The signal processing-based approach proposed by Miller et al. (2012) also uses matrix decomposition to find anomalous nodes in dynamic unattributed networks. A residual matrix is constructed by taking the difference between the adjacency matrices of observed network and the expected network. A probabilistic model called ChungLu random graph model (Chung et al. 2003) is used for building the expected network. The major analysis algorithm used in the approach is the partial eigen decomposition of the residual matrix and calculation of its eigenvectors and eigenvalues. In order to detect anomalous time windows, a linear ramp filter is applied on the residual matrices, and then partial eigen decomposition is performed. The change in the top eigen vectors is analyzed, and the anomalous nodes are identified by examining the eigen values.

Another method using matrix decomposition is presented in Yu et al. (2013) that identifies important localized regions of change in a fast network stream by applying Principal Component Analysis (PCA) locally. An edge correlation matrix, containing a row and column for every neighbor of the node, is maintained for each node of the network. The authors designed a localized PCA algorithm that can continuously maintain information about the changes in different neighborhoods of the network, and used a fast incremental eigenvector update algorithm to efficiently maintain local correlation information. The eigenvector update algorithm is used to compute anomaly score for each node at each time point. The nodes whose anomaly scores are higher than a threshold are reported as anomalous at that point of time. The non-negative matrix factorization-based approach proposed by (Baingana and Giannakis 2016) jointly tracks temporal communities and anomalous nodes in dynamic unattributed social networks.

In the last few years, tensors and tensor decompositions have gained increasing popularity in the data mining community (Kolda and Bader 2009; Sael et al. 2015). The anomaly detection methods proposed in Sun et al. (2006) and Kolda and Sun (2008) use tensors instead of matrices for representing dynamic networks. Like matrix decomposition, tensor decomposition also approximates the original data in a low dimensionality. Sun et al. (2006) proposed two new techniques called dynamic tensor analysis (DTA) and streaming tensor analysis (STA) to incrementally analyze and summarize large tensors. These techniques are scalable, space efficient, and fully automatic. In order to detect anomalies, a multi-level screening process was followed, where the anomaly is found from the broadest level and gradually narrowed down. More specifically, the anomaly detection detection process consists of three steps: 1) given a sequence of tensors, find the abnormal ones, 2) locate the abnormal modes on those suspicious tensors and, then 3) identify the abnormal dimensions of the given mode. The major idea of the algorithms is the decomposition of the tensors into projection matrices, and the modification of the matrices at each time step. If the modification causes high reconstruction error, then the tensor of that time step is flagged anomalous; and anomalous nodes, subgraphs and events are revealed.

The method presented in Kolda and Sun (2008) use Tucker decomposition (Tucker 1966) to decompose tensors. The problem with Tucker decomposition for sparse tensors is that the input and output tensors cause memory overflow during the decomposition process. This problem is termed as intermediate blowup problem and the authors propose a novel technique called memory-efficient tucker (MET) decomposition to address this issue. MET decomposition optimally utilizes the available memory and is computationally efficient. The MET decomposition approximates a higher order tensor using a smaller core tensor and a matrix for every mode of the original tensor. The reconstruction error is calculated for the smaller sub-tensors. If the error is high for a specific sub-tensor, it shows a deviation form the norm and the sub-tensor is flagged as anomalous.

Tensor analysis is a powerful and promising tool for detecting anomalies from dynamic and multi-aspect network (e.g., DBLP networks with information about the authors, papers and conference). Koutra et al. (2012) developed a PARAFAC tensor decomposition-based method called TENSORSPLAT for identifying micro-clusters and anomalies from such networks. In a recent work, Jeon et al. (2015) put forth HATEN2, a scalable and distributed suite of tensor decomposition algorithms running on HADOOP, the open source version of the MAPREDUCE platform. HATEN2 unifies Tucker and PARAFAC decompositions into a general framework such that the intermediate storage space and computational cost are minimized.

Community-based approach: Gupta et al. (2012b) introduced the concept of *evolutionary community outliers* (ECOutliers) which are outliers with respect to evolving communities. These outliers deviate from the common evolutionary trend of the remaining nodes in a community, and can be found by matching the communities across multiple snapshots. The authors proposed an optimization framework that integrates community matching and outlier detection. The framework employs a coordinate descent algorithm to improve community matching and outlier detection performance iteratively. After matching the communities from successive snapshots, it generates outlierness score for every node in the community. These outlierness scores are used to report the ECOutliers. On similar lines, Gupta et al. (2012a) introduced the notion of *Community Trend Outliers* (CTOutliers), nodes that evolve in a dramatically different manner compared with the rest of the community members, and propose a two-stage approach to detect them. In the first stage, the normal evolutionary trend of the communities is modeled using soft patterns discovered from the dataset (pattern extraction). The second stage analyzes the soft patterns and detects nodes that deviate significantly from the normal trend as CTOutliers (outlier detection).

Another method that detects nodes that do not follow normal community evolutionary trend is proposed by Ji et al. (2013). The authors formulated an incremental algorithm to discover *local evolutionary outliers*, LEOutliers, which are nodes with unusual evolutionary behavior only with respect to their local neighborhood, in a weighted dynamic network. The local neighborhood of a node is described by its local neighborhood subgraph called *Corenet* that includes itself and all the nodes within the two-step neighborhood of the node that have a weighted path greater than a threshold. The algorithm that detects LEOutliers consists of two stages: 1) discovering *Corenets* based on the network topology and edge weights, and 2) measuring outlier score by inspecting and comparing *Corenets* at different snapshots. Nodes having top outlier scores are declared as outliers.

Rossi et al. (2013) presented a fully automatic, community-based method called Dynamic Behavioral Mixed-Membership (DBMM) model to identify node roles, such as star-centers or bridge nodes, and to detect anomalous nodes. DBMM model uses non-negative matrix factorization approach as well as MDL principle to calculate node role memberships. It generates the role transition model for each node based on the evolutionary behavior, and uses this transition model for predicting the network memberships at the next time step. The anomaly score of each node is computed by comparing the predicted network role memberships and actual role memberships.

Another community-based method to detect anomalous nodes is by Araujo et al. (2014). The method spots nodes that constitute *comet* communities which appear and disappear periodically. The dynamic network is modeled as a three-mode tensor, and PARAFAC decomposition is performed to obtain candidate communities. The method then applies MDL principle to identify important communities and to determine the correct community size. Based on the important communities detected, the authors use the principle of tensor deflation to find novel *comet* communities in subsequent iterations.

Probability-based approach: One of the earliest works on network-based anomaly detection by Priebe et al. (2005) uses scan statistics, commonly known as moving window analysis, to detect the anomalous nodes. The basic idea of scan statistics is to scan a small window over data, computing some local statistics for each window. The highest value of the local statistics is referred to as the scan statistic. The method proposed in Priebe et al. (2005) applies scan statistics on disjoint one-week windows of the ENRON who-emails-whom network to find network instances that have remarkably high interactions compared to the past. The number of edges in the neighborhood of each node is taken as the local statistics. The network instance is flagged as anomalous, if the scan statistics is above a threshold value, and the nodes and edges that contribute most to the change are deemed as anomalies.

Another method that aims to find anomalous nodes based on probabilistic models is by Pandit et al. (2007), who proposed an approach called NetProbe to detect fraudsters in online auction networks. NetProbe represents auction users and their transactions as a Markov Random Field to identify the subgraphs of fraudsters in the network, and uses Belief Propagation to predict which users are likely to commit frauds in the future. It classifies the users of the auction network as fraudster, accomplice, or honest. The interaction between fraudsters and accomplices form bipartite cores, and fraudsters can be uncovered by discovering those cores.

The anomaly detection method developed by Heard et al. (2010) exploits a probability - based approach to find anomalous nodes, edges, and subgraphs in the dynamic network. The method uses a two-step Bayesian approach. In the first step, the interaction between node pairs is tested for anomalousness by calculating a predictive p-value according to the Bayesian learning of the count distributions. If the derived predictive p-value falls below a fixed threshold, it represents a deviation from previously modeled behavior. The node pairs are then said to be anomalous and are added to the set of anomalous nodes for this period. The algorithm considers the network snapshots as a stream. It discovers changes in the incoming networks according to the history (sequential analysis), and modifies the history according to the new network snapshot (retrospective analysis). In the second stage, a subnetwork is constructed around this set of nodes, normally extended to include other nodes that have recently interacted with a node in this set, and then standard clustering techniques are applied to investigate the structure in this small subnetwork.

Anomalous Edge Detection

Anomalous edge detection methods aim to find patterns of interaction among individuals that deviate from the usual pattern of interaction in the network. The anomalous edge detection methods on dynamic unattributed networks are based on *probability-based* approach.

Probability-based approach: Anomalous edges in a dynamic network can be discovered by applying link prediction techniques. Future interactions can be predicted through link prediction and the interactions that are very unlikely to occur are deemed as anomalous. Huang and Zeng (2006) developed an anomalous email detection framework by applying a link prediction based formulation. The probability that an interaction occurs between two individuals is estimated using expectation maximization, and is used for assigning likelihood scores to each interaction afterwards. The emails that have low likelihood scores are flagged as anomalous.

Another probabilistic approach that identifies events and anomalous edges in streaming networks is Goutlier that is proposed by Aggarwal et al. (2011). It uses a reservoir sampling approach to capture a structural summary of the network. The sampling method employs node partitioning to yield a summary of the network. When a new network arrives, the likelihood probability of every edge in the network is computed according to the edge generation models of the various node partitions. After calculating the edge likelihood probabilities, the incoming network's global network likelihood is computed by taking the average of all the edge likelihood probabilities. The network is flagged as an outlier, if its likelihood probability is t standard deviations below average of all previous networks. The scan statistics approach proposed by Priebe et al. (2005), discussed in Section 2.1.3.2, can detect anomalous edges as well, apart from detecting anomalous nodes on a dynamic unattributed network.

Anomalous Subgraph Detection

The methods that detect anomalous subgraphs in the dynamic unattributed graphs are based on *community-based, distance-based, matrix/tensor decomposition-based*, and *probability-based* approaches. This section discusses the different anomalous subgraph detection methods that have been developed for dynamic unattributed networks.

Community-based approach: When considering the dynamic nature of networks, new types of community-based anomalies such as formation of new communities, and splitting up and disappearance of existing communities are also possible. The notion of community-based anomalies are introduced by Chen et al. (2012d). They proposed a representative-based technique to identify six types of community-based anomalies that are possible in dynamic networks: growing, shrinking, merging, splitting, appearing, and disappearing communities. The technique is based on graph representatives and community representatives. The graph representative of a network snapshot is the set of nodes that appear in the previous as well as the succeeding snapshots; the community representative of a community is the node that appear in less number of other communities in a network snapshot. The communities are modeled as maximal cliques and therefore enumerating communities is an NP-hard problem. The network snapshots in consecutive time steps are compared using graph and community representatives, thus reducing the computational cost involved in enumerating various communities.

Distance-based approach: Mongiovi et al. (2013a) developed an iterative formulation called NetSpot for spotting and summarizing anomalous subgraphs called Significant Anomalous Regions (SAR) in weighted dynamic networks. SAR is a generalization of an NP-hard problem called Heaviest Dynamic Subgraph (HDS) problem. The anomalousness of each edge in each graph instance is calculated as its statistical p-value, based on the distribution of weights on the edge. If the p-value is lower, the edge is anomalous. NetSpot alternates between detecting the subgraph that maximizes the anomaly score for a given time window, and detecting the time window that maximizes the score for a given subgraph. The algorithm outputs the regions of the network whose anomaly scores are higher than a threshold value, along with their respective time windows.

This work is extended in Mongiovi et al. (2013b) to allow for the subgraphs to change gradually in consecutive time steps. The network may shrink or expand between contiguous snapshots. *Hamming distance* between the set of edges is used as a distance measure to determine the amount of variation between two adjacent subgraphs. The authors presented a filtering based framework called LEGATO for mining smoothly evolving subgraphs in weighted dynamic networks.

Matrix/Tensor decomposition-based approach: The matrix/tensor decomposition - based methods (Tong et al. 2008; Sun et al. 2006, 2007; Kolda and Sun 2008; Koutra et al. 2012) for anomalous node detection discussed in Section 2.1.3.2 can detect anomalous subgraphs as well.

Probability-based approach: Thompson and Eliassi-Rad (2009) put forth a probability-based algorithm to find anomalous subgraphs from a time-evolving network. The recent behavior pattern of the network is modeled as a cumulative network that summarizes all past edges but gives more weight to recent edges by using an exponential decay model. The normal behavior of the subgraphs is modeled by identifying *persistent patterns* among nodes. A persistent pattern is a set of nodes that communicate frequently and form a connected component. Persistent patterns are identified from a given cumulative network by extracting subgraphs whose edge weights are beyond a threshold value. In order to detect anomalous subgraphs, the current activity at a particular point of time is compared with the expected normal activity based on recent behavioral patterns. If the subgraph deviates significantly from the expected activity, it is classified as anomalous.

2.1.3.3 Dynamic Attributed Networks

For detecting anomalies from dynamic attributed networks, the attributes associated with nodes and/or edges are considered, in addition to the alterations in network structure over time. In spite of the fact that the additional information provided by the nodes and/or edges can improve the anomaly detection significantly, only very few papers are found on anomaly detection in dynamically evolving attributed networks. In this section, firstly the methods that have been developed for detecting anomalies from dynamic attributed networks are classified based on the underlying approach, and then an overview of the methods are provided.

Types of Approaches

The methods that have been proposed for detecting anomalies from dynamic attributed networks can be categorized into three: *tensor decomposition-based*, *probability-based*, and *signal processing-based* approaches.

a) Tensor decomposition-based approach: As explained in Section 2.1.3.2, the decomposition - based anomaly detection methods use matrices or tensors to represent the dynamic networks, and exploit their decomposition to detect the anomalies. The method proposed by Papalexakis et al. (2012) uses tensor decomposition to detect anomalies in dynamic attributed networks.

b) Probability-based approach: Based on probability theory, the approaches presented in Heard et al. (2010) build models of normal behavior of dynamic networks by analyzing several previous network instances. Every new incoming network instance is compared with this normal model, and any deviations are flagged as anomalous.

c) Signal processing-based approach: As discussed in Section 2.1.2.1, signal processing on graphs treat comparatively smaller anomalous subgraphs as signals embedded in a much bigger network taken as the background noise.

Anomalous Node, Edge, and Subgraph Detection

This section surveys the anomaly detection methods that have been developed for spotting anomalies in dynamic attributed networks.

Tensor decomposition-based approach: Papalexakis et al. (2012) presented a fast and parallelizable approach called ParCube, for speeding up sparse tensor decompositions by using random sampling techniques. ParCube is a method for PARAFAC decomposition and can process tensors that do not fit in memory. In order to represent attributed networks using tensors, the labels are to be mapped into real numbers. The anomalous nodes at particular time steps are detected by tracking the unusual patterns in the factors of tensor decomposition.

Probability-based approach: The probability-based method by Heard et al. (2010), discussed in Section 2.1.3.2, can detect anomalous nodes, edges, and subgraphs in dynamic networks with categorical attributes as well. A recent probability-based approach called HCODA (Pandhre et al. 2016) detects Holistic Community Outliers (HCOutliers) that are nodes connected to several nodes of other communities in attributed networks.

Signal processing-based approach: The proposed framework by Miller et al. (2013) exploits signal processing for graphs to find anomalous nodes and subgraphs from dynamic attributed networks. This framework treats a network as an instance drawn from a distribution of random graphs, and performs spectral analysis of graph residuals (i.e., the difference between the observed graph and its expected value) to determine the presence of anomalies. The expected value is obtained by assuming that the probability of occurring an edge between any two nodes is a function of the linear combination of the labels on the nodes. Apart from the aforesaid methods, the anomaly detection methods developed by Araujo et al. (2014) and Koutra et al. (2012), discussed in Section 2.1.3.2, can incorporate attributes as well, even though they are not applied in dynamic attributed networks.

In this section, the various anomaly detection methods that have been developed for dynamic social networks have ben discussed. Even though there has been considerable amount of work in developing anomaly detection methods for dynamic unattributed networks, it is observed that there is a minimal published information relating to mining dynamic attributed networks for anomalies. However, a qualitative comparison and summary of anomaly detection methods proposed for dynamic social networks is shown in Table 2.2.

2.1.4 Research Directions and Challenges

In this section, a multitude of methods for mining social networks for anomalies are examined. Even though there are a wide variety of methods for anomaly detection in social networks, this field is still relatively young and rapidly growing, and there is a significant scope for future research. In this section, the open issues and research challenges in this area are discussed.

Unlike the traditional anomaly detection methods which analyze independent and identically distributed data objects, anomaly detection in social networks is primarily based on the interactions among different individuals in the network.

Table 2.2: Summary and comparison of articles on anomaly detection in dynamic social networks

	Network		Anomaly					
Research Article	Unattributed	Attributed	Node	Edge	Subgraph	Event	Approach Used	Reporting of anomalies
Sun et al. (2007)	✓		✓		✓	✓	Matrix decomposition	Graph encoding cost for each time step
Akoglu and Faloutsos (2010)	✓		✓			✓	Matrix decomposition	z-scores
Yu et al. (2013)	✓		✓				Matrix decomposition	Node subsets
Baingana and Giannakis (2016)	✓		✓				Matrix decomposition	Node subsets
Sun et al. (2006)	✓		✓		✓	✓	Tensor decomposition	Reconstruction error
Kolda and Sun (2008)	✓		✓		✓	✓	Tensor decomposition	Reconstruction error
Gupta et al. (2012b)	✓		✓				Community-based	Community memberships
Gupta et al. (2012a)	✓		✓				Community-based	Anomaly scores at time step
Ji et al. (2013)	✓		✓				Community-based	Anomaly scores at each time step
Rossi et al. (2013)	✓		✓				Community-based	Role memberships
Priebe et al. (2005)	✓		✓	✓			Probability-based	Scan statistics
Huang and Zeng (2006)	✓			✓		✓	Probability-based	Likelihood scores
Pandit et al. (2007)	✓		✓				Probability-based	Node labels

Table 2.2 – Continued from previous page

	Network		Anomaly					
Research Article	**Unattributed**	**Attributed**	**Node**	**Edge**	**Subgraph**	**Event**	**Approach Used**	**Reporting of anomalies**
Thompson and Eliassi-Rad (2009)	✓				✓		Probability-based	Anomaly scores
Aggarwal et al. (2011)	✓			✓		✓	Probability-based	Likelihood at each time step
Chen et al. (2012d)	✓				✓		Community-based	Community outliers and time stamps
Miller et al. (2012)	✓		✓			✓	Signal processing	Node subsets
Mongiovi et al. (2013a)	✓				✓		Distance-based	Anomaly scores
Mongiovi et al. (2013b)	✓				✓		Distance-based	Anomaly scores
Heard et al. (2010)	✓	✓	✓	✓	✓	✓	Probability-based	p - values
Koutra et al. (2012)	✓	✓	✓		✓	✓	Tensor decomposition	Tensor factors at each time step
Araujo et al. (2014)	✓	✓	✓				Community-based	Tensor factors at each time step
Papalexakis et al. (2012)		✓	✓			✓	Tensor decomposition	Tensor factors for each time step
Miller et al. (2013)		✓	✓		✓	✓	Signal processing	Node subsets
Pandhre et al. (2016)		✓	✓				Probability-based	Node subsets

Moreover, as the social network is highly interconnected, the spreading of anomalies also needs to be taken care of. This makes the anomaly detection problem on social networks different from other forms of anomaly detection. Similar to the traditional anomaly detection problem, anomaly detection in social networks is also highly application-specific, and most of the approaches were developed by keeping a set of requirements and constraints in mind. Hence, a quantitative comparison among the methods is not practical as they identify different types of anomalies. Moreover, with the different methods discussed, deciding a particular algorithm for anomaly detection is a difficult task. There is no generic algorithm for anomaly detection in social networks. The factors that are to be considered when selecting an appropriate algorithm are the different aspects of the application such as the type of the network examined and the types of anomalies to be detected.

The challenges associated with traditional anomaly detection are already discussed in Section 1.1. Apart from those challenges, anomaly detection in social networks encounters the following challenges specific to network data:

- **Computational complexity:** In this era of Big Data, detecting anomalies in social networks is a computationally intensive task due to their huge size and dynamic nature. More specifically, in order to enumerate anomalous units in networks, the detection algorithms have to search the entire complex network, due to which the time and space complexities of the detection algorithms are often high. Therefore, it is hard to develop an efficient and scalable method for mining anomalies from social networks.

- **Streaming networks:** The area of streaming social networks is relatively new, and poses many new challenges as a result of the complexities in maintaining real-time structural summaries of the network streams arriving over time. They typically require real-time analytical techniques for anomaly detection. Developing scalable approaches for detecting anomalies in streaming networks is a major area for future research.

- **Dynamic attributed networks:** Even though there has been substantial amount of work in developing anomaly detection methods for dynamic unattributed networks, it is observed that there is a minimal published information relating to mining dynamic attributed networks for anomalies. Therefore, exploring and identifying the real-world applications for anomalies in such networks, and developing definitions and algorithms for such anomalies are promising research directions.

- **Maintaining history of updates:** When a network has evolved significantly, the dynamic anomaly detection algorithms do not exploit the past updates occurred to the nodes or edges. For instance, for a user having a strong interaction with a malicious user in the past and the edge is later deleted, the existence of such an edge should be considered while making future evaluations. The existing methods treat the edge removal as a simple edge alteration. Therefore, developing anomaly detection methods that can maintain history of dynamic updates is an open area of research.

- **Feature extraction:** Selection and extraction of appropriate, meaningful, and unique feature space are essential in many of the anomaly detection techniques on social networks. The features should be easy to compute, able to differentiate anomalous and normal objects, and able to resist noises. Considering the huge size of the networks, the selection and extraction of a suitable feature space that can map the real-world behavior of interest are challenging tasks in the anomaly detection process.

- **Performance evaluation:** Due to the lack of publicly available network datasets with explicit ground truths, the evaluation of the detection methods is often difficult. In many cases, researchers have to perform their experimental analysis on synthetic data by injecting anomalies or have to manually investigate the top ranked anomalies by using the real-world domain knowledge. Furthermore, there is no accepted standard for evaluating an anomaly detection system developed for social networks.

- **Anomaly detection in multi-layer social networks:** The social relationships between individuals are often multiple in nature. For example, same set of individuals can interact through different social networks such as Facebook, Twitter, and Google Plus. This information can be modeled by using a multi-layer network, where each layer captures the interaction in one social network (Boccaletti et al. 2014; Dong et al. 2014). The anomaly detection methods discussed in the previous sections have been developed for single-layer networks with only one mode of interaction. Combining the rich information from all the network layers for detecting anomalies is an interesting research direction in this area.

2.2 MULTI-LAYER NETWORKS

The study of multi-layer networks provides new insights into diverse areas of science including social systems, biology, physics, web, and engineering systems. Recently, multi-layer networks have become extremely popular in multi-disciplinary research domains and a substantial amount of effort has been dedicated to their mathematical modeling and characterization. More recently, a great effort has been directed to extending the classical single-layer network problems to multi-layer settings. Bródka et al. (2012) analyze the neighborhood in multi-layer networks and introduce cross-layer degree centrality, different variants of multi-layer degree centralities, and cross-layer clustering coefficient. Halu et al. (2013) extend the classical PageRank centrality to multi-layer networks. Brodka et al. (2011) address the problem of shortest-path discovery in multi-layer social networks.

The significant research directions in multi-layer networks include clustering/ community detection (Dong et al. 2012; Mucha et al. 2010; Rodriguez and Shinavier 2010; Berlingerio et al. 2011; Shiga and Mamitsuka 2012; Tang et al. 2012; Berlingerio et al. 2013b; Afsarmanesh and Magnani 2016; Jeub et al. 2015; Hmimida and Kanawati 2015), layer communities (Kao and Porter 2017), layer-reduction (De Domenico et al. 2015), link prediction (Ahmad et al. 2010; Rossetti et al. 2011; Pujari and Kanawati 2015; Jalili et al. 2017), information diffusion (Eslami et al. 2011; Ramezanian et al. 2015; Salehi et al. 2015) (For a comprehensive survey on spreading processes, the readers are referred toSalehi et al. (2015)), and visualization (De Domenico et al. 2014; Redondo et al. 2015; Renoust et al. 2015; Rossi and Magnani 2015). The methods proposed by Rodriguez and Shinavier (2010); Tang et al. (2012); Berlingerio et al. (2011) detect communities in multi-layer networks by transforming them into single-layer networks and by applying the existing community detection methods on the single-layer networks. This results in the loss of information embedded in individual layers. Dong et al. (2012) propose a spectral approach for clustering multiple graphs, and Afsarmanesh and Magnani (2016) identify overlapping communities in multiplex networks by extending the popular clique percolation method for single-layer networks.

In contrast to the above-mentioned research areas, anomaly detection in multi-layer networks is an unexplored area of research. Hence, in this work, a pioneer approach for anomaly detection in multi-layer networks is proposed.

Research Challenges

The challenges faced by researchers in multi-layer network analysis are as follows:

1. Multi-layer social networks are much more difficult to analyze than single-layered networks, because there is no well-known and widely-accepted mathematical framework and measures for them.

2. Multi-layer social networks are often heterogeneous, i.e., they can be directed vs. undirected, weighted vs. unweighted, signed vs. unsigned, or have different degree densities.

3. Real-world multi-layer social networks are often huge and non-trivially noticeable, since no single organization has full control over all the layers.

4. The structure of the multi-layer social networks changes over time leading to dynamic multi-layer social networks, which makes the analysis much more complex and time-consuming.

2.3 SPAMMER DETECTION IN SOCIAL NETWORKS

Most of the research on spam detection in social networks has been developed for detecting spam messages or individual social spammers (Benevenuto et al. 2010, 2008; Lee et al. 2010; Stringhini et al. 2010; Yang et al. 2013; Wang 2010). Limited amount of research has been dedicated to understand the social relationships existing among the spammers in social networks. The spam content detection usually includes content-based filtering (Benevenuto et al. 2010), URL blacklists (Gao et al. 2010a), and spam traps known as honeypots (Lee et al. 2010, 2011; Stringhini et al. 2010) to build classifier algorithms.

Initial works on spammer 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. (2011a,b). These works mainly focus on information diffusion in Twitter based on content using a supervised algorithm (Ratkiewicz et al. 2011a) and use of network information using a clustering algorithm (Ratkiewicz et al. 2011b). Gao et al. (2010a) present 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 performed. 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) analyze 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). These techniques employ 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) has developed a machine learning algorithm that uses textual features of spammer profile and their interactions in the network to develop spam signatures. Initially, it involves 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). Wang (2010) and DeBarr and Wechsler (2009) utilize more robust characteristics such as graph-based metrics and degree centrality-based metrics to detect spammers. Benevenuto et al. (2008, 2010) employ video rating, user behavioral characteristics, and topological characteristics to detect spammers in video sharing online social networks.

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 for detecting spammers that post the same tweet with the same meaning but different words, and 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 utilization of user-content for detecting spammers is often being reported as a violation of privacy by many users.

Finally, many existing studies on spammer detection 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 which monitor and log 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) employ 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.

Zheng et al. (2015) analyze the message content and user behavior in the social network to extract a set of features and apply the feature set on an SVM classifier to detect spammers and obtain better classification results. But, this approach requires more training time and requires manual adjustment for selecting parameters. On similar lines, Zheng et al. (2016) propose an extreme learning machine (ELM) based supervised classification approach to detect spammers by analyzing message content and user behavior. The ELM-based approach provides better performance than SVM-based approaches.

A summary and comparison of the features and methods used by popular spam detection techniques are presented in Table 2.3. The methods popularly used by researchers include supervised learning, URL blacklisting, clustering, and use of social honeypots to trap spammers. 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 provides most accuracy in detecting spammers but involves lot of computation to recognize the credibility of content. Next, the spam 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 work, all the three kinds of features, ie., content, network, and behavior, are employed to detect spammers. 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.

The overlapping community structure exists even for spammers in social networks. Spammers are known to form a close-knit communities 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

Table 2.3: Summary and comparison of articles on spam detection in social networks

	Features			**Methods**			
Research Article	**Content**	**Network**	**Behavior**	**Supervised**	**URL Blacklist**	**Clustering**	**Social Honeypot**
Benevenuto et al. (2008)		✓	✓	✓			
Yardi et al. (2009)		✓	✓	✓			
DeBarr and Wechsler (2009)		✓		✓			
Lee et al. (2010)	✓			✓			✓
Benevenuto et al. (2010)	✓		✓	✓			
Gao et al. (2010a)	✓		✓			✓	
Grier et al. (2010)	✓				✓		
Stringhini et al. (2010)	✓	✓		✓			✓
Wang (2010)	✓	✓		✓			
Mustafaraj and Metaxas (2010)		✓	✓	✓			
Chu et al. (2010)	✓	✓	✓	✓			
Ratkiewicz et al. (2011a)	✓			✓			
Ratkiewicz et al. (2011b)		✓				✓	
Song et al. (2011)		✓		✓			
Fire et al. (2012)		✓	✓			✓	
Yang et al. (2012)		✓				✓	
Hu et al. (2013)	✓	✓		✓			
Bhat and Abulaish (2013)		✓				✓	
Yang et al. (2014)	✓		✓	✓			✓
Zheng et al. (2015)	✓		✓	✓			
Zheng et al. (2016)	✓		✓	✓			

et al. 2008). Generally, the clustering coefficient is a good feature that can be exploited to detect RLA attacks. Hence, the authors use 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.

Fire et al. (2012) incorporate 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 edges 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.

The works of Yardi et al. (2009); Gao et al. (2010a); 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 networks. Compared to the existing literature, our work primarily focuses on detection of spammer community ecosystem - 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.

2.3.1 Research Directions and Challenges

Most of the existing work has been based on learning content or user based features to detect spammers. Commonly, the features used to detect spammers include the number of follow, the number of followers, the number of malicious URLs, follower to follow ratio, reputation, number of re-tweets, 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.

2.4 SUMMARY

In this chapter, a structured review of the various methods proposed for anomaly and spammer detection in social networks is presented. Mining social networks for anomalies is a challenging and computationally intensive task due to the huge size of the network and its dynamic nature. In the past decade, there are a wide variety of methods developed for social network anomaly detection in different problem settings. This chapter organizes the state-of-the-art methods into different categories based on the elementary approach followed by each method and briefly introduces the corresponding methods. Finally, the various research challenges and open issues for future research in this domain are discussed. With the different methods discussed, deciding a particular algorithm for anomaly detection is a difficult task. When selecting an appropriate algorithm, one has to consider different aspects of the application such as the type of the network being examined and the types of anomalies to be detected. This comprehensive review provides a better understanding of the various techniques that have been developed for mining social networks for anomalies and spammers. Additionally, this chapter presents a review on the significant research areas in multi-layer social networks and discusses the major research challenges in multi-layer social network analysis.

CHAPTER 3

PROBLEM DESCRIPTION

In this research work, the novel problem of anomaly detection on multi-layer social networks is put forward. Th first objective of this work is to develop an unsupervised, parameter-free, and network-feature based method to discover anomalous nodes or point anomalies in a multi-layer social network. In the social network arena, the majority of the nodes follow the rule of "friends of friends are often friends" and minority follows either "stars/near-stars" or "cliques/near-cliques". The nodes whose egonets follow star/near-star and clique/near-clique patterns can be linked to suspicious behavior and have been established as anomalies by previous researchers (Akoglu et al. 2010; Hassanzadeh et al. 2012; Gupta et al. 2013; Hassanzadeh and Nayak 2013a,b; Kaur and Singh 2017). Hence, in this work, we focus on detecting the anomalous nodes with such abnormal patterns in a multi-layer social network. Existing anomaly detection methods do not consider the multi-layer structure of human interactions. They have been devised for networks with only one type of interaction among the entities. Hence, it is essential to consider the multi-layer structure of social networks for anomaly detection, by combining the rich information embedded in multiple layers. The goal of this work is to rank the nodes in a multi-layer network according to their anomalousness or suspiciousness, by assigning anomaly scores to the nodes based on the degree of similarity of the nodes' egonets in different layers to stars or cliques.

The problem is formally stated as follows: Suppose we have an undirected and unweighted multi-layer social network $G = \{G^1, G^2, ..., G^L\}$ with a finite sequence of L network layers, each of which corresponds to one type of interaction. Each layer in the multi-layer network can be considered as a network on its own, and the l^{th} layer of the multi-layer network is represented as G^l, where V^l and E^l are the set of nodes and set of edges respectively in layer l. A node can be present in one or more layers. If all the nodes do not appear in every layer of the multi-layer network, a union of the nodes in the network layers is taken as the shared node set, i.e., $V = \bigcup_{l=1}^{L} V^l$ and $n = |V|$, number of nodes in V. Now a sequence of L $n \times n$ adjacency matrices or *sociomatrices* (Wasserman and Faust 1994) can be defined, one for each layer: A_G = $\{A^{[1]}$, $A^{[2]}$, ..., $A^{[L]}\}$, where $A^{[l]} = \{a_{ij}^{[l]}\}$ and $a_{ij}^{[l]} = 1$ if and only if i and j are connected in layer l. The first objective of this work is to develop an unsupervised approach to rank the nodes of the multi-layer network G according to the anomaly scores that are calculated based on the degree of similarity of the nodes' egonets in different layers to stars or cliques.

This work secondly addresses the detection of spammer communities in Twitter. Spamming is the most predominant form of anomalous activity prevalent in online social networks that involves malicious users sending unsolicited messages to legitimate users with the intention of wasting their time, bandwidth, and money. Being one of the fastest growing online social networks, Twitter has become a primary target platform for social spammers. A substantial amount of research work has been carried out for detecting spam messages and social spammers in Twitter. However, one of the important issues in Twitter is that the social spammers collaborate with each other and form collective anomalies or spammer communities to spread spam messages to a large set of legitimate users. Accordingly, it is highly desirable to detect such spammer communities prevailing in Twitter. Hence, the second objective of this work is to develop an unsupervised approach for detecting spammer communities in Twitter by analyzing the spammers' community features and robust characteristics of these accounts that are difficult to evade by the spammers. 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. The goal is to classify the accounts as spammers and legitimate users, and find social connections between the spammers to unearth the spammer communities.

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 multi-layer 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 multi-layer social network considered in this 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.

Given the directed and attributed Twitter multi-layer 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, the aim of this work 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.

To summarize, the primary goal of this research work is to develop unsupervised approaches for detecting anomalies in multi-layer social networks by using graph-theoretic features of the networks and data mining techniques. The primary objective of the work is further subdivided into two specific objectives as listed below:

1. Developing an unsupervised approach to detect anomalous nodes in a multi-layer social network by analyzing the structure of the network

2. Developing an unsupervised approach to detect anomalous spammer communities in a multi-layer social network by analyzing the structure and attributes of the network

CHAPTER 4

ANOMALOUS NODE DETECTION IN MULTI-LAYER SOCIAL NETWORKS

4.1 INTRODUCTION

The definition of anomaly is usually subjective and depends on the problem at hand. If we need to identify the individual users whose behavior deviates considerably from the usual behavior of users in the social network, a subset of users or nodes are viewed as anomalies. Anomalous nodes are also known as point anomalies, as they are scattered in the network. In our problem setting, an anomaly is a node in the multi-layer social network whose egonets in different layers follow either "stars/near-stars" or "cliques/near-cliques" patterns.

The objective of the work presented in this chapter is to develop an unsupervised, parameter-free, and network-feature based method to discover anomalous nodes in a multi-layer social network. As already mentioned in Chapter 3, in the social network arena, the majority of the nodes follow the rule of "friends of friends are often friends" and minority follows either "stars/near-stars" or "cliques/near-cliques". The nodes whose egonets follow star/near-star and clique/near-clique patterns can be linked to suspicious behavior and have been established as anomalies (Akoglu et al. 2010; Hassanzadeh et al. 2012; Gupta et al. 2013; Hassanzadeh and Nayak 2013a,b; Kaur and Singh 2017). Hence, this work focuses on identifying anomalous nodes with such abnormal patterns in a multi-layer social network. The state-of-the-art anomaly detection methods do not consider the multi-layer structure of human interactions. They have been devised for networks with only one type of interaction among the entities. Hence, it is essential to consider the multi-layer structure of social networks for anomaly detection, by combining the rich information embedded in multiple layers. The objective of this work is to rank the nodes in a multi-layer network according to their anomalousness or suspiciousness, by assigning anomaly scores to the nodes based on the degree of similarity of the nodes' egonets in different layers to stars or cliques.

This chapter contributes to the existing literature in the following ways:

- Introducing and studying the problem of anomaly detection on multi-layer networks. To the best of our knowledge, the proposed approach is a pioneering work on detecting anomalies in multi-layer networks.
- Proposing Anomaly Detection On Multi-layer Social networks, ADOMS, an unsupervised, network feature-based, and parameter-free methodology, to automatically rank the nodes of a multi-layer network based on the extent of similarity of the nodes' egonets in different layers to cliques or stars.
- Parallelizing the feature extraction and anomaly detection operations on different layers of the multi-layer network to significantly speed up the computation, by distributing the tasks to different cores of the machine.
- Evaluating the proposed approach via extensive experiments on multiple real-world multi-layer network datasets. The experimental results substantiate that the proposed approach can effectively detect anomalous nodes in multi-layer social networks.
- Developing a baseline method by aggregating the layers of the multi-layer network into a single-layer network, as there is no known method for anomaly detection on multi-layer social networks. The results of the proposed approach are compared with that of the baseline method and it is found that the proposed approach outperforms the baseline.

The rest of this chapter is organized as follows. Section 4.2 presents the terminologies used in this chapter, and Section 4.3 deals with the problem statement which includes the formal definition of the problem. Section 4.4 presents the proposed solution methodology for the problem of anomalous node detection in multi-layer networks. Section 4.5 discusses the experimental analysis and discussion on results, and Section 4.6 presents a summary of the chapter.

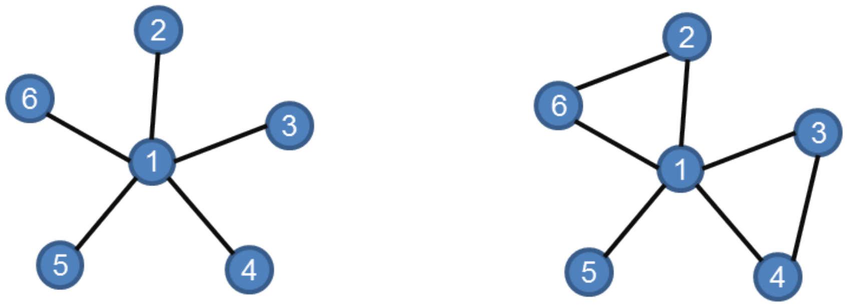


Figure 4.1: Star/Near-star. In star structure, the neighbors of the node are fully disconnected

4.2 DEFINITIONS AND TERMINOLOGIES

To set scene for this chapter, a brief overview of the terminologies used in the chapter is presented below:

1. *Egonets:* In a social network, an egonet is an individual's personal network. In other words, an egonet of a node (called "ego") is the one-step neighborhood subgraph of the node. An egonet describes the local network structure of a node, i.e. the network around a single node. It consists of the node itself, its immediate neighbors (called "alters"), and all the edges or ties among them. For example, in Figure 1.2 the egonet of node 1 in Layer I is the subgraph containing the nodes 1, 2, and 4. In Layer II, the egonet of node 1 is the subgraph containing the nodes 1, 4, and 5 and in Layer III, it is the subgraph conatining the nodes 1, 2, 4, and 5. Each alter in an egonet has its own egonet, and all egonets interconnect to form the social network.

2. *Stars/Near-stars:* In a star topology, the neighbors of the node are fully disconnected with each other. The number of edges in a star with n nodes is $(n - 1)$. In a near-star pattern, few of the neighbors of the node can be connected to each other. An example for star/near-star is shown in Figure 4.1.

In social networks, a near-star topology can correspond to a highly influential person such as a movie star, a politician, or a sports person. E-Commerce companies can use this information to advertise their products on the influential person's network to obtain maximum spread. It can also be a spammer or fraudster who sends unsolicited or fraudulent messages to random legitimate users who are otherwise unconnected. Therefore, stars/near-stars can signify suspicious behavior.

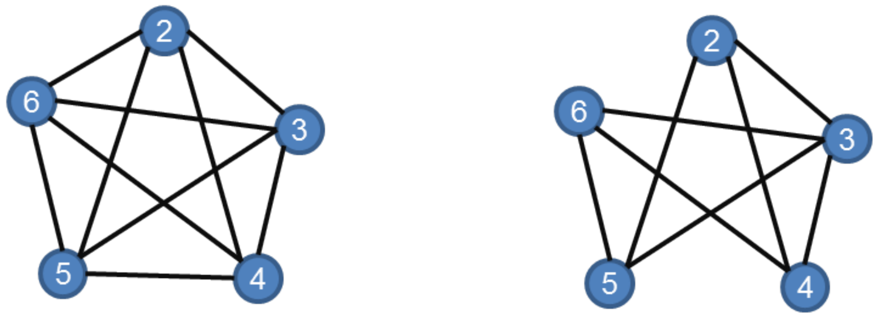


Figure 4.2: Clique/Near-clique. In a clique structure, the neighbors are fully connected to each other

3. *Cliques/Near-cliques:* A clique is a fully-connected subgraph in a network. If the egonet of a node is a clique, all the neighbors of the node are connected to each other. The number of edges in a clique of n nodes is $n * (n - 1)/2$. A node follows a near-clique pattern, if few of the neighbors of the node are disconnected. An example for clique/near-clique is shown in Figure 4.2.

In social networks, a clique can correspond to a close-knit community of friends, a group of spammers (spammer community) who are fully connected to each other to evade from being detected by spam detection algorithms, or a group of fraudsters who collude with each other to commit a scam such as electronic auction fraud. Therefore, cliques/near-cliques can signify suspicious behavior.

4. *Local Outlier Factor:* Local Outlier Factor (LOF) (Breunig et al. 2000) is a widely used unsupervised and density-based outlier detection technique. Density-based outlier detection techniques consider an object as an outlier if its density is comparatively less than that of its neighbors. LOF compares the local density of an object to the local densities of its neighboring objects, and assigns a high outlier score if the local density of the object is considerably less than the average local density of its neighbors. If the object is in a dense area, its local density will be similar to that of its neighboring objects and therefore, its outlier score will be low.

LOF assigns an anomaly score close to one for normal objects, and much larger scores for anomalous objects. In Figure 4.3, the local density of point A is much lower than that of its neighbors, and therefore it will be assigned a high anomaly score compared to its neighbors. LOF is more advantageous compared to other density-based outlier detection methods if the objects in the data set lie in different density regions.

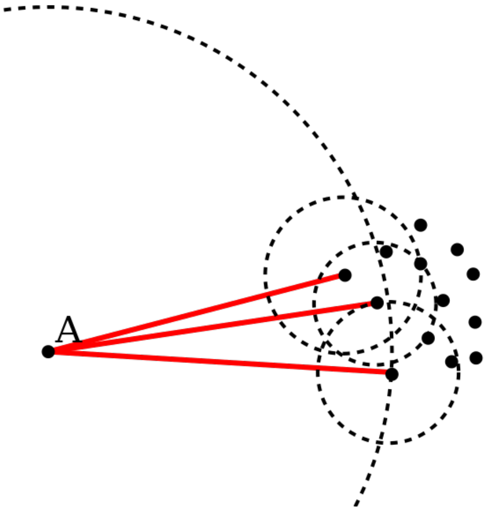


Figure 4.3: Local Outlier Factor

4.3 PROBLEM STATEMENT

The problem is formally stated as follows: Suppose we have an undirected and unweighted multi-layer social network $G = \{G^1, G^2, ..., G^L\}$ with a finite sequence of L network layers, each of which corresponds to one type of interaction. Each layer in the multi-layer network can be considered as a network on its own, and the l^{th} layer of the multi-layer network is represented as G^l, where V^l and E^l are the set of nodes and set of edges respectively in layer l. A node can be present in one or more layers. If all the nodes do not appear in every layer of the multi-layer network, a union of the nodes in the network layers is taken as the shared node set, i.e., $V = \bigcup_{l=1}^{L} V^l$ and $n = |V|$, number of nodes in V. Now a sequence of L $n \times n$ adjacency matrices or *sociomatrices* (Wasserman and Faust 1994) can be defined, one for each layer: A_G = $\{A^{[1]}$, $A^{[2]}$, ..., $A^{[L]}\}$, where $A^{[l]} = \{a_{ij}^{[l]}\}$ and $a_{ij}^{[l]} = 1$ if and only if i and j are connected in layer l. The objective of the work presented in this chapter is to develop an unsupervised approach to rank the nodes of the multi-layer network G according to the anomaly scores that are calculated based on the degree of similarity of the nodes' egonets in different layers to stars or cliques. The major symbols used in this chapter are defined in Table 4.1.

Table 4.1: Symbols and definitions used in this chapter

Symbols	**Definitions**
G	Multi-layer network
V	Set of nodes in G
L	Number of layers in G
n	Number of nodes in G
G^l	l^{th} network layer of G
E^l	Set of edges in G^l
A_G	Set of $n \times n$ adjacency matrices corresponding to G
$A^{[l]}$	$n \times n$ adjacency matrix of G^l with elements $a_{ij}^{[l]}$
N_i^l	Number of nodes in the egonet of node i in G^l
E_i^l	Number of edges in the egonet of node i in G^l
$aScore_i^l$	Anomaly score for node i in G^l
LR_i^l	Layer relevance of layer l for node i
$multiScore_i$	Anomaly score of node i in the multi-layer network G
G_A	Aggregated topological network of G
A	Adjacency matrix corresponding to G_A

4.4 PROPOSED METHODOLOGY

This section provides the solution methodology for the problem of anomaly detection on multi-layer social networks. At first, an egonet is generated for each node in each layer of the network. Then, an anomaly score is computed for each node in each individual network layer based on the local features of its egonet such as the number of edges and edges in the egonet. The anomaly scores of the corresponding nodes from individual layers are then combined, based on the relevance of the layers, to form the anomaly scores of the nodes in the multi-layer network. The nodes of the multi-layer network are then ranked based on the anomaly scores.

The proposed approach for detecting anomalies in multi-layer networks namely Anomaly Detection On Multi-layer Social networks (ADOMS) is divided into two logical phases: Phase 1 and Phase 2. The details of Phase 1 and Phase 2 are described in Sections 4.4.1 and 4.4.2 respectively. In Phase 1, the anomaly scores for the nodes in individual layers are calculated based on their degree of cliqueness or starness in the layers. In Phase 2, the anomaly scores of the nodes in the multi-layer network are computed by employing the anomaly scores of the nodes in the individual network layers from Phase 1, and the nodes are then ranked according to the anomaly scores.

4.4.1 Phase 1: Computing Anomaly Scores of Nodes in Individual Network Layers

The objective of the first phase of ADOMS approach is to mine patterns and to assign anomaly scores to the nodes in the individual layers of the multi-layer network. Anomaly scores are assigned to each and every node in the individual network layers according to the degree of similarity of the nodes' egonets to stars or cliques. The primary task in assigning anomaly scores to data items is identifying a suitable feature space and finding the patterns or rules obeyed by the majority of data items within that feature space. The data items that violate these patterns or rules are ranked as anomalies. The major steps involved in the first phase of our approach are explained below:

1. Feature Extraction: The first step in anomaly detection in social networks is to identify an appropriate feature space that can map the online behavior of users. Identifying a suitable feature space is a key challenge for spotting anomalies in social networks. The features should be effective and meaningful, and should be fast to compute. In addition, the numerical features are to be selected in such a way that the deviation from the normal nodes can be measured and compared in an efficient way. In this work, the numerical features that characterize and summarize the neighborhood of nodes in the individual network layers are identified. The characteristics of the local neighborhood of nodes such as edge count and node count in the egonets of the nodes, as specified by Akoglu et al. (2010), are selected as feature space. The feature pairs of the number of edges and the number of nodes in the egonet of a node are easy to compute from the network, and can characterize the neighborhood and online behavior of the node effectively. After identifying the suitable feature space, the numerical features that reflect the online behavior of users are extracted from each network layer to form a two-dimensional feature space. More specifically, the online behavior of users in individual social network layers are transformed into numerical feature pairs: edge count and node count of egonets. Consequently, a point in the two-dimensional vector space corresponds to a node in the individual network layer. Therefore, a two-dimensional feature space is obtained for each network layer in the multi-layer social network.

Algorithm 4.1 Computing anomaly scores of nodes in individual network layers

```
Input: Multi-layer network $G$
Output: Anomaly scores of nodes in each layer of the multi-layer network
for each layer $l$ of $G$ do
    for each node $i$ do
        Identify the 1-step neighborhood (egonet) of node $i$
        Extract the number of nodes $N_i^l$ and number of edges $E_i^l$ in the egonet of node $i$ to form a two-dimensional feature space;
        Compute $aScore_i^l = LOF(E_i^l, N_i^l)$, the anomaly score for node $i$ in layer $l$;
    end for
end for
return $aScore$;
```

2. Pattern Mining: One of the main challenges of anomaly detection is understanding the typical/normal behavior followed by majority of data items in the set. The suspicious user behavior can be identified by finding the deviation from this normal behavior. In this step, the pattern obeyed by normal nodes, within the extracted low-dimensional feature space, is discovered. The neighborhood of normal nodes follow the pattern of a power-law, and the points that deviate from this pattern can be flagged as anomalies (Akoglu et al. 2010). The Egonet Density Power-Law (Akoglu et al. 2010) that defines the correlation between the edge count and node count in the egonet of a node is adopted for representing the normal behavior obeyed by majority of nodes. According to Egonet Density Power-Law, the feature pairs, the number of edges E_i^l and the number of nodes N_i^l in the egonet of a node i in layer l, are correlated as

$$E_i^l \propto {N_i^l}^{\theta},$$

where $1 \leq \theta \leq 2$ is the power-law exponent. Consequently, E_i^l is approximately equal to N_i^l for near-stars ($\approx N_i^l - 1$), and is close to ${N_i^l}^2$ for near-cliques ($\approx N_i^l * (N_i^l - 1)/2$). Because of the power-law relationship, E_i^l and N_i^l follow a linear correlation with slope θ in log-log scales. In our experiments, the power-law exponent θ ranges from 1.02 to 1.55.

3. Anomaly Scoring: As the Egonet Density Power-Law defines the correlation between the extracted feature pairs of edge count and node count, the deviation from the normal pattern can be identified using a traditional non-network based anomaly detection method. Any outlier detection method that assigns anomaly scores to data points can be employed. The well-known traditional non-network based anomaly detection method Local Outlier Factor (LOF) (Breunig et al. 2000) is chosen because it works efficiently on low-dimensional data points. LOF assigns a measure of outlierness or anomaly score to each data point. Hence, LOF is applied on the two-dimensional data points and the anomaly scores are generated. Specifically, the anomaly score for node i in layer l is computed as

$$aScore_i^l = LOF(E_i^l, N_i^l) \tag{4.1}$$

Consequently, the anomalies detected in this two-dimensional feature space correspond to the specific structures of stars/near-stars and cliques/near-cliques in the individual network layers. The pseudocode of Phase 1 of ADOMS is shown in Algorithm 4.1.

As there are no interdependency among the operations on individual layers of the multi-layer network, the feature extraction and anomaly detection operations on different layers (Phase 1) can be performed in a parallel and distributed manner. The operations can be distributed to different cores of the processor for parallel execution. Thus, significant speed up can be achieved.

4.4.2 Phase 2: Ranking Nodes of the Multi-layer Network According to the Anomaly Scores

The most challenging step of multi-layer network analysis is effectively integrating the information hidden in different layers. The second phase of ADOMS finds the anomalies in the multi-layer network by combining the layer-wise anomalies obtained in the first phase. The anomaly score of a node in the multi-layer network is computed as a function of the anomaly scores of the node in the individual network layers. More specifically, the anomaly scores of the nodes in the multi-layer network are obtained by taking linear combinations of anomaly scores of the nodes in the individual network layers. The weight considered for linear combination is a measure of the density of connectivity McPherson et al. (1992); Hanneman and Riddle (2005) of a node and its

Algorithm 4.2 Computing anomaly scores of nodes in the multi-layer network

```
Input: Layer-wise anomaly scores of nodes in $G$
Output: Nodes ranked according to the anomaly scores
for each node $i$ do
    Find total edges in the egonets of $i$ in all layers, $Total\_edges_i = \Sigma_{l=1}^{L} E_i^l$
end for
for each layer $l$ do
    for each node $i$ do
        Find layer relevance, $LR_i^l = E_i^l / Total\_edges_i$
    end for
end for
for each node $i$ do
    $multiScore_i = \Sigma_{l=1}^{L} (LR_i^l . aScore_i^l)$
end for
return Ranking of nodes in descending order of $multiScore$
```

neighbors in a layer compared to other layers. It shows whether they are densely connected or weakly connected to each other in that particular layer in comparison with other layers. When a node and its neighbors are densely connected in a network, the tie strenth among them is high (McPherson et al. 1992). A high density also signifies a larger number of connections among the nodes, and nodes in such a network have a high chance of being connected among each other (Luarn et al. 2014). Therefore, if the connectivity among the neighbors of a node is high in a network layer, the activity of the node is high in the layer, and that particular layer is more important for the connectivity of the node in the multi-layer network. A weight termed as *layer relevance* is defined to each node in every layer of the multi-layer network, and is defined as the relevance of the layer for the connectivity of the node compared to its connectivity in other layers of the network. The pseudocode for Phase 2 is given in Algorithm 4.2 and the associated key steps are described below:

1. Compute layer relevance: As each characteristic of relationships may have different relevance in the real-world, assigning different importance to individual layers is more applicable than assigning uniform importance to each layer of the network. Therefore, a layer relevance is assigned to each node in every layer of the multi-layer network according to the layer's characteristics. The layer relevance of a node in a network layer signifies the connectivity of the egonet in the layer compared to other layers in the multi-layer network. It indicates how a node and its neighbors are connected to each other in a network layer, compared to the other layers. If the connectivity of a user is sparse in a layer, the activity of the user in that layer is less. On the other hand, if the user is connected to many other users and the connectivity among them is dense in a layer, that layer is more relevant to the connectivity of the user in the multi-layer network. In other words, the layer relevance shows the importance of that particular layer for the connectivity of the node in the multi-layer network. The layer relevance of a node with respect to a layer is computed as the ratio of the number of edges in the egonet of the node in that layer to the total number of edges in the egonets of the node in all the layers in the multi-layer network. More formally, the layer relevance of node i with respect to layer l is defined as

$$LR_i^l = \frac{E_i^l}{\sum_{l=1}^{L} E_i^l} \tag{4.2}$$

2. Compute the final anomaly score: After computing the layer relevance of the nodes in the individual network layers, the anomaly scores for the nodes in the multi-layer network are computed as linear combinations of anomaly scores in the individual network layers. The layer relevance of the nodes are considered as the constants for linear combination. The layer relevance indicates how much the particular layer contributes to the anomaly score of the node in the multi-layer network. If a node is connected to many other nodes and the connectivity among them is dense in a layer, that layer contributes more to the anomaly score of the node in the multi-layer network, compared to other layers. Similarly, if the connectivity of a node is sparse in a layer, that layer contributes less to the anomaly score of the node in the multi-layer network. Therefore, the layer with more layer relevance for a node imparts more proportion to the total anomaly score of the node in the multi-layer network. Hence, the anomaly score of node i in the multi-layer network is computed as

$$multiScore_i = \sum_{l=1}^{L} LR_i^l.aScore_i^l \tag{4.3}$$

3. Anomaly ranking: Finally, the anomaly scores of nodes are sorted in a descending fashion and the ranking of the nodes are returned with the corresponding node indices.

In other words, the nodes in the multi-layer social network are ranked based on their degree of deviation from the norm or anomalousness.

The advantage of anomaly ranking is that it overcomes binary classification of objects into anomalies and normal, which is not sufficient in many application areas. With anomaly ranking, the analysts are able to explore a manageably-small subset of top-ranked objects first (Chandola et al. 2009). In addition, they can decide a domain-specific cutoff threshold between anomalous and normal objects in a flexible manner.

4.5 EXPERIMENTAL RESULTS AND ANALYSIS

Extensive experiments are carried out on six real-world multi-layer networks to validate the effectiveness and efficiency of the proposed ADOMS approach. This section presents the experimental analysis of the ADOMS approach.

4.5.1 Experimental Setup

We implement the algorithms of Phase 1 and Phase 2 and perform the evaluation on an Intel Core I7 CPU@3.40 GHz machine with 8-core processor and 8 GB RAM running Ubuntu 15.10 operating system. The algorithms are implemented in R programming language making use of the igraph[1] library. The overall execution time of the approach is reduced by implementing the first phase of ADOMS in parallel and distributed manner, as there are no dependencies among the operations in different layers. The anomaly scores of the nodes in different layers are computed by applying Phase 1 on each layer in parallel by distributing the work to different cores of the processor. ADOMS is implemented by using the *parallel* package in R by distributing the workload to six cores of the processor.

In the experimental evaluation, the potential and capabilities of the ADOMS approach are demonstrated on six real-world multi-layer networks for identifying meaningful anomalies.

4.5.2 Datasets

This section presents a description of the real-world multi-layer networks considered for our experimental analysis. For the experimental evaluation, the proposed approach

[1] http://igraph.sourceforge.net/

is applied on six undirected and unweighted real-world multi-layer network datasets - two collaboration networks, two terrorist networks and two social interaction networks. The basic characteristics of the networks are summarized in Table 4.2 and described in further detail below.

Noordin Top Terrorist Network: Noordin Top Terrorist dataset (Roberts and Everton 2011) is a four-layer multi-layer network of Indonesian terrorists. The layers represent information on communication, financial, operation, and trust relationships among a group of 78 terrorists headed by Noordin Mohammad Top. Noordin Mohammad Top built a personal terrorist group and was Indonesia's most wanted terrorist, prior to his death in 2009. He was responsible for several bombing attacks during 2003 to 2005. An exhaustive multi-layer analysis of this network has been performed by Battiston et al. (2014). The layer map is: Financial 1, Communication 2, Operational 3, Trust 4.

Social Evolution Dataset: Social Evolution experiment (Madan et al. 2012) was conducted to study the everyday life of a social community. The experiment covered more than 80% of students residing in an MIT dormitory. The students were surveyed on the different social relationships among them. The multi-layer network of the social evolution experiment consists of five layers representing the five relationships among the students and the layer map is the following: CloseFriend 1, SocializeTwicePerWeek 2, PoliticalDiscussant 3, FacebookAllTaggedPhotos 4, BlogLivejournalTwitter 5.

Aarhus: Aarhus multiplex network (Magnani et al. 2013) is a multi-layer social network representing the social interactions of a research department at Aarhus University. The multi-layer social network consists of five types of online and offline interactions namely Facebook, Leisure, Work, Co-authorship, and Lunch among the employees of the department of Computer Science at Aarhus. There are 61 nodes in total with 620 connections among them. The layer map is: Lunch 1, Facebook 2, Co-authorship 3, Leisure 4, Work 5.

DBLP_C: DBLP_C is a co-authorship network from the popular computer science bibliographic database DBLP[2] extracted by Berlingerio et al. (2013a). The conference names are the layers of the network. The authors of the conferences are the nodes of the network, and two authors are connected in a layer if they wrote one or more papers together in the conference. The network consists of six layers (conferences) and the

[2] http://dblp.uni-trier.de/

Table 4.2: Characteristics of the multi-layer networks analyzed in this study

Dataset	Nodes	Edges	Layers	Description
Noordin Top (Battiston et al. 2014)	78	911	4	Network of Indonesian terrorists
Social Evolution (Dong et al. 2011)	84	31,918	5	Social evolution experiment
Aarhus (Magnani et al. 2013)	61	620	5	Employee interaction network
DBLP_C (Berlingerio et al. 2013a)	6,771	19,345	6	Co-authorship network of conference authors
arXiv (De Domenico et al. 2015)	14,065	59,026	13	Co-authorship network of arXiv authors
GTD (Berlingerio et al. 2011)	2,509	32,279	124	Systematic data of terrorist incidents

layer map is the following: VLDB 1, SIGMOD 2, CIKM 3, SIGKDD 4, ICDM 5, SDM 6.

arXiv: Another dataset used in this chapter is a multi-layer co-authorship network of the free scientific repository arXiv[3] built by De Domenico et al. (2015). The network consists of 13 layers corresponding to different arXiv categories. The authors considered exclusively the articles with the word “networks” in the abstract or the title up to May 2014. The layer map is the following: physics.soc-ph 1, physics.data-an 2, physics.bio-ph 3, math-ph 4, math.OC 5, cond-mat.dis-nn 6, cond-mat.stat-mech 7, q-bio.MN 8, q-bio 9, q-bio.BM 10, nlin.AO 11, cs.SI 12, cs.CV 13.

GTD: Another dataset used in this chapter is a multi-layer network constructed by Berlingerio et al. (2011) from the Global Terrorism Database[4] (GTD). GTD is the most extensive database on terrorist attack incidents in the world. The GTD multi-layer network used in this chapter depicts the terrorist attacks occurred during the years 1970-2008. The nodes of the network are terrorist organizations and they are connected to each other if they have attacked the same country in the same year. In order to be connected, the terrorist groups attacked the same country in same year, but need not be collaborated for the attack. The network contains 2,509 terrorist organizations (nodes) active in 124 countries (layers).

[3] https://arxiv.org/

[4] https://www.start.umd.edu/gtd/about/

The Egonet Density Power-Law equations of the different layers of the datasets are shown in Table 4.3. In our experiments, the power-law exponent θ ranges from 1.02 and 1.55.

4.5.3 Baseline Method

As there is no known method for anomaly detection on multi-layer networks, it is hard to find a baseline method to compare with. As mentioned in Section 1.4, the standard way of representing a multi-layer network is to aggregate the information present in different layers together to obtain a single-layer network so that the traditional network analysis tools can be directly applied. Hence, for comparison purposes, an interesting baseline method would be to first generate a single-layer network called aggregated topological network (Battiston et al. 2014) from the multi-layer network by taking a union of the network layers and then applying the single-layer network anomaly detection algorithm over the aggregated network.

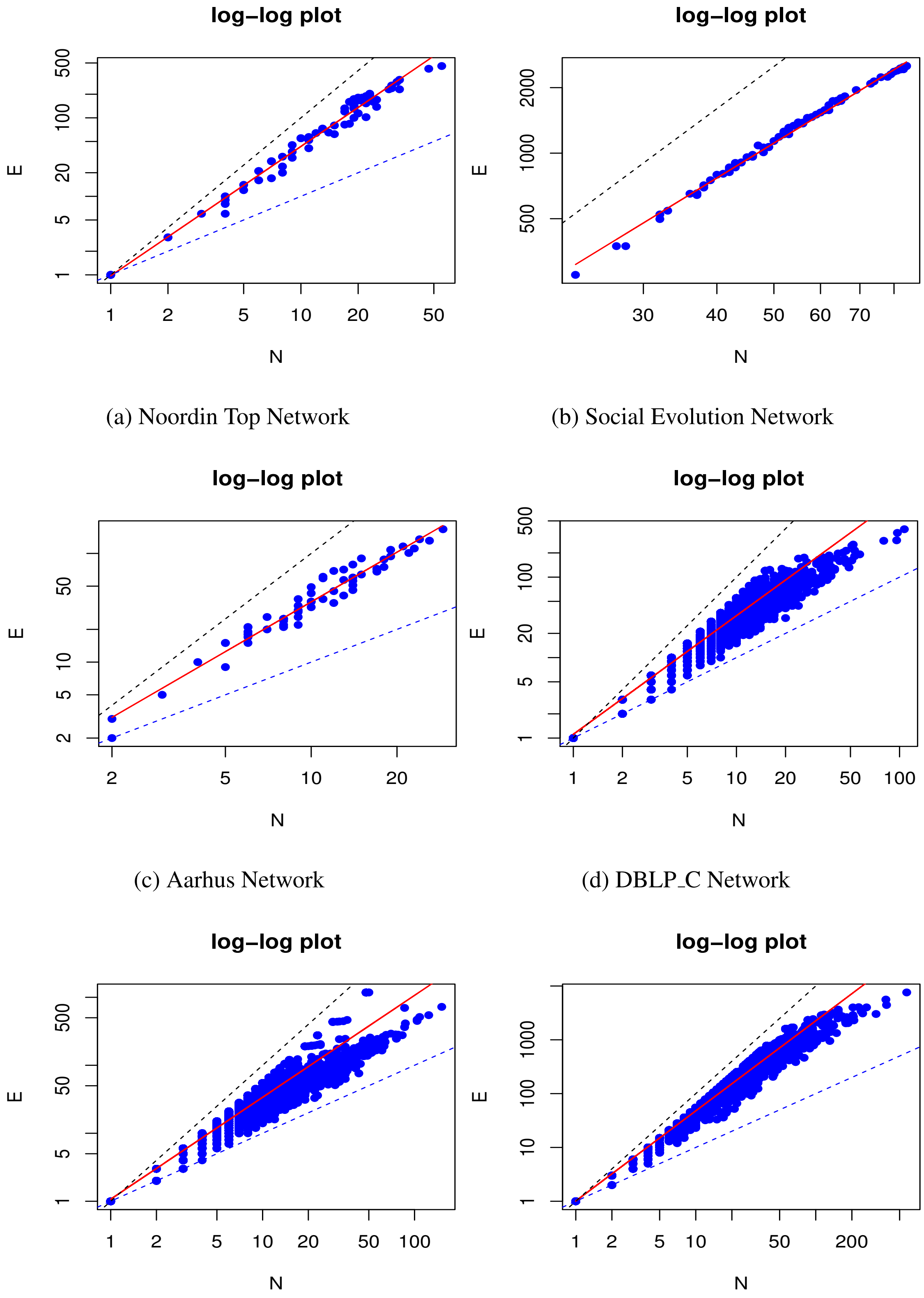
The aggregated topological network is constructed by flattening the layers of the multi-layer network. Given a multi-layer network G, the aggregate topological network of G is denoted as $G_A = (V, E_A)$, where V is the set of nodes in G, $E_A = \{E^1 \cup E^2 \cup ... \cup E^L\}$, and $E^1, E^2, ..., E^L$ are the sets of edges in the different layers of the multi-layer network. The adjacency matrix corresponding to the aggregated topological network is $A = \{a_{ij}\}$, where

$$a_{ij} = \begin{cases} 1, & \exists 1 \leq l \leq L : a_{ij}^{[l]} \neq 0. \\ 0, & \text{otherwise.} \end{cases} \tag{4.4}$$

In order to discover anomalous nodes whose egonets are near-cliques or near-stars in the aggregated topological network, the feature pairs - edge and node counts of egonets - are extracted to form a two-dimensional feature space. The aggregated topological network follows the Egonet Density Power-Law (Akoglu et al. 2010) that defines the correlation between edge count and node count of the egonets. The power-law equations of the aggregated networks of different datasets are shown in the last column of Table 4.3 and the Egonet Density Power-Law plots (in log-log scale) of the aggregated topological networks are shown in Figure 4.4.

Table 4.3: Power-law equations of different layers of the multi-layer network datasets

Dataset	**Layer 1**	**Layer 2**	**Layer 3**	**Layer 4**	**Layer 5**	**Aggr. Network**
Noordin Top	$E = 1.38 * N^{1.29}$	$E = 1.64 * N^{1.19}$	$E = 3.82 * N^{1.19}$	$E = 1.35 * N^{1.47}$	-	$E = 3.32 * N^{1.25}$
Social Evolution	$E = 2.17 * N^{1.31}$	$E = 2.30 * N^{1.48}$	$E = 1.93 * N^{1.49}$	$E = 3.18 * N^{1.48}$	$E = 3.07 * N^{1.47}$	$E = 2.65 * N^{1.55}$
Aarhus	$E = 2.32 * N^{1.13}$	$E = 1.51 * N^{1.33}$	$E = 1.20 * N^{1.13}$	$E = 1.32 * N^{1.20}$	$E = 1.79 * N^{1.19}$	$E = 1.58 * N^{1.38}$
DBLP_C			No. of layers = 6			$E = 1.871 * N^{1.17}$
arXiv			No. of layers = 13			$E = 2.57 * N^{1.28}$
GTD			No. of layers = 124			$E = 13.22 * N^{1.02}$



(a) Noordin Top Network

(b) Social Evolution Network

(c) Aarhus Network

(d) DBLP_C Network

(e) arXiv Network

(f) GTD Network

Figure 4.4: Log-log plots of the aggregated topological networks

The red lines in the plots correspond to the least-squares regression fitting lines of the two-dimensional feature space of edge count and node count. The blue and black dashed lines with slopes 1 and 2 symbolize stars and cliques respectively. The blue line in the log-log plot of Social Evolution network crosses the boundaries of the plot, and therefore it is not shown in the plot.

The anomaly scores of the nodes in the network are then computed as in Phase 1 of ADOMS. More specifically, as the edge count and node count of the egonets of nodes in the aggregated network are highly correlated, LOF can be applied over the two-dimensional feature space to generate the anomaly scores. Thus, after applying LOF over the extracted feature-space, the nodes of the network are ordered in descending fashion according to the anomaly scores to obtain the anomaly ranking. The baseline method will be referred as Agg+AD.

4.5.4 Results and Discussion

The validation or evaluation of anomaly detection techniques is a quite difficult and challenging issue because of the following two reasons: (i) the unavailability of labeled benchmark datasets with annotated ground truth (data objects labeled as normal and anomalous) (Chandola et al. 2009; Savage et al. 2014), and (ii) there is no standard technique for the evaluation of an anomaly detection method (Akoglu et al. 2015; Bindu and Thilagam 2016). However, the results of the proposed approach are evaluated on real-world multi-layer networks using the baseline method and case studies. In this section, the results obtained by applying the proposed approach to the different multi-layer network datasets are analyzed and compared with that of the baseline method. The proposed ADOMS approach and the baseline method Agg+AD are compared along two dimensions: (i) *anomaly ranking*, and (ii) *running time*. From the experimental results, it is observed that ADOMS outperforms the baseline in both anomaly ranking and running time dimensions.

a) Anomaly Ranking

With respect to anomaly ranking, the top ten ranked nodes obtained from ADOMS and Agg+AD are compared. A qualitative analysis is also carried out to make sure that the anomalous nodes determined by ADOMS are meaningful. Due to the unavailability of labeled data with ground truths, the top ranked anomalous nodes are investigated manually to verify whether they are actually anomalous. The top ten ranked anomalous nodes discovered by ADOMS and Agg+AD on different datasets are shown in Tables 4.4 and 4.5.

The distribution of anomaly scores determined by ADOMS and Agg+AD on different datasets are depicted in Figures 4.5 and 4.6 respectively. Figures 4.5a, c, e and 4.6a, c, e illustrate the anomaly score distributions of ADOMS, and Figures 4.5b, d, f and 4.6b, d, f illustrate the anomaly score distributions of Agg+AD. From the figures it can be observed that majority of nodes are normal and have anomaly scores near to one. Only very few nodes are anomalous with high anomaly scores and follow the pattern of near-stars or near-cliques in individual layers. Thus, the anomaly scores computed by ADOMS are quite discriminative.

When ADOMS was applied on Noordin Top terrorist multi-layer network, the top three outliers found were nodes 57, 22, and 4. It is observed that node 57 is Noordin Mohamed Top, the head of the terrorist group. He had connections with 41 other terrorists in the Communication layer with 134 interactions among themselves (egonet node count: 42, edge count: 134), 41 terrorists in Operational layer with 285 interactions among themselves (egonet node count: 42, edge count: 285), 3 neighbors in Financial layer with 6 connections among themselves forming a clique (egonet node count: 4, edge count: 6), and 22 neighbors in Trust network with 90 connections among themselves (egonet node count: 23, edge count: 90). Therefore, node 57 is central in the individual network layers. The distribution of anomaly scores determined by ADOMS for the Noordin Top network is shown in Figure 4.5a.

In the aggregated topological network (Agg+AD), the top three anomalies are nodes 57, 22, and 71. The egonet of node 57 contains 54 nodes and 457 edges in the aggregated network. To be more clear, the neighborhood of node 57 has 54 nodes and they interact with each other through 457 edges in the aggregated network. The nodes are connected among each other through any of the four types of interactions. As such, node 57 is the key actor of the terrorist network. Node 71 is a near-clique and is the third ranked anomalous node in Agg+AD, even though it's neighborhood in indi-

Table 4.4: Top ten ranked nodes identified by ADOMS and the baseline method Agg+AD along with the anomaly scores

Dataset	**Rank**	**ADOMS**		**Agg+AD**	
		Node	**Score**	**Node**	**Score**
Noordin Top	1	57	5.2029	57	5.1897
	2	22	4.4601	22	4.505
	3	4	3.6415	71	3.0123
	4	63	2.5531	23	2.6049
	5	20	2.3108	66	2.1823
	6	51	2.1686	44	1.9535
	7	12	1.9003	68	1.8973
	8	44	1.8975	4	1.8787
	9	69	1.8633	51	1.4194
	10	73	1.8508	50	1.356
Social Evolution	1	61	2.7454	47	2.7454
	2	10	2.6721	52	2.6721
	3	13	2.6051	83	2.6051
	4	41	2.1626	63	2.1626
	5	52	2.1164	50	2.1164
	6	50	1.7814	10	1.7814
	7	28	1.7414	61	1.7414
	8	83	1.7058	9	1.7058
	9	47	1.7025	40	1.7025
	10	8	1.647	13	1.647
Aarhus	1	7	2.5493	7	2.4012
	2	21	1.8163	1	1.9929
	3	11	1.741	60	1.9206
	4	23	1.6015	22	1.7571
	5	51	1.5921	34	1.6992
	6	44	1.5726	44	1.6446
	7	29	1.5472	26	1.4655
	8	20	1.4953	53	1.4401
	9	8	1.4144	23	1.4372
	10	57	1.3345	61	1.4163

Table 4.5: Top ten ranked nodes identified by ADOMS and the baseline method Agg+AD along with the anomaly scores

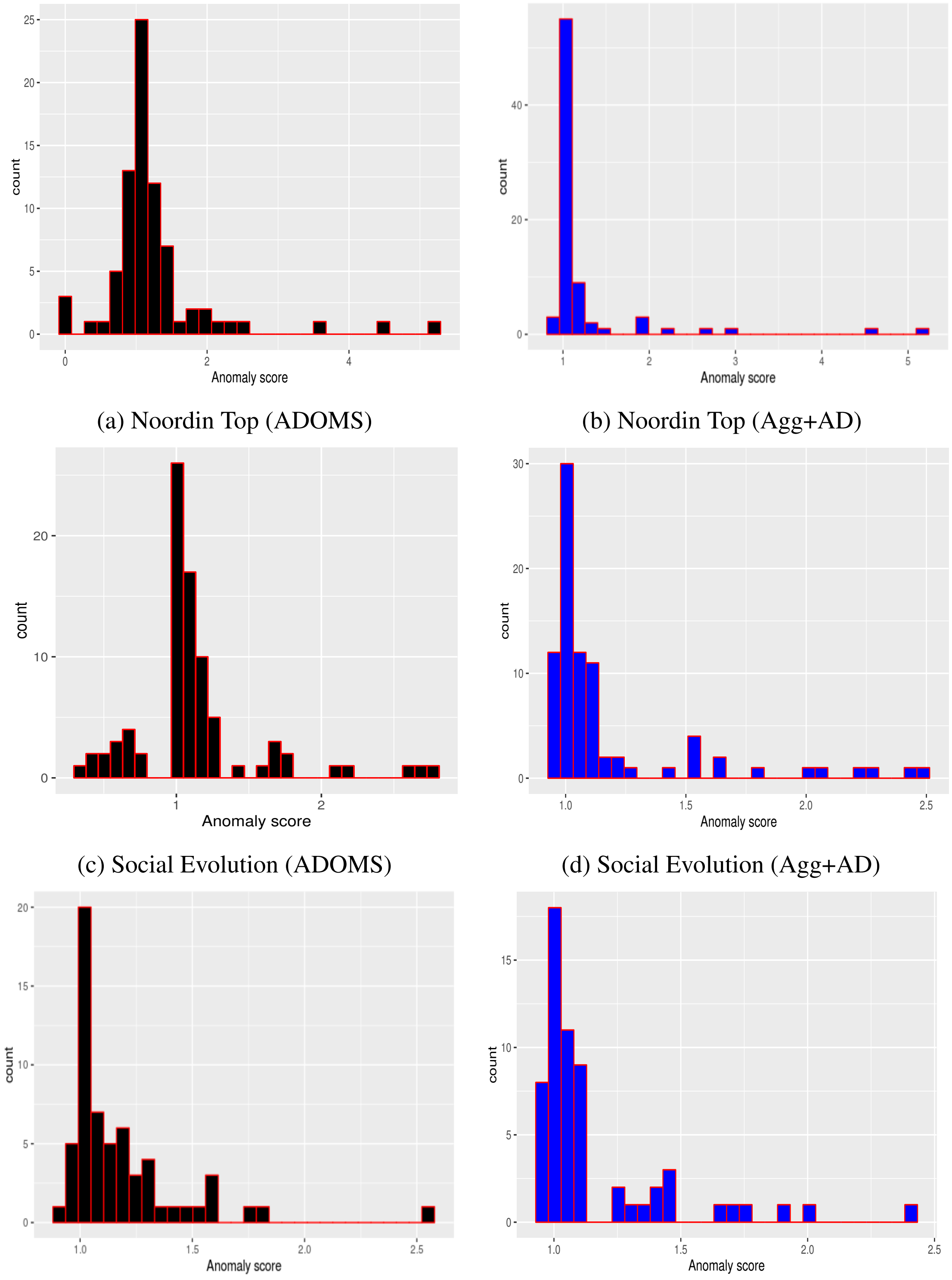
Dataset	Rank	ADOMS		Agg+AD	
		Node	Score	Node	Score
DBLP_C	1	6818	3.6122	558	4.5179
	2	558	3.0036	4181	3.8143
	3	720	2.9881	720	2.813
	4	6177	2.6809	554	2.6648
	5	4181	2.4325	1358	2.1376
	6	7559	2.3525	6818	2.0014
	7	6810	2.3157	14234	1.9106
	8	6178	1.9915	3525	1.8958
	9	3525	1.9875	16216	1.8703
	10	6813	1.8408	233	1.707
arXiv	1	125	4.1105	479	6.4046
	2	127	3.9053	83	3.9465
	3	8156	3.4699	218	3.9279
	4	468	3.2253	172	3.4367
	5	10463	2.5829	54	3.3265
	6	10464	2.5829	715	3.1468
	7	3680	2.3129	578	2.8186
	8	10936	2.3128	80	2.1459
	9	10939	2.3127	1751	2.1403
	10	480	2.2618	842	2.0029
GTD	1	406	9.4574	81	7.0232
	2	253	9.2963	161	4.0266
	3	241	7.7554	134	3.8925
	4	372	7.747	1460	3.8573
	5	1998	7.4901	2245	3.7863
	6	2057	7.4901	109	3.6398
	7	1784	6.876	571	3.5248
	8	952	6.8502	836	3.3788
	9	299	6.6552	203	3.1201
	10	993	6.6467	2495	2.9675

vidual layers are neither close to stars nor cliques. Therefore, node 71 is not detected as a top anomalous node by ADOMS. The distribution of anomaly scores in the aggregate topological network for the Noordin Top network is shown in Figure 4.5b.

In the Social evolution network, the top ranked node discovered by ADOMS is node 61. The student with node index 61 has 2 close friends who are close friends with themselves forming a clique, socialized with 4 other other students twice per week with 10 interactions among themselves forming a clique in that layer, discussed political matters with 3 friends with 6 interactions among themselves forming a clique, tagged photos among 20 friends in Facebook who have 210 among themselves forming a clique, and shared blog/ LiveJournal/ Twitter activities with 16 friends who have 136 interactions among themselves forming a clique. However, in the aggregated network of Social evolution network, the top ranked node is node 47 with 75 nodes and 2137 edges in its egonet. The neighborhood of node 47 in different layers are not close to stars or cliques. The distributions of the anomaly scores by ADOMS and Agg+AD for the social evolution network are shown in Figures. 4.5c and 4.5d respectively.

Similar scenarios were observed in the other datasets also. The difference in anomaly ranking of ADOMS and Agg+AD is due to the fact that Agg+AD considers only whether an interaction exists between two nodes. It can not affirm through which dimensions they interact or the types of interactions. Thus, the expressive power of aggregated network is very less compared to its multi-layer counterpart and it just shows whether an interaction exists between the nodes. In the aggregated network, as the edges are combined from different layers, egonets can be stars or cliques even if they are not so in the individual layers. In other words, in the case of Agg+AD, nodes get high anomalous scores even if they do not follow the pattern of stars or cliques in multiple individual layers. Similarly, in the aggregated network, the anomaly score can be high even if the egonet of at least one layer is near to star or clique. In contrast to Agg+AD, ADOMS considers the multiple interactions existing among the users in the network, and each type of interaction contributes to the total anomaly score of the nodes in the network.

If a node follows the pattern of near-stars or near-cliques in multiple layers, the node is anomalous and the anomaly score of the node in the multi-layer network will be high. More specifically, ADOMS utilizes the representational power of multi-layer networks. As a result, ADOMS is superior in ranking the anomalous nodes in comparison with Agg+AD.



(a) Noordin Top (ADOMS)

(b) Noordin Top (Agg+AD)

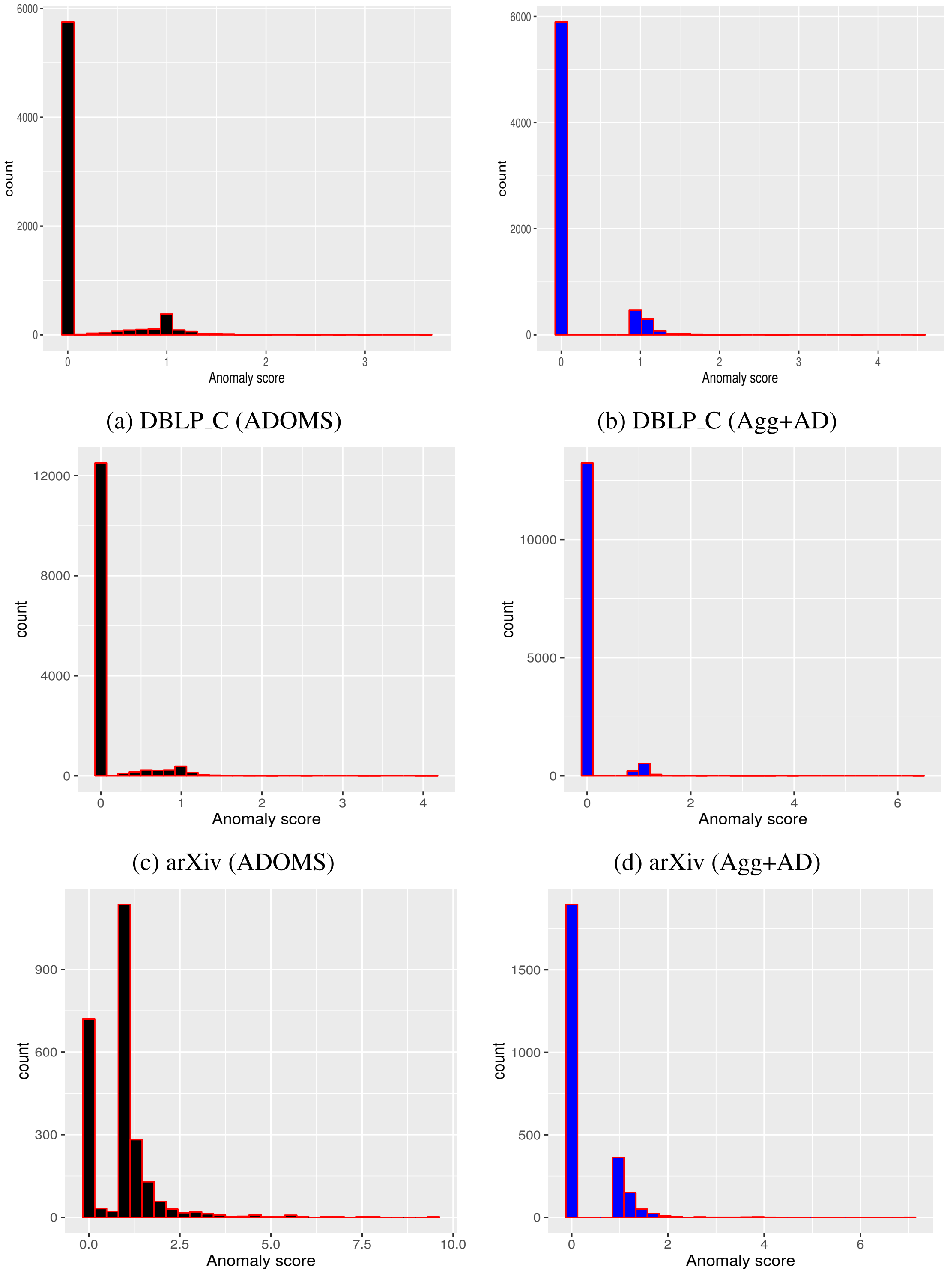
(c) Social Evolution (ADOMS)

(d) Social Evolution (Agg+AD)

(e) Aarhus (ADOMS)

(f) Aarhus (Agg+AD)

Figure 4.5: Anomaly score distributions of ADOMS and Agg+AD on different datasets. Figures 4.5(a), (c), (e) illustrate the anomaly score distributions of ADOMS and Figures 4.5(b), (d), (f) illustrate the anomaly score distributions of Agg+AD respectively.



(a) DBLP_C (ADOMS)

(b) DBLP_C (Agg+AD)

(c) arXiv (ADOMS)

(d) arXiv (Agg+AD)

(e) GTD (ADOMS)

(f) GTD (Agg+AD)

Figure 4.6: Anomaly score distributions of ADOMS and Agg+AD on different datasets. Figures 4.6(a), (c), and (e) illustrate the anomaly score distributions of ADOMS and Figures 4.6(b), (d), and (f) illustrate the anomaly score distributions of Agg+AD respectively.

Table 4.6: Running times of ADOMS and baseline method on different datasets

Sl. No.	Dataset	Running Time (in seconds)	
		ADOMS	Agg+AD
1	Noordin Top	2.74	4.87
2	Social Evolution	39.46	90.68
3	Aarhus	1.50	4.46
4	DBLP_C	290.96	2130.08
5	arXiv	1572.16	8435.53
6	GTD	125.63	638.97

b) Scalability and Running Time

As mentioned earlier, the first phase of ADOMS is implemented in parallel and distributed manner by distributing the feature extraction and anomaly detection operations on individual layers to different cores of the machine. As a result, ADOMS can scale well to large datasets and the overall running time of ADOMS is much less compared to that of the baseline method. The running time of the baseline method Agg+AD is taken as the total time taken for aggregating the layers of the network and applying the anomaly detection on the resulting aggregated topological network. The average running times for ADOMS and the baseline method Agg+AD on different datasets are shown in Table 4.6. It can be observed from the table that the time taken by ADOMS is far less compared to that of Agg+AD.

4.5.5 Advantages of ADOMS Approach

The main advantages of the proposed ADOMS approach are the following:

- **Unsupervised anomaly detection method:** It does not require any labeled dataset for training, as it is often difficult to find labeled data with explicit ground truth.

- **Network-structure based method:** It is solely based on network topology, and does not require any node or edge attributes.

- **Parameter-free method:** It does not require any user-defined parameters. It detects the anomalies automatically without user intervention.

- **Less running time:** It is scalable for large datasets. As ADOMS is implemented in a parallel and distributed fashion, the overall running time is less.

However, ADOMS approach is developed for static multi-layer networks where the structure of the network does not change over time. Future work could generalize ADOMS for dynamic networks. ADOMS identifies anomalous individuals in a multi-layer social network. Moreover, fraudulent individuals can collude with each other to perform collaborative frauds such as auction scam, review spam, etc. Hence, detecting the collaborative groups of anomalous individuals in multi-layer social networks is a major area for future research. Furthermore, in social networks, nodes or edges may be associated with features such as age, interests, educational qualifications, etc. of the individuals, or the duration of the interactions among them. Hence, another promising future research direction is to investigate the significance of attributes associated with the nodes or edges of the multi-layer social networks in improving the anomaly detection process.

4.6 SUMMARY

Anomaly detection is a daunting problem in social network analysis. Even though several techniques and tools have been developed for anomaly detection in single-layer social networks, detecting anomalies in multi-layer social networks is an unexplored area of research. In this chapter, we introduce and study the problem of anomaly detection on multi-layer social networks and consider the two well-known anomalous topologies of star/near and clique/near-clique in multi-layer social networks. In a social network, if the neighborhood of a user is a star/near-star or a clique/near-clique, the online behavior of the user can be linked to a suspicious behavior. An unsupervised, network feature-based, and parameter-free methodology called Anomaly Detection On Multi-layer Social networks, ADOMS, is proposed to automatically rank the nodes of a multi-layer social network based on the degree of similarity of the nodes' neighborhoods in different layers to cliques or stars. An anomaly score is computed for each node in each individual network layer based on the local features of its neighborhood such as the edge count and node count of the egonet. The anomaly scores of the corresponding nodes in individual layers are then combined based on the relevance of the layers to form the anomaly scores of the nodes in the multi-layer social network. The nodes of the network are then ranked based on the anomaly scores. The experiments on multiple real-world multi-layer networks demonstrate that the proposed approach can detect anomalies effectively.

CHAPTER 5

SPAMMER COMMUNITY DETECTION IN MULTI-LAYER SOCIAL NETWORKS

5.1 INTRODUCTION

Spamming is the most prevalent malicious activity in social networks. Spamming involves undesirable users sending tweets consisting of text and HTTP URLs to large number of legitimate users. The motivations for spammers to spread spam messages is with an intention 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.

Twitter is one of the most popular and fastest growing online social networks. 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 URLs leading to unrelated websites. Due to the 140 character limitation in twitter, the users usually share URLs 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.

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 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 spammer 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 majorly form a bunch of fake accounts and follow all of them and other spammers forming a closely knit community. Thus, spammers tend to be socially well-connected with high clustering coefficient (Yang et al. 2012). Essentially, the spammers collaborate with each other and form a close knit community to increase their credibility. The accounts

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

sitting at the center of such communities are generally referred to as spammer 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 chapter, the spammer communities residing in Twitter are detected by analyzing the spammers' 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 multi-layer social network*. The goal is to classify the accounts as spammers and legitimate users, and find social connections between the spammers. To the best of our knowledge, this is the first in-depth effort to detect spammer communities existing in Twitter.

In summary, the major contributions of this 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 this chapter is organized as follows. Section 5.2 presents how the Twitter network is modeled as a multi-layer social network. Section 5.3 deals with the problem statement which includes the formal definition of the problem. In Section 5.4, the solution methodology for the problem is explained. The experimental results are presented in Section 5.5 and the summary of the chapter is presented in the last section.

5.2 TWITTER MULTI-LAYER SOCIAL NETWORK

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 multi-layer 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 multi-layer 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.

5.3 PROBLEM STATEMENT

Given the directed and attributed Twitter multi-layer 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, the aim of the work presented in this chapter 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 this work is defined in Table 5.1.

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.

Table 5.1: Symbols and definitions used in this chapter

Symbols	**Definitions**
M	Twitter multi-layer 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≤i, j≤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

- Local mining: Local mining uses the features that are local to a community to detect spammers.
- Global mining: Global mining uses the features that are globally the same throughout the communities.

An example of possible social network ecosystem is shown in Figure 5.1. 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 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

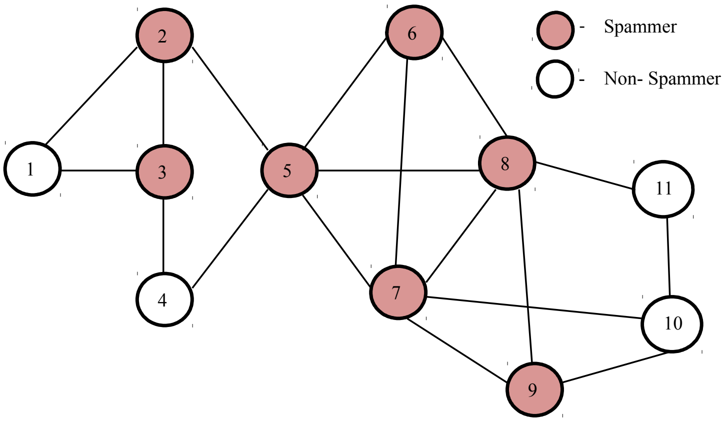


Figure 5.1: Example for a social network

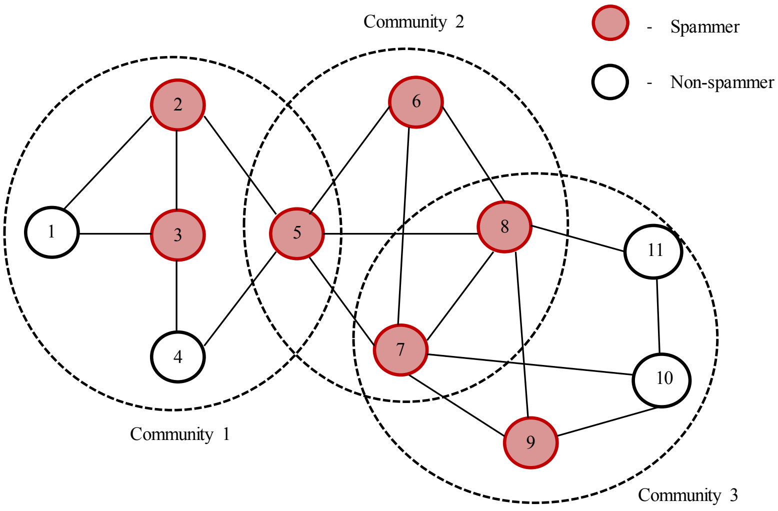


Figure 5.2: Overlapping communities in the network

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 Figure 5.2. 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 (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 (spam 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 spam community that spreads malicious links to users in other communities.

Using the overlapping community structure in Twitter, the aim of this work 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.

5.4 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

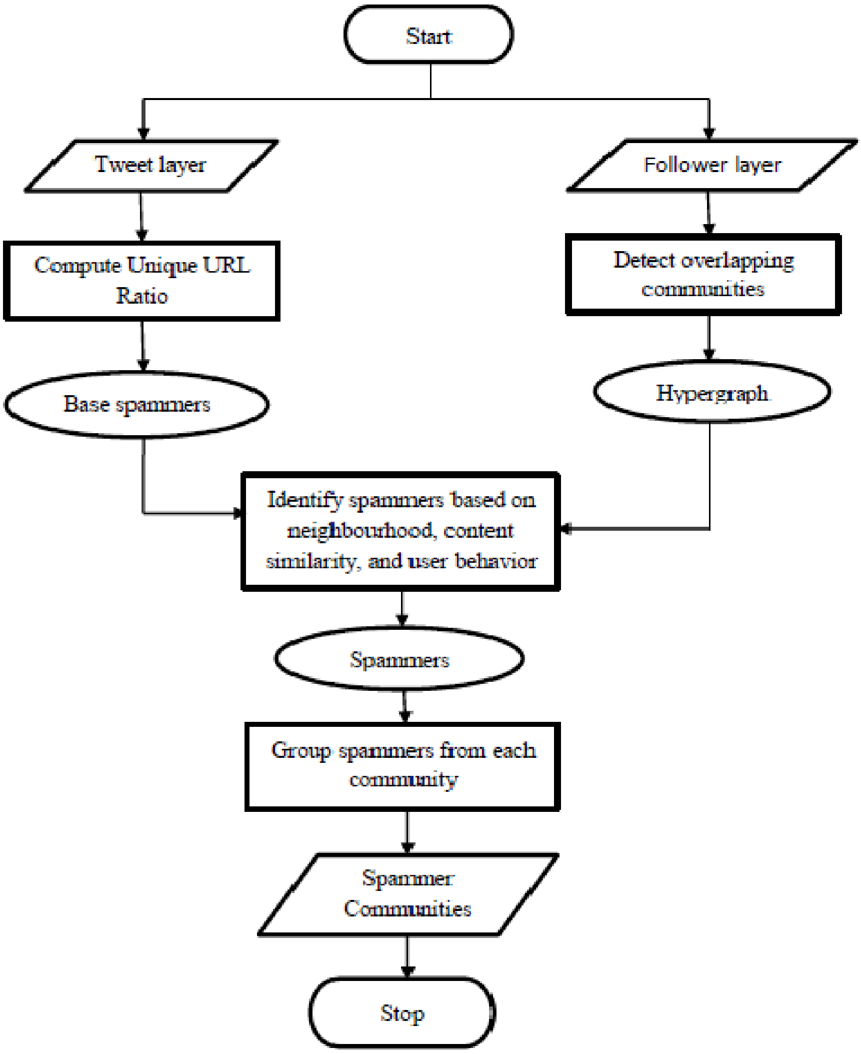

Start
Tweet layer
Follower layer
Compute Unique URL Ratio
Detect overlapping communities
Base spammers
Hypergraph
Identify spammers based on neighbourhood, content similarity, and user behavior
Spammers
Group spammers from each community
Spammer Communities
Stop

Figure 5.3: Flow description of SpamCom

into a well-formed community. Figure 5.3 shows the flowchart of **SpamCom**. The description of each step is given below:

5.4.1 Identifying Base Spammers

As a first step towards detecting 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

Algorithm 5.3 BaseSpammers(G^T)

```
Input: Tweet graph $G^T(V, E^T, U)$
Output: Set of base spammers
$Base\_Spammers \leftarrow \phi$;
for $all\, v \in V\, in\, G^{\mathrm{T}}$ do
    $U \leftarrow Unique\_URL\_Ratio(v)$;
    if $U \leq threshold$ then
        $Base\_Spammers \leftarrow Base\_Spammers \cup \{v\}$;
    end if
end for
return $Base\_Spammers$;
```

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:

$$Unique\_URL\_Ratio(v) = \frac{Number\, of\, unique\, URLs(v)}{Total\, number\, of\, URLs(v)} \tag{5.1}$$

The set of suspect nodes that will be at the base of attack cluster is identified using Algorithm 5.3. 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.

5.4.2 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, such groups in Follower network layer are extracted 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^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.

5.4.3 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 5.4 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. Let *S* be 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 edges 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 as:

$$LC(v,G) = \frac{2.|e^v|}{N_v.(N_v\ -\ 1)} \tag{5.2}$$

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.

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 5.5) by using certain 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 work 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 $\mathrm{U_{base}}$ and $\mathrm{U_{sus}}$be the URLs posted by base spammer and suspect accounts respectively. The Jaccard index for URL similarity is thus defined as:

$$J(U_{base}, U_{sus}) = \frac{|U_{base} \cap U_{sus}|}{|U_{base} \cup U_{sus}|} \tag{5.3}$$

Average Neighbors' Followers: Average Neighbors' Followers (Yang et al. 2013)

Algorithm 5.4 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
$Spam\_accounts \leftarrow \phi$;
for $all\, E^{h} \in H$ do
    $Suspect\_set \leftarrow \phi$;
    for $all\, nodes\, v \in E^{h}$ do
        if $v \in Base\_Spammers$ then
            $S \leftarrow MAX - CLIQUE(v)$;
            $N_S \leftarrow neighborhood\, of\, S$;
            for $all\, nodes\, u \in N_S$ do
                if $LC(u, S) \geq support$ then
                    $S \leftarrow S \cup \{u\}$;
                end if
            end for
            $Suspect\_set \leftarrow Suspect\_set \cup S$;
            $spams \leftarrow Spammers(Suspect\_set, v)$;
            $Spam\_accounts \leftarrow Spam\_accounts \cup spams$
        end if
    end for
end for
return $Spam\_accounts$;
```

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 account's 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*

Algorithm 5.5 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

```
$spammers \leftarrow \phi$;
for $all\, v \in Suspect\_set$ do
    $J \leftarrow J(v, base\_spammer)$;
    $A \leftarrow ANF(v)$;
    $U \leftarrow URL\_Tweet\_Ratio(v)$;
    $age \leftarrow Age\_of\_Account(v)$;
    $spam\_score \leftarrow GetSpamScore(J, A, U, age)$;
    if $spam\_score \geq spam\_threshold$ then
        $spammers \leftarrow spammers \cup \{v\}$;
    end if
end for
return $spammers$;
```

Neighbors' Followers is defined as:

$$ANF(v) = \frac{1}{N_{fer}(v)} . \sum_{u \in N_{fing}(v)} N_{fer}(u) \tag{5.4}$$

URL to Tweet Ratio: Spammers post a large amount of URLs as compared to legitimate users. Based on this impression, the ratio of number of URLs posted by the suspect to the number of tweets posted by suspect is taken. 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:

$$URL\_Tweet\_Ratio(v) = \frac{U_v}{Tweet_v} \tag{5.5}$$

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 the 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:

$$Age\_of\_Account(v) = \frac{t_v - t_{oldest}}{t_{newest} - t_{oldest}} \tag{5.6}$$

Based on 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.

5.4.4 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.

Table 5.2: 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

5.5 EXPERIMENTAL RESULTS AND ANALYSIS

In this section, the experimental results of the proposed approach 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.

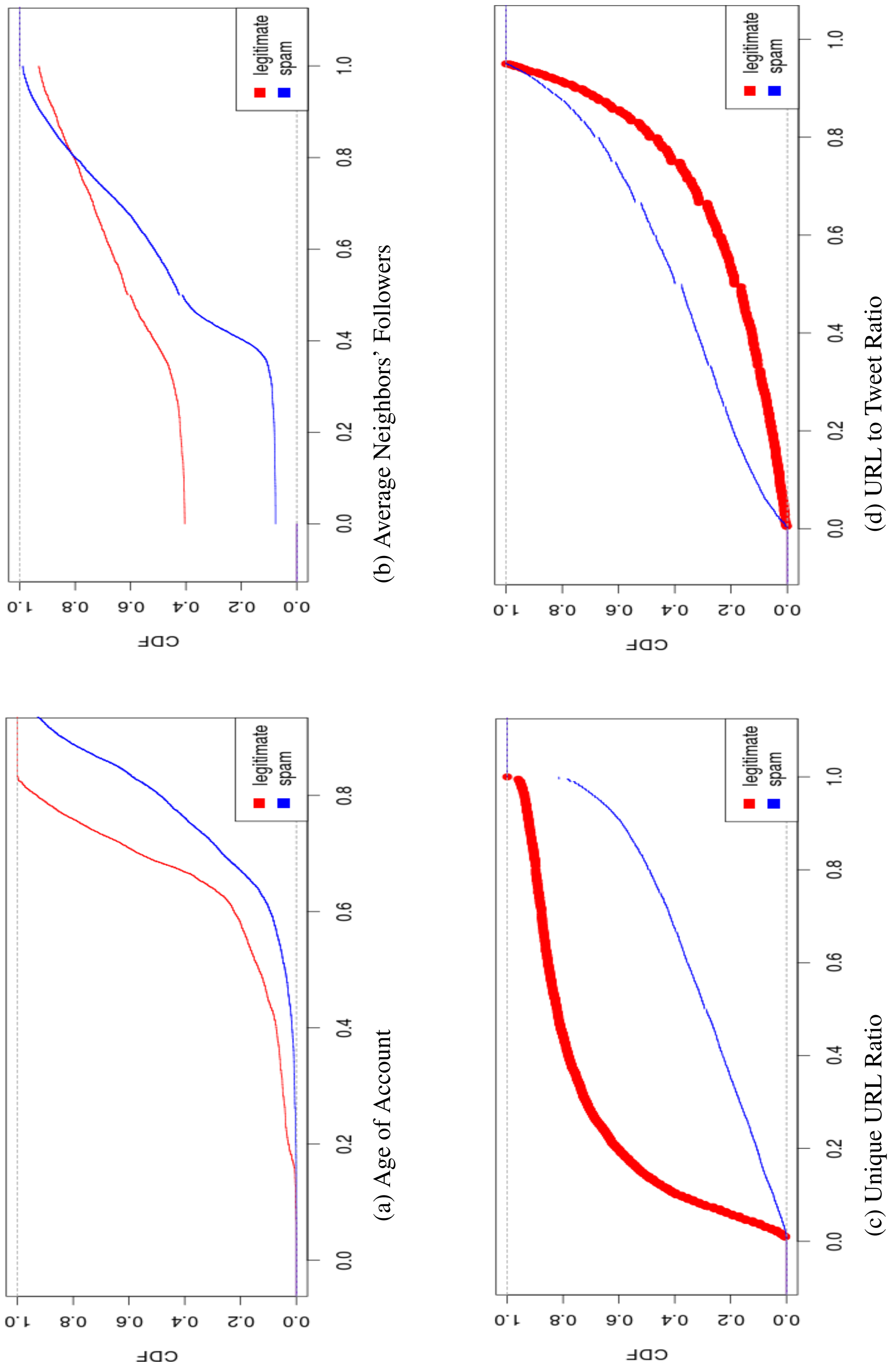
The dataset used to demonstrate the effectiveness of the proposed approach is the Twitter Honeypot dataset (Lee et al. 2011) which is explained in the next subsection. The effectiveness of the features in detecting spammers is studied and the results obtained by the proposed approach when applied to the experimental setup are evaluated. Finally, the characteristics of the communities and relationships among spammers are studied.

5.5.1 Twitter Honeypot Dataset

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 honeypot dataset contains tweets that were captured during the eight month period of 2010. The dataset consists of the tweets posted by users that are 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 5.2. The Twitter multi-layer social network consisting of the *Follower* and *Tweet* layers is constructed from the honeypot dataset.

5.5.2 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 attributes 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*. The CDFs of these attributes are shown in Figs. 5.4. It can be clearly noted from Figure 5.4a 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 5.4b shows that the average number of followers of non-spammers is much higher as compared to spammers as they follow a good quality of accounts usually. Figure 5.4c 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, Figure 5.4d 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 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.



(a) Age of Account

(b) Average Neighbors' Followers

(c) Unique URL Ratio

(d) URL to Tweet Ratio

Figure 5.4: Cumulative Distribution Functions of attributes for Honeypot dataset

Table 5.3: 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

5.5.3 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 of precision and recall obtained for the classifiers are compared with the results obtained from SpamCom in Table 5.3. 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 results that the proposed approach yields better performance using the community-based features and other robust features compared to other machine learning algorithms.

Table 5.4: 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

5.5.4 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 5.4. The table shows the nodes and edges in weakly connected component (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. We identify 18 and 19 communities existing in spammer network for unidirectional and bidirectional links respectively.

Based on the experimental results, it is evident that there are communities of spammers working collectively to spread spam and evade spam 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.

5.6 SUMMARY

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 chapter. After identifying overlapping community structure existing in Twitter, the suspects are identified based on content similarity and connectivity with spam 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.

CHAPTER 6

CONCLUSIONS AND FUTURE SCOPE

Detecting suspicious and illegal behavior is a daunting problem in social network analysis. Anomaly detection is helpful in detecting such behavior. Even though several techniques and tools have been developed for anomaly detection in single-layer social networks, anomaly detection in multi-layer social networks is an unexplored area of research. In this thesis, the problem of anomaly detection on multi-layer social networks is introduced and studied. The two well-known anomalous topologies of star/near and clique/near-clique in multi-layer social networks are considered as anomalies in this work. In a social network, if the neighborhood of a user is a star/near-star or a clique/near-clique, the online behavior of the user can be linked to a suspicious behavior. An unsupervised, network feature-based, and parameter-free methodology called Anomaly Detection On Multi-layer Social networks, ADOMS, is proposed to automatically rank the nodes of a multi-layer social network based on the degree of similarity of the nodes' neighborhoods in different layers to cliques or stars. An anomaly score is computed for each node in each individual network layer based on the local features of its neighborhood such as the edge count and node count of the egonet. The anomaly scores of the corresponding nodes in individual layers are then combined based on the relevance of the layers to form the anomaly scores of the nodes in the multi-layer social network. The nodes of the network are then ranked based on the anomaly scores.

The experiments on multiple real-world multi-layer networks demonstrate that either the analysis of one mode of interaction or the analysis of the aggregated topological network does not provide a complete picture of the relationships among the users of the networks. Thus, multi-layer analysis of the networks is required to identify the anomalies by using the rich information hidden in individual network layers. Even though the proposed approach is applied on multi-layer social networks, it can be applied to any multi-layer network with intra-layer connections. For example, the proposed approach can identify important hub cities in multi-layer transportation networks and important proteins and genes that have critical roles in biological networks.

This work secondly addresses the detection of spammer communities in Twitter. Spamming is the most predominant form of anomalous activity prevalent in online social networks that involves malicious users sending unsolicited messages to legitimate users with the intention of wasting their time, bandwidth, and money. Being one of the fastest growing online social networks, Twitter has become a primary target platform for social spammers. One of the important security issues in Twitter is that the social spammers collaborate with each other and form collective anomalies or spammer communities to spread spam messages to a large set of legitimate users. Therefore, in this work, an unsupervised approach called Spammer Community detection (SpamCom) is developed for detecting spammer communities in Twitter by using graph-theoretic features of the network and the network attributes.

The overlapping community based features existing in the Twitter network, the structural characteristics, URL (content) based characteristics, user behavior, and user account characteristics are employed to detect spammer communities in Twitter. After identifying overlapping community structure existing in Twitter, the suspects are identified based on content similarity and connectivity with spam 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. It is observed that the social spammers tend to be well-connected with high clustering coefficient. The approach is evaluated on real-world dataset, and the experimental results demonstrate significant performance in detecting spammers and spammer communities.

Future Scope

As the approaches proposed in this thesis are pioneering approaches for detecting anomalies in multi-layer networks, there is a significant scope for future research. Further research can be carried out in the following directions:

- ADOMS identifies anomalous individuals in a multi-layer social network. However, fraudulent individuals can collude with each other to perform collaborative frauds such as auction scam, review spam, etc. Hence, detecting the collaborative groups of anomalous individuals in multi-layer social networks is a major area for future research.

- ADOMS is solely based on network topology and does not require node/edge attributes. However, in social networks, nodes or edges may be associated with features such as age, interests, educational qualifications, etc. of the individuals, or the duration of the interactions among them. Hence, another promising future research direction is to investigate the significance of attributes associated with the nodes or edges of the multi-layer social networks in improving the anomalous node detection process.

- SpamCom is an effort to study the spammer community structure existing in social networks. The approach needs evaluation in much finer detail, and a much detailed and extended study of the approach and its performance in real-world scenario are interesting future works.

- The approaches proposed in this thesis are developed for static multi-layer social networks where only one snapshot of the networks is considered for anomaly detection. However, social networks are highly time-evolving, and generalizing the approaches for time-evolving multi-layer social networks is a future work. Here, the challenges are to identify the suitable feature space that characterizes the neighborhood of the nodes, and to extract the features dynamically.

- Exploring and identifying the different types of anomalies (anomalous nodes, edges, and/or subgraphs) that can occur in multi-layer social networks and their real-world applications are promising research directions in this area.

In conclusion, this dissertation proposes unsupervised approaches for detecting anomalies in multi-layer social networks by using graph-theoretic features of the networks and data mining techniques. More specifically, an unsupervised approach is proposed to detect anomalous nodes in a multi-layer social network by analyzing the structure of the network. In addition, an unsupervised approach is proposed to detect anomalous spammer communities in a multi-layer social network by analyzing the structure and attributes of the network.

BIBLIOGRAPHY

Afsarmanesh, N. and Magnani, M. (2016). "Finding overlapping communities in multiplex networks." *arXiv preprint arXiv:1602.03746*.

Aggarwal, C. and Subbian, K. (2014). "Evolutionary network analysis: A survey." *ACM Computing Surveys (CSUR)*, 47(1), 10.

Aggarwal, C. C., Zhao, Y. and Yu, P. S. (2011). "Outlier detection in graph streams." In *2011 IEEE 27th International Conference on Data Engineering (ICDE)*, IEEE, 399–409.

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

Ahmad, M. A., Borbora, Z., Srivastava, J. and Contractor, N. (2010). "Link prediction across multiple social networks." In *2010 IEEE International Conference on Data Mining Workshops*, 911–918.

Akoglu, L., Chandy, R. and Faloutsos, C. (2013). "Opinion fraud detection in online reviews by network effects." In *Proceedings of the 7th International AAAI Conference on Weblogs and Social Media (ICWSM)*, volume 13, The AAAI Press, Massachusetts, USA, 2–11.

Akoglu, L. and Faloutsos, C. (2010). "Event detection in time series of mobile communication graphs." In *Army Science Conference*, 77–79.

Akoglu, L., McGlohon, M. and Faloutsos, C. (2010). "Oddball: Spotting anomalies in weighted graphs." In *Advances in Knowledge Discovery and Data Mining*, Springer, 410–421.

Akoglu, L., Tong, H. and Koutra, D. (2015). "Graph based anomaly detection and description: a survey." *Data Mining and Knowledge Discovery*, 29(3), 626–688.

Al-garadi, M. A., Varathan, K. D. and Ravana, S. D. (2016). "Cybercrime detection in online communications: The experimental case of cyberbullying detection in the twitter network." *Computers in Human Behavior*, 63, 433 – 443.

Arab, M. and Afsharchi, M. (2014). "Community detection in social networks using hybrid merging of sub-communities." *Journal of Network and Computer Applications*, 40, 73 – 84.

Araujo, M., Papadimitriou, S., Gnnemann, S., Faloutsos, C., Basu, P., Swami, A., Papalexakis, E. E. and Koutra, D. (2014). "Com2: Fast automatic discovery of temporal (comet) communities." In *Advances in Knowledge Discovery and Data Mining*, Springer, 271–283.

Asam, A. E. and Samara, M. (2016). "Cyberbullying and the law: A review of psychological and legal challenges." *Computers in Human Behavior*, 65, 127 – 141.

Baingana, B. and Giannakis, G. B. (2016). "Joint community and anomaly tracking in dynamic networks." *IEEE Transactions on Signal Processing*, 64(8), 2013–2025.

Bakshy, E., Rosenn, I., Marlow, C. and Adamic, L. (2012). "The role of social networks in information diffusion." In *Proceedings of the 21st international conference on World Wide Web*, ACM, 519–528.

Barabsi, A.-L., Jeong, H., Nda, Z., Ravasz, E., Schubert, A. and Vicsek, T. (2002). "Evolution of the social network of scientific collaborations." *Physica A: Statistical mechanics and its applications*, 311(3), 590–614.

Barnett, V. and Lewis, T. (1994). *Outliers in statistical data*, John Wiley & Sons.

Battiston, F., Nicosia, V. and Latora, V. (2014). "Structural measures for multiplex networks." *Physical Review E*, 89(3), 032804.

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

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

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

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

Berlingerio, M., Coscia, M. and Giannotti, F. (2011). “Finding and characterizing communities in multidimensional networks.” In *2011 International Conference on Advances in Social Networks Analysis and Mining*, 490–494.

Berlingerio, M., Coscia, M., Giannotti, F., Monreale, A. and Pedreschi, D. (2013a). “Multidimensional networks: foundations of structural analysis.” *World Wide Web*, 16(5-6), 567–593.

Berlingerio, M., Pinelli, F. and Calabrese, F. (2013b). “ABACUS: Frequent pattern mining-based community discovery in multidimensional networks.” *Data Mining and Knowledge Discovery*, 27(3), 294–320.

Bhat, S. Y. and 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*, ACM, 100–107.

Bindu, P. V. and Thilagam, P. S. (2016). “Mining social networks for anomalies: Methods and challenges.” *Journal of Network and Computer Applications*, 68, 213–229.

Bliss, N. T. (2015). *Statistical Signal Processing for Graphs*. PhD thesis, ARIZONA STATE UNIVERSITY.

Boccaletti, S., Bianconi, G., Criado, R., Del Genio, C. I., Gmez-Gardees, J., Romance, M., Sendina-Nadal, I., Wang, Z. and Zanin, M. (2014). “The structure and dynamics of multilayer networks.” *Physics Reports*, 544(1), 1–122.

Boccaletti, S., Latora, V., Moreno, Y., Chavez, M. and Hwang, D. (2006). “Complex networks: Structure and dynamics.” *Physics Reports*, 424(4-5), 175–308.

Breunig, M. M., Kriegel, H.-P., Ng, R. T. and Sander, J. (2000). “Lof: identifying density-based local outliers.” *ACM sigmod record*, 29(2), 93–104.

Bright, D. A., Greenhill, C., Ritter, A. and Morselli, C. (2015). "Networks within networks: using multiple link types to examine network structure and identify key actors in a drug trafficking operation." *Global Crime*, 16(3), 219–237.

Bródka, P. and Kazienko, P. (2014). "Multilayered social networks." *Encyclopedia of Social Network Analysis and Mining*, 998–1013.

Bródka, P., Kazienko, P., Musial, K. and Skibicki, K. (2012). "Analysis of neighbourhoods in multi-layered dynamic social networks." *International Journal of Computational Intelligence Systems*, 5(3), 582–596.

Brodka, P., Stawiak, P. and Kazienko, P. (2011). "Shortest path discovery in the multi-layered social network." In *2011 International Conference on Advances in Social Networks Analysis and Mining*, 497–501.

Carrington, P. J., Scott, J. and Wasserman, S. (2005). *Models and methods in social network analysis*, volume 28, Cambridge University Press.

Chakrabarti, D. "Autopart: Parameter-free graph partitioning and outlier detection." In *Knowledge Discovery in Databases: PKDD 2004*, Springer.

Chakrabarti, D., Zhan, Y. and Faloutsos, C. (2004). "R-MAT: A recursive model for graph mining." In *Proceedings of the 2004 SIAM International Conference on Data Mining*, volume 4, SIAM, Lake Buena Vista, Florida, 442–446.

Chandola, V., Banerjee, A. and Kumar, V. (2009). "Anomaly detection: A survey." *ACM Computing Surveys (CSUR)*, 41(3), 15.

Chaoji, V., Al Hasan, M., Salem, S., Besson, J. and J Zaki, M. (2008). "Origami: A novel and effective approach for mining representative orthogonal graph patterns." *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 1(2), 67–84.

Chau, D. H., Pandit, S. and Faloutsos, C. (2006). "Detecting fraudulent personalities in networks of online auctioneers." In *Knowledge Discovery in Databases: PKDD 2006*, Springer, 103–114.

Chen, C., Lin, C. X., Yan, X. and Han, J. (2008). "On effective presentation of graph patterns: a structural representative approach." In *Proceedings of the 17th ACM conference on Information and knowledge management*, ACM, Napa Valley, CA, USA, 299–308.

Chen, C., Yan, X., Yu, P. S., Han, J., Zhang, D.-Q. and Gu, X. (2007a). "Towards graph containment search and indexing." In *Proceedings of the 33rd international conference on Very large data bases*, VLDB Endowment, 926–937.

Chen, C., Yan, X., Zhu, F. and Han, J. (2007b). "gapprox: Mining frequent approximate patterns from a massive network." In *Seventh IEEE International Conference on Data Mining (ICDM)*, IEEE, 445–450.

Chen, W., Wang, Y. and Yang, S. (2009). "Efficient influence maximization in social networks." In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, 199–208.

Chen, Y. and Malin, B. (2011). "Detection of anomalous insiders in collaborative environments via relational analysis of access logs." In *Proceedings of the first ACM conference on Data and application security and privacy*, ACM, 63–74.

Chen, Y., Nyemba, S. and Malin, B. (2012a). "Auditing medical records accesses via healthcare interaction networks." In *AMIA Annual Symposium Proceedings*, volume 2012, American Medical Informatics Association, 93–102.

Chen, Y., Nyemba, S. and Malin, B. (2012b). "Detecting anomalous insiders in collaborative information systems." *IEEE Transactions on Dependable and Secure Computing*, 9(3), 332–344.

Chen, Y., Nyemba, S., Zhang, W. and Malin, B. (2011). "Leveraging social networks to detect anomalous insider actions in collaborative environments." In *2011 IEEE International Conference on Intelligence and Security Informatics (ISI)*, IEEE, 119–124.

Chen, Y., Nyemba, S., Zhang, W. and Malin, B. (2012c). "Specializing network analysis to detect anomalous insider actions." *Security informatics*, 1(1), 1–24.

Chen, Z., Hendrix, W. and Samatova, N. F. (2012d). "Community-based anomaly detection in evolutionary networks." *Journal of Intelligent Information Systems*, 39(1), 59–85.

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

Chung, F., Lu, L. and Vu, V. (2003). "Spectra of random graphs with given expected degrees." *Proceedings of the National Academy of Sciences*, 100(11), 6313–6318.

Clauset, A. (2005). "Finding local community structure in networks." *Phys. Rev. E*, 72, 026132.

Cook, D. J. and Holder, L. B. (1994). "Substructure discovery using minimum description length and background knowledge." *Journal of Artificial Intelligence Research*, 231–255.

Dang, X. H., Assent, I., Ng, R. T., Zimek, A. and Schubert, E. (2014). "Discriminative features for identifying and interpreting outliers." In *2014 IEEE 30th International Conference on Data Engineering (ICDE)*, IEEE, 88–99.

Das, S., Eğecioğlu, O. and El Abbadi, A. (2010). "Anonymizing weighted social network graphs." In *2010 IEEE 26th International Conference on Data Engineering (ICDE)*, IEEE, 904–907.

Davis, M., Liu, W., Miller, P. and Redpath, G. (2011). "Detecting anomalies in graphs with numeric labels." In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*, CIKM '11, ACM, New York, NY, USA, 1197–1202.

De Domenico, M., Lancichinetti, A., Arenas, A. and Rosvall, M. (2015). "Identifying modular flows on multilayer networks reveals highly overlapping organization in interconnected systems." *Phys. Rev. X*, 5, 011027.

De Domenico, M., Nicosia, V., Arenas, A. and Latora, V. (2015). "Structural reducibility of multilayer networks." *Nature communications*, 6, 6864.

De Domenico, M., Porter, M. A. and Arenas, A. (2014). "MuxViz: a tool for multilayer analysis and visualization of networks." *Journal of Complex Networks*, 159–176.

De Domenico, M., Solé-Ribalta, A., Cozzo, E., Kivelä, M., Moreno, Y., Porter, M. A., Gómez, S. and Arenas, A. (2013). "Mathematical formulation of multilayer networks." *Physical Review X*, 3(4), 041022.

DeBarr, D. and Wechsler, H. (2009). "Spam detection using clustering, random forests, and active learning." In *Sixth Conference on Email and Anti-Spam. Mountain View, California*, Citeseer.

Dehaspe, L., Toivonen, H. and King, R. D. (1998). “Finding frequent substructures in chemical compounds.” In *KDD*, volume 98, 1998.

Deshpande, M., Kuramochi, M., Wale, N. and Karypis, G. (2005). “Frequent substructure-based approaches for classifying chemical compounds.” *IEEE Transactions on Knowledge and Data Engineering*, 17(8), 1036–1050.

Dickison, M., Havlin, S. and Stanley, H. E. (2012). “Epidemics on interconnected networks.” *Physical Review E*, 85(6), 066109.

Diesner, J., Frantz, T. L. and Carley, K. M. (2005). “Communication networks from the Enron email corpus It’s always about the people. Enron is no different.” *Computational & Mathematical Organization Theory*, 11(3), 201–228.

Dong, W., Lepri, B. and Pentland, A. S. (2011). “Modeling the co-evolution of behaviors and social relationships using mobile phone data.” In *Proceedings of the 10th International Conference on Mobile and Ubiquitous Multimedia*, ACM, 134–143.

Dong, X., Frossard, P., Vandergheynst, P. and Nefedov, N. (2012). “Clustering with multi-layer graphs: A spectral perspective.” *IEEE Transactions on Signal Processing*, 60(11), 5820–5831.

Dong, X., Frossard, P., Vandergheynst, P. and Nefedov, N. (2014). “Clustering on multi-layer graphs via subspace analysis on grassmann manifolds.” *IEEE Transactions on Signal Processing*, 62(4), 905–918.

Eberle, W., Graves, J. and Holder, L. (2010). “Insider threat detection using a graph-based approach.” *Journal of Applied Security Research*, 6(1), 32–81.

Eberle, W. and Holder, L. (2007). “Anomaly detection in data represented as graphs.” *Intelligent Data Analysis*, 11(6), 663–689.

Eslami, M., Rabiee, H. R. and Salehi, M. (2011). “DNE: A method for extracting cascaded diffusion networks from social networks.” In *IEEE Conference on Social Computing*, IEEE, 41–48.

Fire, M., Goldschmidt, R. and Elovici, Y. (2014). “Online social networks: threats and solutions.” *IEEE Communications Surveys & Tutorials*, 16(4), 2019–2036.

Fire, M., Katz, G. and Elovici, Y. (2012). “Strangers intrusion detection-detecting spammers and fake profiles in social networks based on topology anomalies.” *Human Journal*, 1(1), 26–39.

Freund, Y. and Mason, L. (1999). “The alternating decision tree learning algorithm.” In *icml*, volume 99, 124–133.

Gao, H., Chen, Y., Lee, K., Palsetia, D. and Choudhary, A. N. (2012). “Towards online spam filtering in social networks.” In *NDSS*.

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

Gao, J., Liang, F., Fan, W., Wang, C., Sun, Y. and Han, J. (2010b). “On community outliers and their efficient detection in information networks.” In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, 813–822.

Ghosh, S., Viswanath, B., Kooti, F., Sharma, N. K., Korlam, G., Benevenuto, F., Ganguly, N. and 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*, ACM, 61–70.

Girvan, M. and Newman, M. E. (2002). “Community structure in social and biological networks.” *Proceedings of the national academy of sciences*, 99(12), 7821–7826.

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

Gupta, M., Gao, J., Aggarwal, C. and Han, J. (2014a). “Outlier detection for temporal data.” *Synthesis Lectures on Data Mining and Knowledge Discovery*, 5(1), 1–129.

Gupta, M., Gao, J., Sun, Y. and Han, J. (2012a). “Community trend outlier detection using soft temporal pattern mining.” In *Machine Learning and Knowledge Discovery in Databases*, Springer, 692–708.

Gupta, M., Gao, J., Sun, Y. and Han, J. (2012b). "Integrating community matching and outlier detection for mining evolutionary community outliers." In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, 859–867.

Gupta, M., Gao, J., Yan, X., Cam, H. and Han, J. (2013). "On detecting association-based clique outliers in heterogeneous information networks." In *2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, IEEE, 108–115.

Gupta, M., Mallya, A., Roy, S., Cho, J. H. and Han, J. (2014b). "Local learning for mining outlier subgraphs from network datasets." In *Proceedings of the 2014 SIAM International Conference on Data Mining*, 73–81.

Halu, A., Mondragón, R. J., Panzarasa, P. and Bianconi, G. (2013). "Multiplex pagerank." *PloS one*, 8(10), e78293.

Hanneman, R. A. and Riddle, M. (2005). *Introduction to social network methods*, University of California Riverside.

Haralabopoulos, G., Anagnostopoulos, I. and Zeadally, S. (2015). "Lifespan and propagation of information in on-line social networks: A case study based on reddit." *Journal of Network and Computer Applications*, 56, 88 – 100.

Harrer, A. and Schmidt, A. (2012). "An approach for the blockmodeling in multi-relational networks." In *2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, IEEE, 591–598.

Hassanzadeh, R. and Nayak, R. (2013a). "A rule-based hybrid method for anomaly detection in online-social-network graphs." In *2013 IEEE 25th International Conference on Tools with Artificial Intelligence (ICTAI)*, IEEE, 351–357.

Hassanzadeh, R. and Nayak, R. (2013b). "A semi-supervised graph-based algorithm for detecting outliers in online-social-networks." In *Proceedings of the 28th Annual ACM Symposium on Applied Computing*, ACM, 577–582.

Hassanzadeh, R., Nayak, R. and Stebila, D. (2012). "Analyzing the Effectiveness of Graph Metrics for Anomaly Detection in Online Social Networks." In *Web Informa-*

tion Systems Engineering - WISE 2012, volume 7651, Springer Berlin Heidelberg, Berlin, Heidelberg, 624–630.

Hawkins, D. M. (1980). *Identification of outliers*, volume 11, Springer.

Haythornthwaite, C. (2005). "Social networks and internet connectivity effects." *Information, Communication & Society*, 8(2), 125–147.

Heard, N. A., Weston, D. J., Platanioti, K., Hand, D. J. and others (2010). "Bayesian anomaly detection methods for social networks." *The Annals of Applied Statistics*, 4(2), 645–662.

Henderson, K., Gallagher, B., Li, L., Akoglu, L., Eliassi-Rad, T., Tong, H. and Faloutsos, C. (2011). "It's who you know: graph mining using recursive structural features." In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, 663–671.

Hmimida, M. and Kanawati, R. (2015). "Community detection in multiplex networks: A seed-centric approach." *Networks & Heterogeneous Media*, 10(1), 71–85.

Hodge, V. J. and Austin, J. (2004). "A survey of outlier detection methodologies." *Artificial Intelligence Review*, 22(2), 85–126.

Holder, L. B., Cook, D. J., Djoko, S. and others (1994). "Substucture discovery in the SUBDUE system." In *KDD workshop*, 169–180.

Hu, X., Tang, J., Zhang, Y. and Liu, H. (2013). "Social spammer detection in microblogging." In *IJCAI*, volume 13, Citeseer, 2633–2639.

Huang, Z. and Zeng, D. D. (2006). "A link prediction approach to anomalous email detection." In *IEEE International Conference on Systems, Man and Cybernetics, SMC '06*, volume 2, IEEE, 1131–1136.

Jalili, M., Orouskhani, Y., Asgari, M., Alipourfard, N. and Perc, M. (2017). "Link prediction in multiplex online social networks." *Royal Society Open Science*, 4(2).

Jeon, I., Papalexakis, E. E., Kang, U. and Faloutsos, C. (2015). "Haten2: Billion-scale tensor decompositions." In *2015 IEEE 31st International Conference on Data Engineering (ICDE)*, IEEE, 1047–1058.

Jeub, L. G., Mahoney, M. W., Mucha, P. J. and Porter, M. A. (2015). "A local perspective on community structure in multilayer networks." *arXiv preprint arXiv:1510.05185*.

Ji, T., Yang, D. and Gao, J. (2013). "Incremental local evolutionary outlier detection for dynamic social networks." In *Proceedings, Part II, of the European Conference on Machine Learning and Knowledge Discovery in Databases - Volume 8189*, ECML PKDD 2013, Springer-Verlag New York, Inc., New York, NY, USA, 1–15.

Jiang, C. (2011). *Frequent subgraph mining algorithms on weighted graphs*. PhD thesis, University of Liverpool.

Jiang, M., Cui, P. and Faloutsos, C. (2016). "Suspicious behavior detection: Current trends and future directions." *IEEE Intelligent Systems*, 31(1), 31–39.

Jin, N., Young, C. and Wang, W. (2009). "Graph classification based on pattern co-occurrence." In *Proceedings of the 18th ACM conference on Information and knowledge management*, ACM, 573–582.

Jindal, N., Liu, B. and Lim, E.-P. (2010). "Finding unusual review patterns using unexpected rules." In *Proceedings of the 19th ACM international conference on Information and knowledge management*, ACM, 1549–1552.

John, G. H. and Langley, P. (1995). "Estimating continuous distributions in bayesian classifiers." In *Proceedings of the Eleventh conference on Uncertainty in artificial intelligence*, Morgan Kaufmann Publishers Inc., 338–345.

Kao, T.-C. and Porter, M. A. (2017). "Layer communities in multiplex networks." *arXiv preprint arXiv:1706.04147*.

Kaur, R. and Singh, S. (2017). "A comparative analysis of structural graph metrics to identify anomalies in online social networks." *Computers & Electrical Engineering*, 57, 294 – 310.

Kelley, B. P., Sharan, R., Karp, R. M., Sittler, T., Root, D. E., Stockwell, B. R. and Ideker, T. (2003). "Conserved pathways within bacteria and yeast as revealed by global protein network alignment." *Proceedings of the National Academy of Sciences*, 100(20), 11394–11399.

Keyvanpour, M., Moradi, M. and Hasanzadeh, F. (2014). "Digital Forensics 2.0." In *Computational Intelligence in Digital Forensics: Forensic Investigation and Applications*, Springer, 17–46.

Kivelä, M., Arenas, A., Barthelemy, M., Gleeson, J. P., Moreno, Y. and Porter, M. A. (2014). "Multilayer networks." *Journal of Complex Networks*, 2(3), 203–271.

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

Kolda, T. G. and Bader, B. W. (2009). "Tensor decompositions and applications." *SIAM Review*, 51(3), 455–500.

Kolda, T. G. and Sun, J. (2008). "Scalable tensor decompositions for multi-aspect data mining." In *Eighth IEEE International Conference on Data Mining*, IEEE, 363–372.

Koutra, D., Papalexakis, E. E. and Faloutsos, C. (2012). "Tensorsplat: Spotting latent anomalies in time." In *2012 16th Panhellenic Conference on Informatics (PCI)*, IEEE, 144–149.

Kulis, B., Basu, S., Dhillon, I. and Mooney, R. (2009). "Semi-supervised graph clustering: a kernel approach." *Machine learning*, 74(1), 1–22.

Kuramochi, M. and Karypis, G. (2001). "Frequent subgraph discovery." In *Proceedings IEEE International Conference on Data Mining (ICDM)*, IEEE, 313–320.

Kuramochi, M. and Karypis, G. (2005). "Finding frequent patterns in a large sparse graph." *Data mining and knowledge discovery*, 11(3), 243–271.

Lee, K., Caverlee, J. and 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*, ACM, 435–442.

Lee, K., Eoff, B. D. and Caverlee, J. (2011). "Seven months with the devils: A long-term study of content polluters on twitter." In *Proceedings of Fifth International AAAI Conference on Weblogs and Social Media (ICWSM)*.

Lee, K.-M., Min, B. and Goh, K.-I. (2015). "Towards real-world complexity: an introduction to multiplex networks." *The European Physical Journal B*, 88(2).

Leskovec, J. and Horvitz, E. (2008). "Planetary-scale views on a large instant-messaging network." In *Proceedings of the 17th international conference on World Wide Web*, ACM, 915–924.

Leskovec, J., Huttenlocher, D. and Kleinberg, J. (2010). "Signed networks in social media." In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM, 1361–1370.

Li, Z., Xiong, H. and Liu, Y. (2012). "Mining blackhole and volcano patterns in directed graphs: a general approach." *Data Mining and Knowledge Discovery*, 25(3), 577–602.

Li, Z., Xiong, H., Liu, Y. and Zhou, A. (2010). "Detecting blackhole and volcano patterns in directed networks." In *2010 IEEE 10th International Conference on Data Mining (ICDM)*, IEEE, 294–303.

Liben-Nowell, D. and Kleinberg, J. (2007). "The link-prediction problem for social networks." *Journal of the American society for information science and technology*, 58(7), 1019–1031.

Liu, Y. and Chawla, S. (2015). "Social media anomaly detection: Challenges and solutions." In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, 2317–2318.

Luarn, P., Yang, J.-C. and Chiu, Y.-P. (2014). "The network effect on information dissemination on social network sites." *Computers in Human Behavior*, 37, 1–8.

Madan, A., Cebrian, M., Moturu, S., Farrahi, K. and Pentland, A. . (2012). "Sensing the" health state" of a community." *IEEE Pervasive Computing*, 11(4), 36–45.

Magnani, M., Micenkova, B. and Rossi, L. (2013). "Combinatorial analysis of multiple networks." *arXiv preprint*, 17.

Magnani, M. and Rossi, L. (2011). "The ml-model for multi-layer social networks." In *2011 International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, IEEE, 5–12.

Martinez-Romo, J. and Araujo, L. (2013). "Detecting malicious tweets in trending topics using a statistical analysis of language." *Expert Systems with Applications*, 40(8), 2992–3000.

McPherson, J. M., Popielarz, P. A. and Drobnic, S. (1992). “Social networks and organizational dynamics.” *American Sociological Review*, 57(2), 153–170.

Miller, B., Beard, M., Wolfe, P. and Bliss, N. (2015). “A spectral framework for anomalous subgraph detection.” *IEEE Transactions on Signal Processing*, 63(16), 4191–4206.

Miller, B., Beard, M. S., Bliss, N. T. and others (2011). “Eigenspace analysis for threat detection in social networks.” In *2011 Proceedings of the 14th International Conference on Information Fusion (FUSION)*, IEEE, 1–7.

Miller, B., Bliss, N. and Wolfe, P. J. (2010). “Subgraph detection using eigenvector L1 norms.” In *Advances in Neural Information Processing Systems*, 1633–1641.

Miller, B. A., Arcolano, N., Beard, M. S., Kepner, J., Schmidt, M. C., Bliss, N. T. and Wolfe, P. J. (2012). “A scalable signal processing architecture for massive graph analysis.” In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 5329–5332.

Miller, B. A., Arcolano, N. and Bliss, N. T. (2013). “Efficient anomaly detection in dynamic, attributed graphs: Emerging phenomena and big data.” In *2013 IEEE International Conference on Intelligence and Security Informatics (ISI)*, IEEE, 179–184.

Mislove, A., Marcon, M., Gummadi, K. P., Druschel, P. and Bhattacharjee, B. (2007). “Measurement and analysis of online social networks.” In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, ACM, 29–42.

Mongiovi, M., Bogdanov, P., Ranca, R., Papalexakis, E. E., Faloutsos, C. and Singh, A. K. (2013a). “Netspot: Spotting significant anomalous regions on dynamic networks.” In *SIAM International Conference on Data Mining*, SIAM.

Mongiovi, M., Bogdanov, P. and Singh, A. K. (2013b). “Mining evolving network processes.” In *2013 IEEE 13th International Conference on Data Mining (ICDM)*, IEEE, 537–546.

Mucha, P. J., Richardson, T., Macon, K., Porter, M. A. and Onnela, J.-P. (2010). “Community structure in time-dependent, multiscale, and multiplex networks.” *Science*, 328(5980), 876–878.

Muller, E., Snchez, P. I., Mulle, Y. and Bohm, K. (2013). “Ranking outlier nodes in subspaces of attributed graphs.” In *IEEE 29th International Conference on Data Mining (ICDM)*, IEEE, 216–222.

Mustafaraj, E. and Metaxas, P. T. (2010). “From obscurity to prominence in minutes: Political speech and real-time search.” .

Nanavati, A. A., Gurumurthy, S., Das, G., Chakraborty, D., Dasgupta, K., Mukherjea, S. and Joshi, A. (2006). “On the structural properties of massive telecom call graphs: findings and implications.” In *Proceedings of the 15th ACM international conference on Information and knowledge management*, ACM, 435–444.

Newman, M. E. (2004). “Analysis of weighted networks.” *Physical Review E*, 70(5), 056131.

Newman, M. E. J. (2012). “Communities, modules and large-scale structure in networks.” *Nature Physics*, 8(1), 25–31.

Noble, C. C. and Cook, D. J. (2003). “Graph-based anomaly detection.” In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, 631–636.

Pandhre, S., Gupta, M. and Balasubramanian, V. N. (2016). “Community-based outlier detection for edge-attributed graphs.” *CoRR*, abs/1612.09435.

Pandit, S., Chau, D. H., Wang, S. and Faloutsos, C. (2007). “Netprobe: a fast and scalable system for fraud detection in online auction networks.” In *Proceedings of the 16th international conference on World Wide Web*, ACM, 201–210.

Papalexakis, E. E., Faloutsos, C. and Sidiropoulos, N. D. (2012). “Parcube: Sparse parallelizable tensor decompositions.” In *Machine Learning and Knowledge Discovery in Databases*, Springer, 521–536.

Pedrycz, W. and Chen, S.-M. (2013). *Social Networks: A Framework of Computational Intelligence*, Springer.

Priebe, C. E., Conroy, J. M., Marchette, D. J. and Park, Y. (2005). “Scan statistics on enron graphs.” *Computational & Mathematical Organization Theory*, 11(3), 229–247.

Pujari, M. and Kanawati, R. (2015). “Link prediction in multiplex networks.” *Networks & Heterogeneous Media*, 10(1), 17–35.

Rahman, M. S., Huang, T.-K., Madhyastha, H. V. and Faloutsos, M. (2012). “Efficient and scalable socware detection in online social networks.” In *USENIX Security Symposium*, 663–678.

Ramezanian, R., Salehi, M., Magnani, M. and Montesi, D. (2015). “Diffusion of innovations over multiplex social networks.” In *International Symposium on Artificial Intelligence and Signal Processing (AISP)*, 1–5.

Ratkiewicz, J., Conover, M., Meiss, M., Gonçalves, B., Flammini, A. and Menczer, F. (2011a). “Detecting and tracking political abuse in social media.” In *Proceedings of Fifth International AAAI Conference on Weblogs and Social Media*, 297–304.

Ratkiewicz, J., Conover, M., Meiss, M., Gonçalves, B., Patil, S., Flammini, A. and Menczer, F. (2011b). “Truthy: mapping the spread of astroturf in microblog streams.” In *Proceedings of the 20th international conference companion on World wide web (ICWSM)*, ACM, 249–252.

Redondo, D., Sallaberry, A., Ienco, D., Zaidi, F. and Poncelet, P. (2015). “Layer-centered approach for multigraphs visualization.” In *International Conference on Information Visualisation (iV)*, 50–55.

Renoust, B., Melançon, G. and Munzner, T. (2015). “Detangler: Visual analytics for multiplex networks.” *Eurographics Conference on Visualization (EuroVis)*, 34(3).

Rissanen, J. (1999). “Hypothesis selection and testing by the MDL principle.” *The Computer Journal*, 42(4), 260–269.

Roberts, N. and Everton, S. F. (2011). “Roberts and everton terrorist data: Noordin top terrorist network (subset).”).

Robins, G., Pattison, P. and Wang, P. (2009). “Closure, connectivity and degree distributions: Exponential random graph (p*) models for directed social networks.” *Social Networks*, 31(2), 105–117.

Rodriguez, M. A. and Shinavier, J. (2010). “Exposing multi-relational networks to single-relational network analysis algorithms.” *Journal of Informetrics*, 4(1), 29 – 41.

Rossetti, G., Berlingerio, M. and Giannotti, F. (2011). “Scalable link prediction on multidimensional networks.” In *International Conference on Data Mining Workshops*, IEEE, 979–986.

Rossi, L. and Magnani, M. (2015). “Towards effective visual analytics on multiplex and multilayer networks.” *Chaos, Solitons and Fractals*, 72, 68–76.

Rossi, R. A., Gallagher, B., Neville, J. and Henderson, K. (2013). “Modeling dynamic behavior in large evolving graphs.” In *Proceedings of the sixth ACM international conference on Web search and data mining*, ACM, 667–676.

Sael, L., Jeon, I. and Kang, U. (2015). “Scalable tensor mining.” *Big Data Research*, 2(2), 82–86.

Saigo, H., Nowozin, S., Kadowaki, T., Kudo, T. and Tsuda, K. (2009). “gBoost: a mathematical programming approach to graph classification and regression.” *Machine Learning*, 75(1), 69–89.

Salehi, M., Sharma, R., Marzolla, M., Magnani, M., Siyari, P. and Montesi, D. (2015). “Spreading processes in multilayer networks.” *IEEE Transactions on Network Science and Engineering*, 2(2), 65–83.

Savage, D., Zhang, X., Yu, X., Chou, P. and Wang, Q. (2014). “Anomaly detection in online social networks.” *Social Networks*, 39, 62–70.

Scott, J. (2011). “Social network analysis: developments, advances, and prospects.” *Social network analysis and mining*, 1(1), 21–26.

Shah, N., Beutel, A., Hooi, B., Akoglu, L., Gunnemann, S., Makhija, D., Kumar, M. and Faloutsos, C. (2016). “Edgecentric: Anomaly detection in edge-attributed networks.” In *2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW)*, IEEE, 327–334.

Sharpnack, J. L., Krishnamurthy, A. and Singh, A. (2013). “Near-optimal anomaly detection in graphs using lovsz extended scan statistic.” In *Advances in Neural Information Processing Systems*, 1959–1967.

Shetty, J. and Adibi, J. (2005). “Discovering important nodes through graph entropy the case of enron email database.” In *Proceedings of the 3rd international workshop on Link discovery*, ACM, 74–81.

Shiga, M. and Mamitsuka, H. (2012). “A variational bayesian framework for clustering with multiple graphs.” *IEEE Transactions on Knowledge and Data Engineering*, 24(4), 577–590.

Shrivastava, N., Majumder, A. and Rastogi, R. (2008). “Mining (social) network graphs to detect random link attacks.” In *IEEE 24th International Conference on Data Engineering (ICDE)*, IEEE, 486–495.

Song, J., Lee, S. and Kim, J. (2011). “Spam filtering in twitter using sender-receiver relationship.” In *Recent Advances in Intrusion Detection*, Springer, 301–317.

Sricharan, K. and Das, K. (2014). “Localizing anomalous changes in time-evolving graphs.” In *Proceedings of the 2014 ACM SIGMOD international conference on Management of data*, ACM, 1347–1358.

Stringhini, G., Kruegel, C. and Vigna, G. (2010). “Detecting spammers on social networks.” In *Proceedings of the 26th Annual Computer Security Applications Conference*, ACM, 1–9.

Sun, H., Huang, J., Han, J., Deng, H., Zhao, P. and Feng, B. (2010). “gskeletonclu: Density-based network clustering via structure-connected tree division or agglomeration.” In *2010 IEEE 10th International Conference on Data Mining (ICDM)*, IEEE, 481–490.

Sun, J., Qu, H., Chakrabarti, D. and Faloutsos, C. (2005). “Neighborhood formation and anomaly detection in bipartite graphs.” In *Fifth IEEE International Conference on Data Mining (ICDM)*, IEEE, 8–pp.

Sun, J., Tao, D. and Faloutsos, C. (2006). “Beyond streams and graphs: dynamic tensor analysis.” In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, 374–383.

Sun, J., Xie, Y., Zhang, H. and Faloutsos, C. (2007). “Less is more: Compact matrix decomposition for large sparse graphs.” In *In Proc. SIAM Intl. Conf. Data Mining*.

Tang, L., Wang, X. and Liu, H. (2012). “Community detection via heterogeneous interaction analysis.” *Data Mining and Knowledge Discovery*, 25(1), 1–33.

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

Thompson, B. and Eliassi-Rad, T. (2009). "Dapa-v10: Discovery and analysis of patterns and anomalies in volatile time-evolving networks." In *Notes of the 1st Workshop on Information in Networks (WIN)*.

Tong, H. and Lin, C.-Y. (2011). "Non-negative residual matrix factorization with application to graph anomaly detection." In *SDM*, SIAM, 143–153.

Tong, H., Papadimitriou, S., Sun, J., Yu, P. S. and Faloutsos, C. (2008). "Colibri: fast mining of large static and dynamic graphs." In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, 686–694.

Tsang, S., Koh, Y. S., Dobbie, G. and Alam, S. (2014). "Span: Finding collaborative frauds in online auctions." *Knowledge-Based Systems*, 71, 389 – 408.

Tucker, L. R. (1966). "Some mathematical notes on three-mode factor analysis." *Psychometrika*, 31(3), 279–311.

Wang, A. H. (2010). "Don't follow me: Spam detection in twitter." In *Proceedings of the 2010 International Conference on Security and Cryptography (SECRYPT)*, IEEE, 1–10.

Wang, G., Xie, S., Liu, B. and Yu, P. S. (2011). "Review graph based online store review spammer detection." In *2011 IEEE 11th international conference on Data Mining (ICDM)*, IEEE, 1242–1247.

Wang, G., Xie, S., Liu, B. and Yu, P. S. (2012a). "Identify online store review spammers via social review graph." *ACM Transactions on Intelligent Systems and Technology (TIST)*, 3(4), 61.

Wang, X., Ding, X., Tung, A., Ying, S. and Jin, H. (2012b). "An efficient graph indexing method." In *2012 IEEE 28th International Conference on Data Engineering (ICDE)*, IEEE, 210–221.

Wang, Y., Chakrabarti, A., Sivakoff, D. and Parthasarathy, S. (2017). "Fast change point detection on dynamic social networks." *CoRR*, abs/1705.07325.

Wang, Y., Yin, G., Cai, Z., Dong, Y. and Dong, H. (2015). “A trust-based probabilistic recommendation model for social networks.” *Journal of Network and Computer Applications*, 55, 59 – 67.

Wasserman, S. and Faust, K. (1994). *Social network analysis: Methods and applications*, volume 8, Cambridge university press.

Xin, D., Cheng, H., Yan, X. and Han, J. (2006). “Extracting redundancy-aware top-k patterns.” In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, 444–453.

Xu, X., Yuruk, N., Feng, Z. and Schweiger, T. A. (2007). “Scan: a structural clustering algorithm for networks.” In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, 824–833.

Xu, Z., Ke, Y., Wang, Y., Cheng, H. and Cheng, J. (2012). “A model-based approach to attributed graph clustering.” In *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data*, ACM, 505–516.

Yan, X., Yu, P. S. and Han, J. (2005). “Substructure similarity search in graph databases.” In *Proceedings of the 2005 ACM SIGMOD international conference on Management of data*, ACM, 766–777.

Yang, C., Harkreader, R. and Gu, G. (2013). “Empirical evaluation and new design for fighting evolving twitter spammers.” *IEEE Transactions on Information Forensics and Security*, 8(8), 1280–1293.

Yang, C., Harkreader, R., Zhang, J., Shin, S. and 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*, ACM, 71–80.

Yang, C., Zhang, J. and Gu, G. (2014). “A taste of tweets: reverse engineering twitter spammers.” In *Proceedings of the 30th Annual Computer Security Applications Conference*, ACM, 86–95.

Yang, W., Shen, G.-W., Wang, W., Gong, L.-Y., Yu, M. and Dong, G.-Z. (2015). “Anomaly detection in microblogging via co-clustering.” *Journal of Computer Science and Technology*, 30(5), 1097–1108.

Yardi, S., Romero, D., Schoenebeck, G. et al. (2009). "Detecting spam in a twitter network." *First Monday*, 15(1).

Yu, R., He, X. and Liu, Y. (2015). "Glad: group anomaly detection in social media analysis." *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 10(2), 18.

Yu, R., Qiu, H., Wen, Z., Lin, C. and Liu, Y. (2016). "A survey on social media anomaly detection." *ACM SIGKDD Explorations Newsletter*, 18(1), 1–14.

Yu, W., Aggarwal, C. C., Ma, S. and Wang, H. (2013). "On anomalous hotspot discovery in graph streams." In *2013 IEEE 13th International Conference on Data Mining (ICDM)*, IEEE, 1271–1276.

Yuan, D., Mitra, P., Yu, H. and Giles, C. L. (2012). "Iterative graph feature mining for graph indexing." In *2012 IEEE 28th International Conference on Data Engineering (ICDE)*, IEEE, 198–209.

Zheng, X., Zeng, Z., Chen, Z., Yu, Y. and Rong, C. (2015). "Detecting spammers on social networks." *Neurocomputing*, 159, 27–34.

Zheng, X., Zhang, X., Yu, Y., Kechadi, T. and Rong, C. (2016). "Elm-based spammer detection in social networks." *The Journal of Supercomputing*, 72(8), 2991–3005.

PUBLICATIONS BASED ON THE RESEARCH WORK

1. **Bindu, P. V.** and Thilagam, P. S. (2016). "Mining Social Networks for Anomalies: Methods and Challenges." *Journal of Network and Computer Applications*, Elsevier, 68:213 - 229.
   URL: `http://doi.org/10.1016/j.jnca.2016.02.021`

2. **Bindu, P. V.**, Thilagam, P. S., and Ahuja, D. (2017). "Discovering Suspicious Behavior in Multilayer Social Networks." *Computers in Human Behavior*, Elsevier, 73:568 - 582.
   URL: `http://doi.org/10.1016/j.chb.2017.04.001`

3. **Bindu, P. V.**, Mishra, R., and Thilagam, P. S. (2018). "Discovering Spammer Communities in Twitter." *Journal of Intelligent Information Systems*, Springer.
   URL: `https://doi.org/10.1007/s10844-017-0494-z`