

Anomaly Detection in Complex Networks

A dissertation submitted to
Jawaharlal Nehru University, New Delhi
in partial fulfilment for award of the degree of

Master of Technology
in
Computer Science & Technology

by
Trishita Mukherjee
(20/10/MT/056)

under the supervision of
Professor Rajeev Kumar



School of Computer and Systems Sciences
Jawaharlal Nehru University New Delhi
August 2022

To my parents...




School of Computer and Systems Sciences
Jawaharlal Nehru University
New Delhi 110067, India

Certificate

This is to certify that the dissertation entitled “**Anomaly Detection in Complex Networks**” submitted by **Trishita Mukherjee** (Enrollment No. 20/10/MT/056) to **Jawaharlal Nehru University New Delhi** towards partial fulfilment of requirements for the award of degree of Master of Technology in Computer Science and Technology is a record of bona fide work carried out by her under my supervision and guidance during Winter Semester, 2021-2022.

(Dean)
School of Computer and Systems Sciences
Jawaharlal Nehru University
New Delhi


Aug. 31, 2022 (Supervisor)
School of Computer and Systems Sciences
Jawaharlal nehru University
New Delhi

Acknowledgments

I would like to sincerely express my heartfelt gratitude to my Supervisor, Prof. Rajeev Kumar for whom this dissertation would not have been possible to complete. His guidance has been immeasurable throughout this dissertation work journey. Also, I would like to extend my special thanks to my seniors and fellow colleagues of D2K lab, who have helped me at the earliest, whenever I needed them.

In addition, I am thankful to my parents and friends, who have endlessly supported me and guided me throughout my dissertation journey. I feel immensely blessed and indebted to have my Supervisor and all my well-wishers beside me through thick and thin.



School of Computer and Systems Sciences
Jawaharlal Nehru University
New Delhi 110 067, India

Declaration

I certify that

1. The work contained in this report has been done by me under the guidance of my supervisor.
2. The work has not been submitted to any other Institute for any degree or diploma.
3. I have conformed to the norms and guidelines given in the Ethical Code of Conduct of the Institute.
4. Whenever I have used materials (data, theoretical analysis, figures, and text) from other sources, I have given due credit to them by citing them in the text of the thesis and giving their details in the references. Further, I have taken permission from the copyright owners of the sources, whenever necessary.



Trishita Mukherjee
(20/10/MT/056)

Date: 31-08-22

Abstract

Discovering anomalies in complex networks is crucial, due to their intricate properties and complexity in the graph mining paradigm. High-end applications like social networks, technological networks, collaboration networks, etc. are divergent and unstructured where several forms of anomalies are present.

Identification of network anomalies is a challenging task that requires topological knowledge of the graph. Complex network anomaly analysis is significant for the welfare and awareness of society. Network-based anomalies are either local or global. Anomaly detection algorithms for discovering global anomalies are quite scarce, and therefore need attention as fraudulent patterns are concealed well inside networks.

In this dissertation, we explored two types of latent anomalies: Node and Community. Suspicious network objects are part of communities. We proposed an algorithm that identifies localized node anomalies within communities based on community detection algorithm and centrality measures. We introduced a community embedding approach for feature extraction of communities in a graph. In addition, we have chosen three clustering algorithms and customized them for implementing the community embedding algorithm to investigate community anomalies in biological, social, and collaborative networks. The algorithms are validated with evaluation metrics to ensure the correctness of the identified anomalous communities.

Keywords: Complex networks, Community, Anomaly detection, Node anomalies, Community anomalies, Clustering.

Contents

Title	i
Acknowledgments	iv
Abstract	vi
List of Figures	ix
List of Tables	x
1 Introduction	1
1.1 Introduction	1
1.2 Terminologies and Background	2
1.2.1 Network Anomalies	3
1.2.2 Network Anomaly Detection Approaches	5
1.2.3 Centrality Measures	7
1.2.4 Community Detection Methods	8
1.3 Motivation	10
1.4 Issues & Challenges	11
1.5 Research Questions	12
1.6 Research Objectives	12
1.7 Organization of the Dissertation	13
2 Literature Survey	15
2.1 Introduction	15
2.2 Algorithms for Anomalous Node Detection	16
2.2.1 Structural Clustering Algorithm in Networks (SCAN)	16
2.2.2 Community Aware Detection Algorithm (CADA)	17
2.2.3 Community Neighbour Algorithm (CNA)	18
2.2.4 Community Outlier Detection Algorithm (CODA)	19
2.2.5 Graph Outlier Ranking Method (GOuTRank)	20
2.3 Algorithms for Community Embedding	21
2.3.1 ComE: Community Embedding	21
2.3.2 SpecRp : Spectral-based Community Embedding	22
2.4 Algorithms for Anomalous Community Detection	22
2.4.1 Co-membership-based Generic Anomalous Communities (CMMAC)	23
2.4.2 Attributed Mining in Entity Networks (AMEN)	23
2.4.3 Anomalous Subgraph Detection	24
2.5 Summary	25

3	Complex Network Analysis	26
3.1	Introduction	26
3.2	Application Specific Networks	27
3.2.1	Synthetic Network Generation	27
3.2.2	Biological Networks	27
3.2.3	Collaboration Networks	28
3.2.4	Social Networks	29
3.3	Characteristics of Complex Networks	29
3.3.1	Small World Phenomenon	30
3.3.2	Density	31
3.3.3	Degree Heterogeneity	32
3.3.4	Clustering Coefficient Distribution	34
3.4	Summary	36
4	Node Level Anomaly detection	37
4.1	Introduction	37
4.2	Localized Community-Based Node Anomalies	38
4.2.1	Problem Formulation	38
4.2.2	Proposed Algorithm	38
4.2.3	Mathematical Explanation	39
4.3	Network Data Statistics	41
4.4	Experimental Results	42
4.5	Discussion	45
4.6	Conclusion	46
5	Community Level Anomaly Detection	47
5.1	Introduction	47
5.2	Anomalous Community Pattern Recognition	48
5.2.1	Problem Formulation	49
5.2.2	Community Representation and Feature Engineering	49
5.2.3	Community Anomaly Detection Algorithms	51
5.2.3.1	Algorithm 1: AC-DBSCAN	51
5.2.3.2	Algorithm 2: AC-CBLOF	53
5.2.3.3	Algorithm 3: AC-SPECTRAL	53
5.3	Network Data Statistics	55
5.4	Evaluation Metrics	55
5.5	Experimental Results	56
5.6	Conclusion	62
6	Conclusion and Future Work	63
6.1	Work Summary	63
6.2	Directions for Future Work	64
	List of Publication	66
	References	67

List of Figures

1.1	A Static Network depicting few dense connections	3
1.2	Communities in a Complex Network applying Community Detection Algorithm	9
2.1	Community-based Node Anomaly-CNA	18
2.2	CODA - Node anomalies within communities	19
3.1	Illustration of a Small world Phenomenon [48]	30
3.2	Degree Distribution (Log-Scale) of Complex Networks	33
3.3	Distribution of Clustering coefficient in Networks	35
4.1	Visualization of the steps of our proposed method	41
4.2	Synthetic network node anomalies (Closeness centrality)	42
4.3	Synthetic network node anomalies (Katz centrality)	42
4.4	Zachary's Karate Club network node anomalies (Closeness centrality)	43
4.5	Zachary's Karate Club network node anomalies (Katz centrality)	43
5.1	Schematic diagram depicting the workflow of Community Anomaly Detection .	49
5.2	t-SNE 2D Visualization of the respective Communities in the Complex Networks	57
5.3	Snapshots of anomalous/normal communities of Drug-Target Network	58
5.4	Snapshots of anomalous/normal communities of Condense-matter collabora- tion Network	59
5.5	Snapshots of anomalous/normal communities of Facebook-pages-company Net- work	59
5.6	Comparative Evaluation of Algorithms based on Silhouette score	60
5.7	Comparative Evaluation of Algorithms based on <i>CH</i> Index	61

List of Tables

3.1	Efficiency of complex networks	31
3.2	Density of complex networks	32
4.1	Properties of complex network data sets	41
4.2	Results of the Proposed Algorithm	44
4.3	Synthetic network anomalous node's Anomaly Scores	44
4.4	Zachary's Karate Club anomalous node's Anomaly Scores	45
5.1	Statistics of complex networks	55
5.2	Evaluation of algorithms based on identified anomalous communities in complex networks	58
5.3	Evaluation of algorithms based on Silhouette Score in complex networks	60
5.4	Evaluation of algorithms based on CH index in complex networks	61

1

Introduction

1.1 Introduction

Complex network anomaly detection has been an emerging field in the graph mining and network science paradigm. Mining network anomalies is challenging due to the heterogeneity and availability of an overwhelming amount of network data. Anomalies can be explored in a myriad of applications such as online social networks [39], fraud detection [35], intrusion detection systems [13], biological networks [49], covid-19 datasets [20], etc. Comprehending anomalies for large networks are *hard* problems. Conventional algorithmic approaches work conveniently for identifying anomalies in small-scale real-world networks. Soft computing approaches, meta-heuristics, and statistical measures give a faster convergence for large-scale real-world networks, e.g., [30, 37, 47, 52]. Due to the scalability and complexity of networks,

it becomes hard to determine network anomalies. Graph-based measures and Machine learning approaches expedite the process of learning global and local anomalies [1]. Embedding techniques [38] are significant for representing complex networks in low dimensional format for exploring inherent outliers.

Complex network objects are interdependent and correlated. Communities formed within networks control their structural and inter-connectivity aspects [3]. Discovering anomalies within communities and recognizing the pattern of abnormal communities is important for understanding the evolution of the network. For example, a few suspicious communities and members are present that perform illegal activities and harm benign users on social networks. Anomalies in technological networks are the web pages that are spam. It is vital to identify the latent anomalies correctly.

In this dissertation, we focused on discovering anomalous members within communities and the patterns of abnormal communities. The words "*network and graph*" are used interchangeably throughout the dissertation, as they are synonymous in literature. Similarly, in the case of "*anomaly and outlier*". In this chapter, we described the background of our work in Section 1.2, including the motivation in Section 1.3, issues and challenges in Section 1.4, research questions in Section 1.5, and the Research objectives in Section 1.6. Lastly, we provided the organization of the dissertation in Section 1.7.

1.2 Terminologies and Background

Complex network anomaly analysis includes two major fields: network science and anomaly detection. Network science is the field of understanding the intricate relationships and the features of complex networks [5]. The earliest work of network science is the "*Seven Bridges of Konigsberg*" [19] proposed by Leonhard Euler. Anomaly detection is the area of study to recognize rare observations whose behaviour is significantly different from other observations of interest [11].

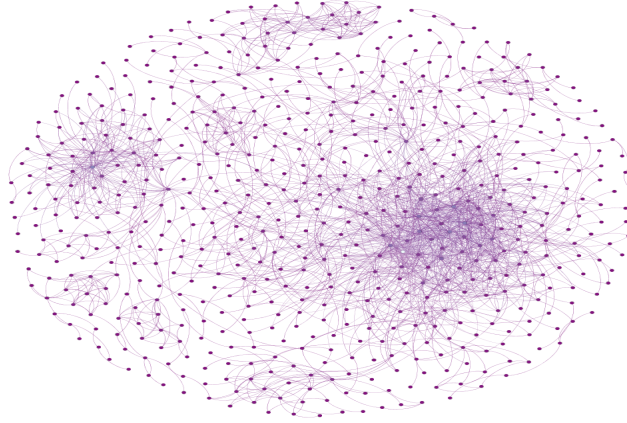


Figure 1.1: A Static Network depicting few dense connections

A complex network is defined as a "set of nodes and edges, where the nodes are connected by edges that represents a relationship between the nodes". They are either static or dynamic. In a static network, the interlinks are fixed and do not change with time, whereas in a dynamic one, the network links and structure change with time. Anomalies in complex networks are investigated using the network's topological features. As the network objects are multidimensional, it is challenging to determine which objects are anomalous. In this section, we detail the nature of network anomalies, network anomaly detection approaches, centrality measures, and community detection methods.

1.2.1 Network Anomalies

Anomalies in the network are defined as objects that deviate significantly from the rest of the objects in the network. Broadly anomalies are of two types: White-crow and In-disguise [11]. White-crow anomalies are the data points that differ substantially from the rest of the points. In contrast, in-disguise anomalies show minor deviations from the set of data points (mainly in clusters). The nature of network anomalies is generalized into four types: *Node anomalies*, *Edge anomalies*, *Subgraph anomalies*, and *Graph anomalies*.

Node Anomalies

Node anomalies are a subset of nodes within respective networks that follows a certain irregular pattern from the rest of the nodes [46]. In a typical scenario, nodes within a static network

are assigned an anomaly score based on certain graph metrics such as degree centrality, eccentricity, ego-net density, etc. Anomalous nodes can be detected globally or locally. In the existing literature, the nodes that bridge (overlap) the clusters are considered as anomalies. In the case of social networks, anomalous nodes can be fraudulent users. The prior knowledge of the network's structure helps in identifying the anomalous nodes. Anomalous nodes can be of three types: global, structural, and community anomalies [38]. Global anomalies consider the attribute information of nodes and select the attributes which significantly differ from other nodes in the network. Structural anomalies are detected based on graph structural information, whereas community anomalies are a hybrid form of both global and structural anomalies.

Edge Anomalies

Edge anomalies are the links within a network that deviates from the normal pattern of the graph and shows certain abnormal behaviour [38]. Anomaly scores are assigned to edges then the scores higher than a given threshold are taken as the anomalous edges. The anomalous edges detected confirm suspicious relations of fraudulent users with benign users. For other domains of the network, the anomalous edges are identified as dominant heavy links or bridge connections amongst the components in the network. Edges are classified as normal or abnormal depending upon the structural information of the domain-specific network. In the state-of-the-art methods, the bridged edges, i.e., edges overlapping other clusters in the networks, are mostly considered anomalous.

Subgraph Anomalies

In real-world social networks, anomalies may behave collectively suspiciously to garner benefits from other users in the network. At a global level, subgraph anomalies are detected where few nodes are clustered together and behave quite differently from the rest of the nodes in the network [3, 38]. Individually, the nodes and edges in the anomalous subgraph behave normally, but when collectively analyzed, they display suspicious characteristics. Therefore, detecting anomalous subgraphs becomes a challenging task. Subgraphs can also be generalized as communities in the network. These types of anomalies are identified by community detec-

tion methods. Suppose, in a recommendation network, if a group of fake users recommends the same type of products constantly, then this group is identified as anomalous subgraphs.

Graph Anomalies

The highest level of an anomaly in a complex network is the graph anomalies [38]. From a database of graphs, certain graphs can appear abnormal from the other graphs. A graph database is a set of K individual graphs. The networks that differ hugely from the set of graphs are labelled as graph anomalies. For example, in biological networks, a network that interacts with drugs and proteins can be analyzed to find unusual structures or patterns. The graph anomalies are the set of networks that conforms to unusual patterns of drug-protein interactions.

1.2.2 Network Anomaly Detection Approaches

Network anomaly detection approaches are broadly categorized as: unsupervised, supervised, and semi-supervised techniques [3]. Unsupervised approaches learn patterns using unlabelled graph datasets to identify graph-based anomalies, such as community-based techniques. Supervised approaches classify the labelled network into normal and anomalous classes. Semi-supervised learning is a hybrid of supervised and unsupervised learning that uses a few labelled graph data and a larger portion of unlabelled graph data, such as the probabilistic-based approaches. Here, we discussed the various network anomaly detection approaches for static networks.

Structural

Structural network approaches use graph representation for the extraction of structural network-centric features. Network-centric features can be global or local. Network-centric features at a node or edge levels are categorized as local network anomalies. Node-level features include in-degree, out-degree, centrality metrics such as closeness, betweenness centralities, radius, degree assortativity, egonets, roles, etc. [3]. Feature-based anomalies extract the local level anomalies. Structure-based methods for anomaly detection in networks have been developed based on the local level anomaly analysis of the network [2, 26]. Structural methods can be

further classified as feature-based and proximity-based. One widely known feature-based approach is ODDBALL [2], which spots anomalous egonet patterns in weighted graphs. Later, Henderson et al. [26] extended the feature-based approach by adding a recursive component combining node and egonet features. Proximity-based approaches [27, 65] measure the closeness of graph objects by capturing the autocorrelation between objects, where the closer objects likely belong to the same class (e.g, normal or abnormal). Link prediction approaches are a classic example that follows the proximity-based technique.

Classification

Classification techniques are applied to labelled graph datasets. It labels the given graph objects into two classes either normal or anomalous. We need labelled networks as classification approaches are supervised in nature. Majorly, deep learning models such as GNN, GCN, etc. [38] use the classification approaches to distinguish the network anomalies. Graph classification techniques takes the graph representation at the node-level or edge-level as the input and generate the normal and abnormal classes. In recent research works, graph anomalies are detected using graph classification algorithms. These algorithms are widely used to understand complex network behaviour in the application of biological networks.

Community

Community or clustering approaches are unsupervised and find the densely connected groups and spot the node/edges that are heavily interconnected with other communities as anomalies. Therefore, the anomalies that are spotted are the “bridge nodes/edges.” Majorly for community-based approaches, we need unlabelled network datasets. Examples of such networks are publication networks, trading networks, social networks, etc. In the literature, an example of a community-based approach is SCAN [64] a structural clustering algorithm that finds network clusters, hubs, and anomalous nodes. Community approaches can also be defined to find anomalous community patterns that deviate from the normal graph behaviour or, in simple terms, sparsely connected communities. Our major work focuses on finding anomalous nodes and communities using community-based approaches.

Probabilistic

Probabilistic-based methods use probability theory and scan statistics functionalities. A model is designed to understand the probabilistic distribution of the networks. Deviations from the normal distribution are categorized as anomalies. Spectral algorithms and Gaussian Mixture Models(GMM) [7] based algorithms use probabilistic approaches to spot node, edge, or sub-graph anomalies. Probabilistic measures include power-law degree distribution, clustering coefficient distribution, etc. Node anomalies, edge anomalies, or subgraph anomalies are computed by assigning probabilistic values based on their attributes and structural information of the network.

1.2.3 Centrality Measures

Centrality measures are utilitarian in analyzing complex networks. They provide a local understanding of the network. Certain values or rankings are assigned to the nodes denoting their network position and influence. Influential nodes can be determined from such measures. These indices are answer to the question "What properties denote an important node?". The answer is denoted by certain values or rankings of the nodes. The propagation of information in the network is also characterized by such measures extending this idea to discover anomalous nodes. We described two of the fundamental centrality measures [21, 28] that are exceedingly used to analyze node anomalies in complex networks.

Closeness Centrality

Closeness centrality [21] is a measurement that helps identify nodes that can diffuse information quickly through a network. The closeness centrality of a node is computed by taking the average shortest path that is reachable to all nodes. Closeness of a node v_1 to another node v_2 in a graph G is computed in Eqn 1.1:-

$$C(v_1) = \frac{1}{\sum_{v_2 \in G} dis(v_1, v_2)} \quad (1.1)$$

Therefore, it is the summation of all the paths from a node v_1 to other nodes in the graph. If

the value of closeness is large, then the corresponding node is quite distant from other nodes. For dense networks, the closeness value is small as the nodes are in compaction. While in sparse networks, the value tends to be larger as the nodes are quite far apart. The lowest possible score of closeness is 1, which depicts that a node is directly connected to everyone. Real-world large networks are known to be sparse. Closeness centrality assumes a network to be undirected but works well even for directed networks.

Katz Centrality

Katz centrality [28] calculates the centrality of a node based on the centrality of its corresponding neighbours. It computes the “relative influence” of a node in the network by measuring the number of immediate or 1-hop neighbours and the other nodes that are connected through these immediate neighbours. By definition in Eqn 1.2, the Katz centrality of node v_1 is:-

$$K(v_1) = \alpha \sum_{v_2 \in G} A_{v_1 v_2} x_{v_2} + \beta \quad (1.2)$$

where A is the adjacency matrix of a network with λ eigenvalues. β controls the initial point of centrality and $\alpha < \frac{1}{\lambda_{max}}$. The significance of this centrality is that it computes a node’s influence by taking the number of walks in the network. Shorter-length walks are exponentially higher valued than longer ones. Katz centrality is effective in analyzing influential nodes in real-world networks, especially in ranking the pages in the world wide web network. Though, as Katz centrality is an eigenvector measure, it is useful only if the network has strongly connected components.

1.2.4 Community Detection Methods

Community detection algorithms unearth the communities present in a complex network. Communities in complex networks are a group of nodes that are more densely interlinked with each other than the rest of the nodes in the network. These methods are major of the "Non-overlapping community detection method" and "Overlapping community detection method." Non-overlapping community detection methods generate distinct communities, whereas Over-

lapping community detection methods find intersecting communities where nodes and edges belong to multiple communities. We discussed two well-known non-overlapping community detection methods [8, 15] based on our dissertation work.

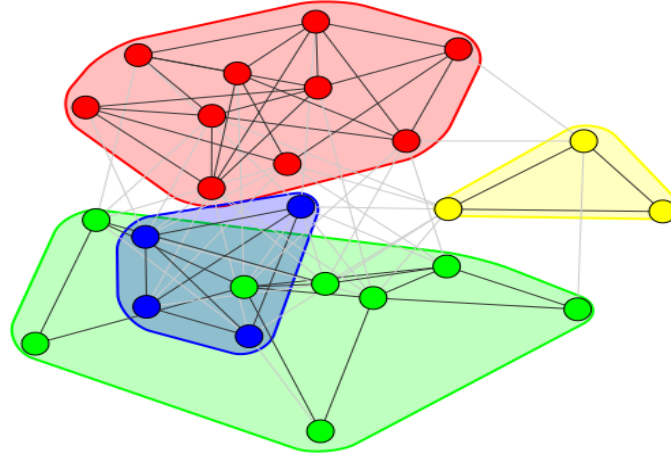


Figure 1.2: Communities in a Complex Network applying Community Detection Algorithm

Louvain Method

The Louvain method [8] is a hierarchical clustering algorithm that applies a recursive procedure for finding disjoint communities and executes modularity optimization techniques on condensed communities. It works on a two-fold mechanism: modularity optimization and aggregation of community. The steps are performed recursively until modularity is maximized. In the modularity optimization stage, the nodes are randomly ordered where, one by one, a node is removed and inserted into a community until there is no significant change in modularity is seen.

In the next phase of community aggregation, nodes in the same community are consolidated in a single node. The connected edges of the singular giant nodes are the summation of the formerly connected nodes belonging to the same different communities. Clustering communities of communities after the first stage, a hierarchical organization of the network is built, leading to several communities for a particular network.

Label Propagation Method

Label propagation method [15] follows a semi-supervised approach that assigns labels to unlabelled graph objects in the network. It is an iterative algorithm that propagates labels throughout the network data set. This method incorporates the advantage of both asynchronous and synchronous models. Each node is allotted a distinct label, and an iterative process is followed so that groups of nodes agree with a label for the formation of communities. Consequently, at each step, the node's labels are updated. Communities are formed where a cluster of nodes exhibits the identical final label w.r.t a particular community.

The synchronous and asynchronous models pose the problem of oscillation of labels. Therefore, Cordasco and Gargano introduced a semi-synchronous approach. The algorithm works in two phases: Coloring and Propagation. In the coloring phase, the nodes are colored such that no two adjacent nodes share a similar color. The propagation phase is divided into a certain number of stages. Stages are named based on a certain color. For stage "k," labels are propagated only to the nodes assigned with the "k" color. The algorithm is stable, allowing limited randomization, leading to uniform results.

1.3 Motivation

This dissertation aims to discover anomalies in complex networks from a local and global perspective. Complex networks possess rich structural information. Partitioning the network allows penetrating deep to extract localized node anomalies for specific communities. Locally, node anomalies can be identified using structural and community-based features. As most complex networks are unlabelled, community detection algorithms and centrality measures form the backbone for node-level anomaly detection. At the global level, various types of networks have communities that operate differently from the rest of the communities. Identifying anomalous communities in social networks is necessary for social welfare. For other domains such as biological or collaboration, it is essential to discover communities as anomalies to understand the evolution of networks.

1.4 Issues & Challenges

In complex networks, there are various challenges in the detection of anomalies [3, 11, 38]. Due to networks' diverse structure and nature, it is impertinent to address these issues. We have enlisted a few of the issues and challenges related to our dissertation that we have addressed in succeeding chapters.

- **Network Structure** - Real-world networks are "heterogeneous" in nature, and therefore, obtaining networks with ground truth labels is a major issue [3, 11]. Supervised learning algorithms have become quite obsolete to work for unstructured and unlabelled networks. The second challenge is to distinguish properly the network object that is normal and anomalous, as they are interdependent. Thirdly, exploring anomalies in large networks incurs a huge time complexity and space. The network representation matrix becomes massive in size. As the network grows, a proper representation learning algorithm needs to be devised.
- **Domain specific challenge** - The anomalies explored in networks vary from domain to domain. Therefore, defining particular objectives for detecting anomalies in a domain is integral. Suppose, in Protein-protein biological network, at the local level, we intend to identify the anomalous proteins that are structurally different from other similar protein structures. In the case of social networks [46], the fraudsters that are hidden can be identified as anomalies.
- **Network aware anomaly detection algorithms** - A challenging task is to design anomaly detection algorithms for complex networks. Exploration of anomalies is heavily dependent on the choice of algorithm. For example, in detecting unattributed network anomalies, unsupervised learning algorithms are required. Outlining network anomaly-aware models and objectives are preminent issues. For specific algorithms, the choice of hyperparameters is crucial as there is a variation of anomalies, as the network scales.
- **Presence and correctness of network anomalies** - Network anomalies can be identified at different topological levels. One of the foremost challenges is correctly detecting

anomalies at the local level (node and edge anomalies) or the global level (community and graph anomalies). The discovery of network anomalies is domain-specific, so it is important to examine the topological nature of the network and whether an anomaly certainly exists at the local or global level. Substantial quantitative measures and visualizations are required for the validation of the correctness of identified anomalies.

1.5 Research Questions

Our dissertation aims to discover the hidden local and global anomalies within complex networks. Specifically, we aim to explore the outlier nodes and anomalous communities for different domains of networks. We designed two research questions for the problems that we addressed in our dissertation:-

Research Question 1. *Can we ascertain latent node level anomalies within each community from a local perspective?*

Research Question 2. *In what manner are the communities of different domain-specific complex networks represented for identifying the anomalous communities correctly?*

1.6 Research Objectives

Our research objectives are mainly based on two concepts to detect node-level and community-level anomalies within the networks. The research objectives defined for this dissertation are:-

1. Detection of localized community-based node anomalies in complex networks based on non-overlapping community detection algorithm and centrality metrics.
2. Discovering communities and aggregating the community members' information by designing a community representation learning mechanism with a feature reduction method.
3. Exploration of anomalous communities from the community representation by applying clustering outlier detection algorithms.
4. Implementation and evaluation of the above-mentioned objectives.

1.7 Organization of the Dissertation

The dissertation is organized as follows: Chapter 2 provides a literature review on the state-of-the-art algorithms in node and community anomaly detection, including a few methods on community embedding. In Chapter 3, we described the various domains of complex networks and their properties with visualizations. We proposed a localized community-based node anomaly detection algorithm and identified the within-community node outliers in Chapter 4. For Chapter 5, we devised a community embedding approach and applied clustering algorithms to detect anomalous communities in different applications of networks. Finally, in Chapter 6, we concluded our work and provided directions for future work.

2

Literature Survey

2.1 Introduction

Complex network anomalies can be either local or global. The anomalies present in static networks can be of various types, such as nodes, edges, subgraphs, and graphs. Previously, we mentioned the background of the types of network anomalies and their identification procedures. In this chapter, we outline the algorithms used to extract two types of existing graph anomalies: Node(local) and Community(global). This chapter describes the existing “state-of-the-art” algorithms on node anomaly detection [22, 25, 42, 61, 64] in Section 2.2 and community anomaly detection [31, 41, 45] in Section 2.4. Additionally, in Section 2.3, we have included the recent works based on community embedding algorithms [58, 66]. Though, the existing research in community-level anomaly detection is still in a nascent stage. Still, we

added a few relevant methods that are in line with our dissertation work.

2.2 Algorithms for Anomalous Node Detection

In this section, we discussed the algorithms [22, 25, 42, 61, 64] that have been proposed by researchers in detecting node-level anomalies in networks. The majority of works have been done using community or clustering-based techniques. The node anomalies identified are specific to the communities in the networks. Network clustering is a crucial task for the exploration of underlying anomalous structures.

2.2.1 Structural Clustering Algorithm in Networks (SCAN)

SCAN [64] is a well-known algorithm based on clustering nodes using structural similarity. The idea of this algorithm is to identify clusters, hubs, and anomalies in networks. The node's neighbourhood is taken as the clustering criteria. Similar nodes are grouped together into clusters. Cross-connection nodes are classified as "hubs," and nodes that do not reside in any cluster or are sparsely connected are identified as "anomalies." SCAN exhibits the following features:-

- It detects "network clusters, hubs and outliers" by exploiting neighbourhood of vertices. Nodes are designated to a cluster depending on how they share neighbours. If a node of a cluster shares an identical structure with one of its neighbours, its computed structural similarity will be large. The proposed method computes all "structure-connected clusters" based on a given parameter setting by inspecting each node of the network.
- The computational complexity of this algorithm is " $O(m)$ ", with n nodes and m edges, which is faster in contrast to modularity-based algorithms.

Identifying hubs and anomalies is essential for applications like biological networks, social networks, etc. SCAN works better for smaller size networks, but for large networks, another algorithm SCAN++ [54] is proposed that detects clusters, hubs, and outliers in an efficient manner. SCAN++ computes the density only for the adjacent nodes, thereby reducing the number of density evaluations. Another method, *pSCAN* [12] similar to these algorithms, aims

to reduce the structural similarity computations and introduces efficient mechanisms to update the clusters when the input network changes dynamically. It also extends the optimization approaches to other similarity metrics, e.g., Jaccard similarity. Therefore, *pSCAN* extends the idea of SCAN and is computationally more effective for SCAN++ and SCAN. Though, the traditional approach SCAN introduces the idea of clustering by structural similarity and detecting hubs and outliers. .

2.2.2 Community Aware Detection Algorithm (CADA)

CADA algorithm [25] is a community-aware approach that identifies anomalous nodes in a global perspective by employing two well-known community algorithms. These algorithms are, namely, the Louvain algorithm (*CADA_L*) [8] and the Infomap approach (*CADA_I*) [50]. This algorithm assigns each node to a particular community employing the two community detection methods. The nodes are allocated anomaly scores, depending on which communities they belong to in the network. CADA explores two kinds of anomalies in a network: Random anomaly and Replaced anomaly.

- **Random anomalies** are the anomalous nodes i.e., " $n/100$ " (where n is the number of nodes) connecting to x randomly existing nodes, where x is between " k (average degree) and k_{max} (maximum degree)" that are infiltrated in the network to see its changing behaviour or patterns established by the "power law degree distribution" of the network.
- **Replaced anomalies** replaces a certain number of nodes from the network; then, randomly, certain nodes are selected wherein an anomaly is inserted by recombining all the edges from the randomly selected nodes to the "new anomaly," later getting removed from the network.

Furthermore, this algorithm is parameter-free and highly efficient. It explores anomalous nodes from a global perspective and also identifies nodes belonging to many communities to one particular community. An advantage of this method is that it is linearly scaled based on the number of edges of the complex network.

2.2.3 Community Neighbour Algorithm (CNA)

Graph partitioning methods efficiently detect anomalies for large-scale complex networks. Though, partitioning large networks are NP-complete problem and require high-quality solutions. Community Neighbour Algorithm (CNA) [61] explores node anomalies i.e., Type3 anomalies within the communities for an attributed network dataset. It identifies node anomalies whose attributes differ majorly from the rest of its community members.

Firstly, the network dataset is visualized for the existence of communities by computing the probability adjacency matrix. Next, the network is partitioned into meaningful communities by applying Markov Cluster (MCL) algorithm [60]. Anomaly scores within the communities are assigned by applying Euclidean distance measures. This technique is particularly useful to identify community anomalies wherein no local anomalies but anomalies exist within the community. This algorithm uses attributed datasets and fits the network in Isolation forest [36] and Autoencoders [6] for identifying "community anomalies."

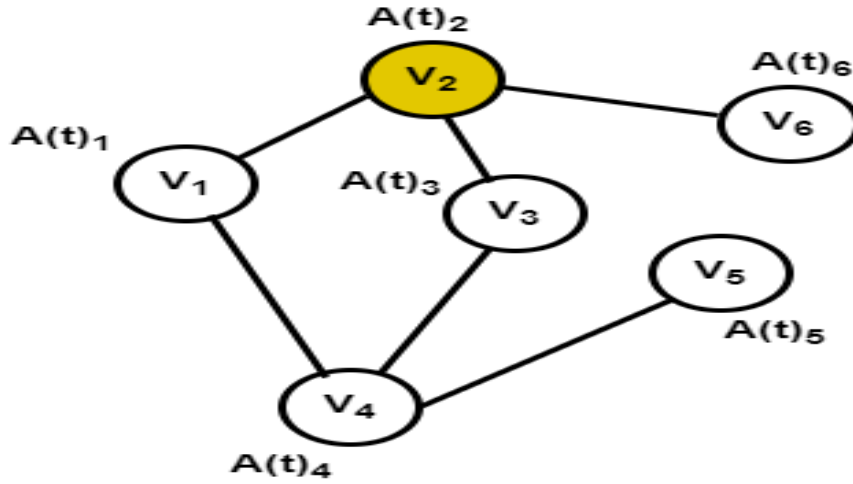


Figure 2.1: Community-based Node Anomaly-CNA

Fig. 3.1 depicts the "community-based node anomaly" identified by CNA. The nodes (V_1 , V_2 , V_3 , V_4 , V_5 , V_6) have specific attributes ($A(t)_1$, $A(t)_2$, $A(t)_3$, $A(t)_4$, $A(t)_5$, $A(t)_6$) associated with them. CNA detects the node V_2 as an anomaly whose attributes are quite different from the other nodes V_1 , V_3 , V_4 , V_5 , and V_6 belonging to the same community.

2.2.4 Community Outlier Detection Algorithm (CODA)

CODA algorithm [22] is a generative model that employs a probabilistic mechanism for community-based anomaly detection in information networks. It unifies the idea of community exploration and anomaly detection by formulating probability statistics based on "Hidden Markov Random Fields (HMRF)." The information obtained from each graph object is taken as a multivariate data point. There are K components in the network that describes normal community behaviour and one component for anomalies. The community components are obtained from Gaussian or multinomial distribution.

A hidden variable(z_i) is induced for each node indicative of its community. The links associated with the network are modeled via the "Hidden Markov Random Field (HMRF)" on the hidden variable. An objective function is designed based on the posterior energy of the HMRF model, and the local minimum is identified by employing an "Iterated Conditional Modes (ICM)" algorithm. In the case of various applications, there is obscurity present in the communities. For large networks, the information of nodes and edges can be noisy. CODA incorporates information from both nodes and edges that reduce the noisy nature of the data and yields a finer solution.

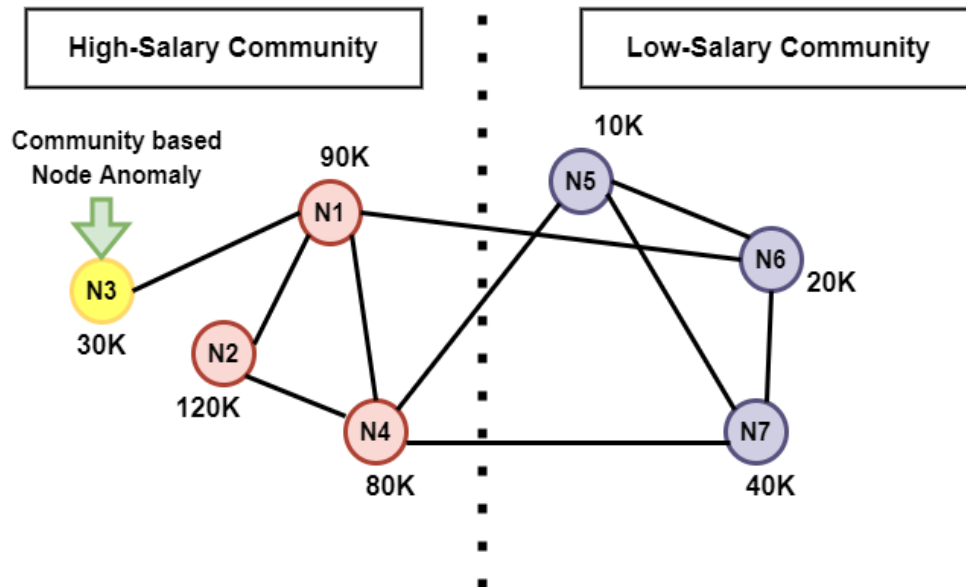


Figure 2.2: CODA - Node anomalies within communities

Fig. 2.2 shows an example of a "community-based node anomaly" detected by CODA. Here, the nodes (N1, N2, N3, N4, N5, N6, N7) are denoted by the persons of interest. The

income of each person is affixed with the nodes. Edges represent the relationship amongst the persons. There are typically two communities: high-salary community (N1, N2, N3, N4) and low-salary community (N5, N6, N7). N3 is an example of a community-based node anomaly, linked only with a high-salary community even after a relatively low salary. This type of person might have great connections with high-salary people or might be settled in an affluent neighbourhood.

2.2.5 Graph Outlier Ranking Method (GOuTRank)

GOuTRank [42] detects anomalous nodes in attributed networks and exploits the concept of graph clusters and their association with subspace analysis. It is designed to extract complex outliers that deviate significantly from a subset of attributes and a local subgraph. It handles the challenge of detecting "hidden" outliers in attributed networks. Outlier scores are assigned to find anomalous nodes in "subspaces of attributed networks". Nodes are ranked based on their degree of divergence in both networks and attributes. This technique introduces two main concepts:-

- **Selection of local subgraphs and subspaces** - In complex network data, densely connected components are clusters having high intra-cluster similarity. The attributed nodes show a correlation between the network structure and a few attribute values. A few clustered nodes exhibit a high similarity of attributes for a subset of selected attributes. Whereas a few subspaces depict a high correlation with the subgraph that is selected. Exploring subspace clusters is a pre-processing step for determining outliers.
- **Anomaly scoring of nodes in different subspace clusters** - Anomaly scoring depends on the frequency of nodes appearing in various subspace clusters. GouTRank applies a two-fold scoring mechanism: (1) Node degree scoring and (2) Eigenvalue scoring. The anomaly scores in different subspaces are computed by integrating these graph centrality measures. The obvious anomalies are not part of any subspace cluster or may belong to sparsely connected clusters. A high deviation value leads to top ranking, indicating the anomalous nodes. Normal nodes belong to densely connected clusters of high dimen-

sions.

This method is able to identify prominent anomalies in attributed networks. Scalability is a major drawback of this method. Two-step integration proves to be costly for detecting outliers in various subspaces for large attributed networks.

2.3 Algorithms for Community Embedding

Community embedding is a "low dimensional" vector space representation of communities in a graph. The characteristic node feature vectors are taken into account while aggregating information for the formulation of community embedding. We have described algorithms [58, 66] on community embedding that uses matrix representation, modularity maximization-based community detection, spectral theory, and deep learning approaches. However, the literature on community embedding methods is quite scarce as it is a relatively evolving research area.

2.3.1 ComE: Community Embedding

ComE algorithm [66] is well-known as the first community embedding approach. Community embedding helps in analyzing graph structure for finding similar pattern communities and assisting in community recommendation. This algorithm provides a novel concept of community-aware high-order proximity where two nodes belonging to an identical community can be seen as close in a low-dimensional space.

The community detection technique is performed to derive communities from the given network. The "Gaussian Mixture Model(GMM) represents the communities in the network" [7]. The Gaussian components thus derived are specified by a mean vector and a covariance matrix. The covariance matrix defines the node's spread. Community prediction is made to get each node's most probable community assignment. The node and community embedding are jointly optimized. A new model ComE+ [10] proposed by the same researchers of the ComE algorithm explores the unknown community assignments and the number of communities that are not specified efficiently. The underlying community embedding structure is examined effectively in both ComE and ComE+.

2.3.2 SpecRp : Spectral-based Community Embedding

SpecRp [58] is a community embedding method pertaining to spectral theory. It primarily takes the community structure, node features, and proximity of nodes into account. The highlight of this algorithm is that it includes an overlapping modularity maximization metric. It explores node attributes but does not need to know the number of communities from beforehand. This method produces both node and community embeddings. The stepwise process followed by this algorithm:-

1. Firstly, It incorporates the input node features, "first and second order proximities" of the node to create an "embedding adjacency matrix".
2. The dimensionality of node attributes is reduced by an autoencoder which generates an "attribute similarity matrix".
3. Then, a combination of an attribute similarity matrix and a second-order proximity matrix generates an embedding adjacency matrix. A graph is designed from the embedding adjacency matrix.
4. SpecDecOV, an overlapping spectral-based community detection heuristic is designed and applied on the graph to produce the node and community embeddings.

Evaluation of this method is done based on Jaccard similarity and F1-score. This method is particularly useful for attributed networks. One disadvantage that SpecRp poses is that it is quite computationally costly. Also, it is applicable for static networks and not well-suited for dynamic networks.

2.4 Algorithms for Anomalous Community Detection

Community level anomaly detection approaches focus on exploring community or global level anomalies [31, 41, 45]. Communities for large-size networks are embedded and assigned probabilistic values and ranked to determine the anomalous ones. Novel measures are designed to extract anomalous communities correctly.

2.4.1 Co-membership-based Generic Anomalous Communities (CMMAC)

Lapid et al. devised an algorithm CMMAC [31] to detect anomalous communities that utilize the nodes' co-membership information within multiple communities. This algorithm is domain-free and unaffected by the densities of the communities. It can detect abnormalities present in both labelled and unlabelled datasets. A classifier is trained to predict the probability of each node belonging to a particular community. Then, the communities are ranked by applying the aggregated probabilities of nodes with respect to a particular community. Communities having the lowest rank are considered to be anomalous. Anomalous communities are infused within randomly generated networks to realize the structure of the networks. The brief working mechanism of this method:-

1. CMMAC works in a two-fold mechanism. Firstly, communities in the network are determined using community detection methods, which are stored in a "partition map". Next, the network is divided into train and test sets. It trains on the data having similar structural properties, and the test set is used to unearth the anomalous communities in the network.
2. Detecting abnormal communities is done in two steps: (a) Two bipartite networks are constructed from the partition maps having a group of nodes that are members of some communities, and (b) Topological features are extracted to train a link-prediction classifier, the meta-features are selected and then ranked. XGBoost [14] is utilized to construct the bipartite link prediction classifier is constructed
3. Finally, the anomalous communities are found at the rear end of the rank-based meta-features.

The algorithm works well when the abnormal communities are disguised in the background, i.e, either sparse or densely connected to other communities.

2.4.2 Attributed Mining in Entity Networks (AMEN)

AMEN [45] proposed by Perozzi et al. ranks communities based on the "normality" measure. A new measure, "normality", is coined to determine the quality of subgraph in attributed net-

works. Normality measure is the summation of internal consistency and external separability. Internal consistency is a covariance matrix of attribute vectors, whereas external separability denotes the border edges leading to quantifying the cross-edges that can be exonerated. A community extraction algorithm is designed that applies normality to discover communities and certain features per community. An objective function is formulated by inferring a few attributes termed "focus" and utilizing attribute weights to maximize the normality score. The focus of the communities having low normality scores is considered to be anomalous.

Community and focus extraction consists of overlapping communities, where a node can belong to multiple communities. Further, community summarization is performed with a multi-criterion objective that selects a subset of communities, covering the entire graph, and is of high quality. This leads to community exploration and detection of anomalous communities that have negative normality. The major contribution of this work is to propose a "normality measure" and generate the "focus" set to differentiate between the normal and anomalous communities. The summaries of the communities can be visualized.

2.4.3 Anomalous Subgraph Detection

Miller et al. [41] proposed an approach to detect anomalous subgraphs by computing the "principal eigenspace value" of a subgraph's residuals matrix. The "residual matrix" is commonly termed as the modularity matrix with respect to community detection. The spectral norms are derived from the adjacency matrix, i.e., signal power, and the residual matrix, i.e., noise power. It basically detects the signal and noise models, where the noise models depict the anomalous subgraphs. Graph Laplacian [55] properties are applied to understand and detect the anomalies that deviate from the normal pattern of the network.

Several statistical-based techniques are applied to analyze the anomalies' variation using the network's spectral properties. Anomalies identified are in the form of signals. The detection algorithms employed for this framework are Chi-squared statistic, Eigenvector L1 norms, and Sparse principal component analysis. Chi-squared statistic detects subgraphs based on the first eigenvector. Eigenvector L1 norms detects subtle anomalies relying on a single eigenvector value. In Sparse principal component analysis [16] a vector is taken that is similar to the

eigenvector having L1 norm constrained. This helps to identify small subgraphs having large residuals. Random networks are generated to observe the pattern of noise signals. For more complicated large real-world networks, this type of framework is intractable.

2.5 Summary

We discussed the existing literature related to node and community anomaly detection algorithms. Also, a few community embedding techniques and representation learning mechanisms are provided in brief as a preprocessing step for community anomaly detection (though it's still a quite new research paradigm) in the case of larger networks. Our work is closely related to these areas focusing on the area of node level and community level anomaly detection in the succeeding chapters.

3

Complex Network Analysis

3.1 Introduction

The main goal of anomaly detection in complex networks is to analyze the behaviour of networks locally or globally. This chapter gives an overview of the behavioural aspects of various networks. Understanding the properties of networks at the node level and the network as a whole is important for anomaly analysis. Detection of anomalies in a network is domain-specific. The probability of whether an anomaly exists locally or globally can be examined through graph-centric metrics. Real-world static networks follow certain statistical graph-level properties based on certain attributes. Therefore, several domains of complex networks (static and undirected) like biological, social, etc., are visualized and analyzed based on density, degree power law distribution statistics, and other graph metrics such as clustering coefficient,

etc. In Section 3.2, we described the various categories of real-world networks. Section 3.3, the main features of complex networks are discussed with visualizations. Finally, we concluded the chapter in Section 3.4.

3.2 Application Specific Networks

Analyzing complex networks of various domains is useful in understanding the inherent anomalies present in the network. Synthetic networks portray a different behaviour from real-world networks. In this section, we provide a brief description of the synthetic network generation and a few of the domain-specific real-world networks.

3.2.1 Synthetic Network Generation

Synthetic networks are generated random graphs to simulate behaviour of real-world networks. Erdos-Renyi [44] graph model is used to generate the synthetic network. The edges are formed using a probability $prob \in (0, 1)$ independent of other edges. This model used the "Bernoulli random variable" to indicate the presence of an edge. The network generated is tightly bound around its mean for a large number of nodes. Random graph models compute the probabilistic events which may be intractable for a fixed number of nodes. The Erdos-Renyi model generates Gaussian networks for a large N (number of nodes).

3.2.2 Biological Networks

Biological networks give an overview of the network connections based on drug-target, cell-cell, function-function, gene-gene, etc. We described two datasets, mainly the drug-target network and the function-function network.

1. **Drug-target network** [40] is a complex network of drug-gene interactions based on which genes (i.e., genes encoding the proteins) the drugs target, which is based on the U.S market. Drug target information facilitates drug discovery, design, screening, drug interaction prediction, metabolism prediction, and pharmaceutical research. These interactions are composed of biotech drugs where on average, the drugs target 5-10 target

proteins. The drug target is composed of protein complexes which further constitute various sub-components of proteins. The nodes are of two types: drug and gene. The edges represent the drug-gene interaction.

2. **Function-function network** [40] classifies the biological networks in a hierarchy. Specification of the hierarchy is done by Gene Ontology, which outlines the biological functions and relationships between them. Biological functions are the biological processes (i.e., pathways and larger processes made up of the activities of multiple gene components), cellular components (i.e., organelles where gene components are active), and molecular functions (i.e., molecular activities of gene components). The data is extracted from Gene Ontology, which describes the concepts related to biological functions, and how these functions are associated with each other.

3.2.3 Collaboration Networks

Collaboration networks represent the relationships of scientific collaborations amongst the authors. Suppose an author 'm' co-authored a paper with author 'n,' and an edge is established from 'm' to 'n.' If 'p' authors co-author the paper, then a completely connected subgraph of 'p' nodes is generated. Here, we have taken two well-known collaboration network datasets for our analysis.

1. **Astro Physics Collaboration network** [34] covers the scientific collaborations by multiple authors submitting papers in the Astro Physics category. This dataset covers papers from January 1993 to April 2003. Here, the nodes are the authors, and relationships are the several collaborations done between authors on a variety of Astro Physics category papers [33].
2. **Condense Matter Collaboration network** [34] is from the e-print arXiv and covers scientific collaborations between authors' papers submitted to Condense Matter category. The data covers papers from January 1993 to April 2003.

3.2.4 Social Networks

Social networks define relationships of people with similar interests in sharing information. The majority of anomalies are detected in social networks. Entity-entity links form these networks. The nodes are entities, and the edges are relationships amongst the entities.

1. **Zachary's Karate Club network** [49] is created from the "members of the karate club by Wayne Zachary". Here, the nodes represent the members of the club, and edges are the relationship tie between two members.
2. **Facebook-pages-company network** [49, 51] is a collection of verified company Facebook pages (November 2017) based on companies category. The nodes are the pages, and the edges are the pages associated with having similar category likes.
3. **Facebook-pages-tvshow network** [49, 51] contains a set of verified Facebook pages (November 2017) pertaining to tv-shows category. Here, the nodes are represented by the pages, and the edges are the linked pages with mutual contents.
4. **Social-hamsterster network** [49] is created from the social and family relationship amongst the users of the hamsterster website. Nodes are the users of hamsterster website, and edges represent the social ties between the users.

3.3 Characteristics of Complex Networks

Complex networks showcase certain properties based on the graph metrics. Characterization of complex networks helps in the evaluation of network behaviour. Each complex network exhibits certain topological features based on its connectivity and degree distribution. Network properties are determined at the local and global levels. Node level (local) synthesis is done by degree distribution, clustering coefficient, bridges, small world phenomenon [29, 62] etc. Community-level (global) views are generally based on modularity, clusters, density, connected components, etc. We have described four major characteristics of complex networks.

3.3.1 Small World Phenomenon

Small world phenomenon [29, 62] is the principle that forms short chain links of the nodes of a network. The feature of the network is that it exhibits a high clustering coefficient and shorter global separation. According to Watts and Strogatz small world network satisfies two properties:-

1. At the global level, the average shortest path length is small.
2. Locally, the nodes of a network have a high clustering coefficient.

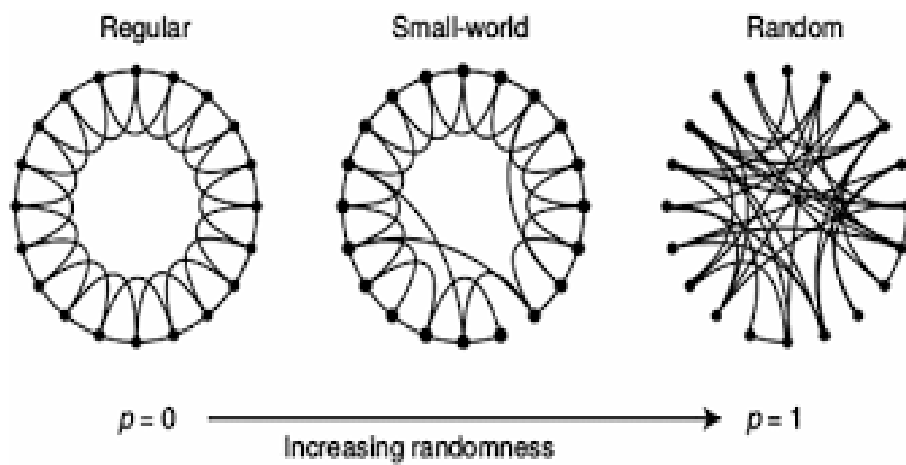


Figure 3.1: Illustration of a Small world Phenomenon [48]

It superimposes the homophilic property of the networks, where nodes of similar nature are connected and weak ties, where the branching structure is wide so that many nodes are reachable in a few hops. It generally follows the concept of "six degrees of separation". In six degrees of separation, nodes are linked at six or a few connections away from each other. The efficiency [32] of small-world networks is determined by "global efficiency" and "local efficiency" values. The efficiency of a network is the reciprocal of the shortest path length between a pair of nodes. The average global efficiency of a network denotes the average efficiency for all pairs of nodes. Similarly, the local efficiency of a node is the average global efficiency based on the induced subgraph by neighbours of the node. Efficiency values range from 0 to 1; where a value of 0 indicates minimum efficiency and a value of 1 indicates maximum efficiency.

Table 3.1 depicts the corresponding local and global efficiency of networks. The Drug-target network has the lowest local efficiency value of 0.0. Astro-Physics collaboration network has the highest local and global efficiency among all networks. The Function-function network has the lowest global efficiency.

Table 3.1: Efficiency of complex networks

Network	Nodes	Edges	$Efficiency_{local}$	$Efficiency_{global}$
Synthetic	100	200	0.471	0.263
Drug-target	7,341	15,138	0.0	0.147
Function-function	46,027	1,06,510	0.159	0.101
Astro-Physics collaboration	18,772	1,98,110	0.718	0.232
Condense-matter collaboration	23,133	93,497	0.687	0.169
Zachary's Karate Club	34	78	0.645	0.492
Facebook-pages-company	14,113	52,310	0.292	0.202
Facebook-pages-tvshow	3,892	17,262	0.444	0.179
Social-hamsterster	2,426	16,630	0.640	0.207

3.3.2 Density

Network density [4] is the ratio between the connections present in a network to the maximum number of connections that the network can contain. It provides the idea of how dense a network can be. It is effective to compute the graph density when the size of the network is large. A complete graph will have a network density value as one i.e., the maximum density. If the density value is 0, the network is disconnected without any edge. Large real-world networks are sparse; therefore density value of such networks is low. The density of an undirected network is given as:-

$$D_{undirected} = \frac{2e}{v(v-1)} \quad (3.1)$$

Similarly, the density of a directed network is given as:-

$$D_{directed} = \frac{e}{v(v-1)} \quad (3.2)$$

In Eqn. 3.1 and Eqn. 3.2, v is the number of nodes, and e represents the number of edges.

We computed the density values of networks to give an overview of how dense or sparse a network is based on its size.

Table 3.2: Density of complex networks

Network	Type	Density
Synthetic	Undirected	0.0404
Drug-target	Undirected	0.0005
Function-function	Undirected	0.0001
Astro-Physics collaboration	Undirected	0.0011
Condense-matter collaboration	Undirected	0.0003
Zachary's Karate Club	Undirected	0.1390
Facebook-pages-company	Undirected	0.0005
Facebook-pages-tvshow	Undirected	0.0023
Social-hamsterster	Undirected	0.0056

Table 3.2 shows the density of the complex networks. The Function-function network has the lowest density amongst other networks. The maximum value density is of Zachary's Karate Club network, as the network is quite small in size with respect to other networks.

3.3.3 Degree Heterogeneity

Degree heterogeneity [43] implies how the nodes are connected with other nodes in the network. The degree of a node v_1 is the number of connections it has with other nodes in a network. To understand the complex behavioural patterns of a network, we analyze the degree distribution pattern. For undirected graphs, the average degree is defined in Eqn. 3.3:-

$$AvgDeg = \frac{1}{n} \sum_{i=1}^n Deg_i \quad (3.3)$$

where i iterates over the number of nodes till n , and Deg_i is the degree of each node i . Real-world networks [17] follow a "power-law degree distribution", where the distribution is heavily right-tailed. Networks exhibiting power-law degree distributions are scale-free. As real-world

networks tend to be sparse, most nodes have a small degree, and only a few (also known as hubs) have a larger degree. In contrast, synthetic networks [57] demonstrate a binomial degree distribution where a larger probability of nodes possesses an average degree, and the distribution curve is bell-shaped. We visualized the degree distributions of a set of complex networks ranging from synthetic to real-world networks.

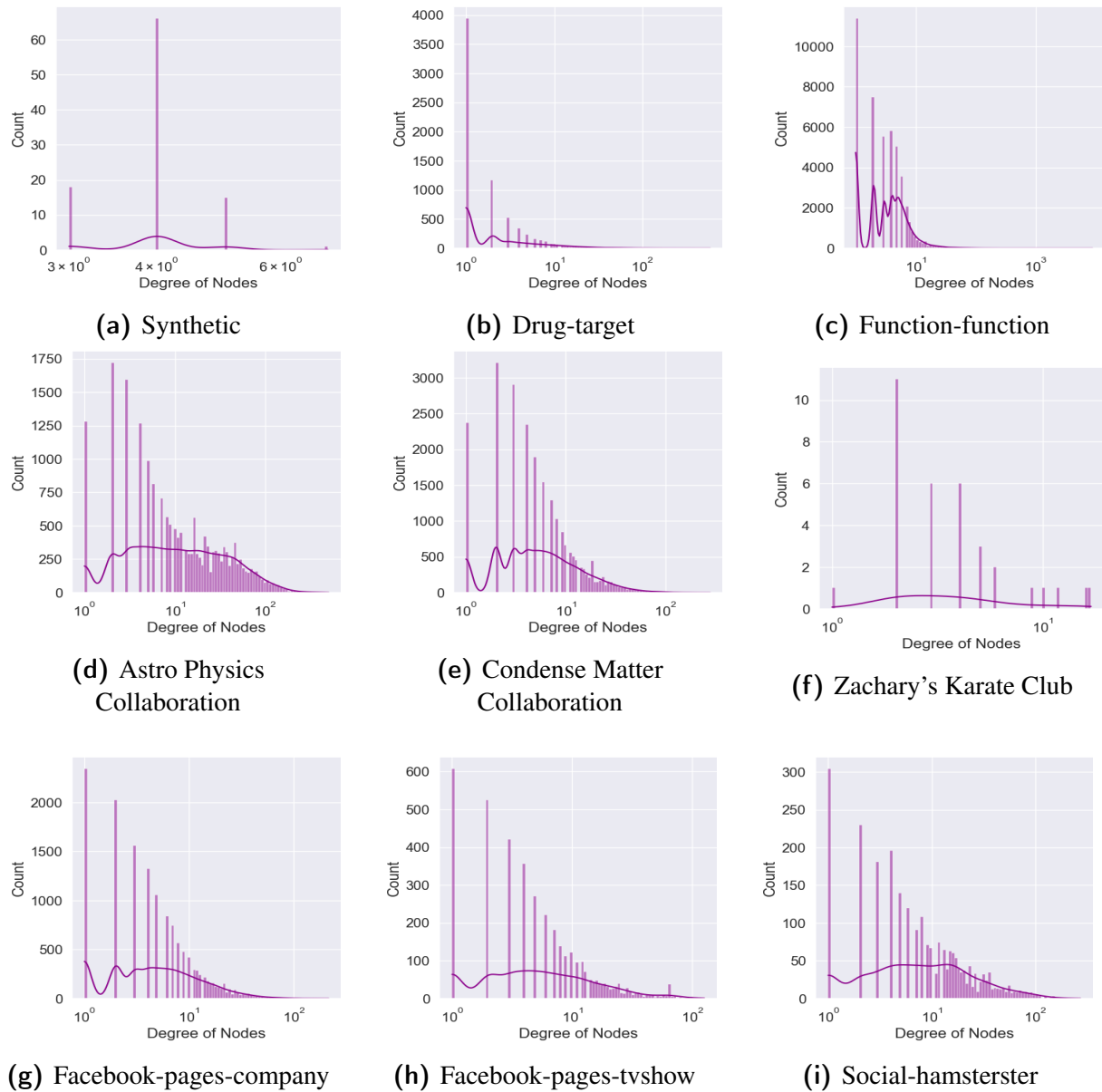


Figure 3.2: Degree Distribution (Log-Scale) of Complex Networks

Fig. 3.2 show a series of degree distributions of complex networks where only Fig. 3.2a shows a binomial distribution, while the rest of the figures depict a power-law scale-free degree distribution where the probability of nodes with a small degree is high. The nodes having large

degrees form the hubs of the network are low in quantity.

3.3.4 Clustering Coefficient Distribution

The clustering coefficient [63] is an important measure to observe the behaviour of nodes in a network. It is defined as the way nodes are connected or clustered with each other in the network. Real-world networks have homophilic properties where similar nodes are clustered together, and dissimilar ones do not belong to any cluster. The clustering coefficient is the exclusive property of nodes in a graph concerning their neighborhood density. Nodes in a network tend to form clusters together. The distribution of the clustering coefficient is either local or global. The local level indicates the clustering of nodes in a graph. The global version gives the average clustering coefficient of the network.

The clustering of nodes in the network follows the triangle power law. The Triangle power law measures the number of nodes that forms a triangle and follows an exponential distribution. In the case of unweighted and undirected networks, the clustering coefficient for v nodes gives the fraction of possible triangles.-

$$C_v = \frac{2 * Triangle(v)}{d(v)(d(v) - 1)} \quad (3.4)$$

In Eqn. 3.4, $Triangle(v)$ depicts the fraction of triangles for node v and $d(v)$ is the degree of node v . Real-world networks exhibit a high clustering coefficient. Nodes of similar nature together form tightly knit groups. Random networks have low clustering coefficients due to a low variation of node degrees. The hubs (nodes having the highest degree) have a relatively higher clustering coefficient than the rest. This measurement helps identify the network's ego nodes (a central node to which all other nodes are heavily connected).

In Fig. 3.3, we visualized the clustering coefficient of nodes in different categories of networks along with the average clustering coefficient (shown in the red dashed line). Fig. 3.3a is a synthetic (random) network that depicts a low scattering of nodes based on the clustering coefficient. The rest of the figures are real-world networks that shows a high probability of variation of the node's clustering coefficient.

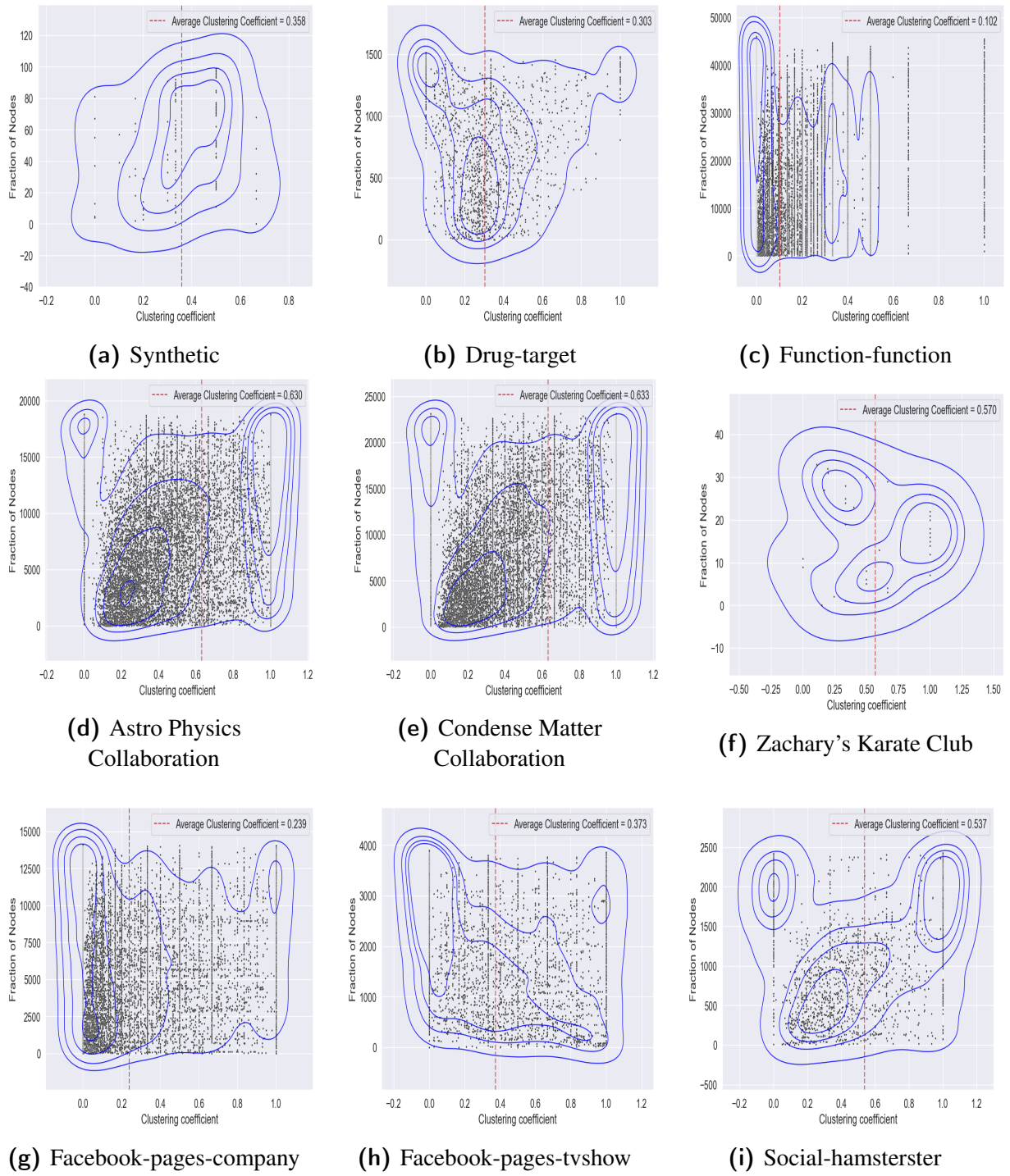


Figure 3.3: Distribution of Clustering coefficient in Networks

3.4 Summary

In this chapter, we addressed the inherent properties of complex networks with quantitative measures and graphical visualizations. We have taken a few domains of complex networks that our dissertation is majorly focused on. These properties govern the characteristic features of application-specific networks and help in discovering the underlying network objects (locally or globally) that supposedly exhibit a certain kind of abnormality from the rest of the objects.

4

Node Level Anomaly detection

4.1 Introduction

In this chapter, we proposed a novel algorithm to detect localized community-based node anomalies in complex networks. Nodes within communities that are hidden and in-disguise are detected. We described the proposed method in detail in Section 4.2. In Section 4.3, we provided the network datasets description for performing the experiments. Section 4.4 evaluates the proposed algorithm based on the experiments. Finally, in Section 4.5 and Section 4.6, we provided the discussion and conclusion of the work, respectively.

4.2 Localized Community-Based Node Anomalies

Certain nodes in the network are in-disguise and to extract these types of nodes, we need to analyze the networks from a local perspective. One approach can be to divide the network into distinct communities and locate the in-disguise node anomalies within the communities. Therefore, such nodes can be termed as localized community-based anomalies, as their behaviour significantly changes with respect to the communities they belong. Node centrality measures [21, 28] can be used to locate the hidden possible anomalies within the communities. A non-overlapping community detection algorithm [8] is applied to partition the network into distinct communities, and then node centrality measures are computed to discover the localized node anomalies. We already mentioned the related literature in Chapter 2 (Section 2.2). We formulated the problem definition and proposed an algorithm to detect localized community-based node anomalies with a mathematical explanation.

4.2.1 Problem Formulation

We defined our problem as identifying localized community-based node anomalies in a complex network by applying the non-overlapping community detection method, i.e., the Louvain method. Each node's identifier in the complex network is assigned an anomaly score based on two centrality measures: (i) Closeness centrality and (ii) Katz centrality. The node identifier having a minimal anomaly score is extracted from the respective communities.

4.2.2 Proposed Algorithm

The proposed algorithm extracts communities and anomalous node identifiers from the respective complex network data sets. The algorithmic steps are included in Algorithm 1.

Algorithm 1 Localized Community based Node Anomalies Algorithm

Input: Complex network data set.

Output: Localized community based node anomalies.

1. The complex network data set is visualized from the given network edge list, i.e., edges represented amongst the linked node IDs.
2. Extract communities (C) using the Louvain community detection method and store the number of communities in N .
3. For each community C_i where $i \in (1, 2, \dots, N)$ extracted from the network:
4. Compute anomaly scores of the node ID's based on the following two sets of centrality measures:-
 - (a) $AnomNodeID_{Closeness}$ is calculated for each node ID based on the Closeness centrality measure.
 - (b) $AnomNodeID_{Katz}$ is calculated for each node ID based on the Katz centrality measure.
5. Compare each anomaly scores of $AnomNodeID_{Closeness}$ and $AnomNodeID_{Katz}$.
6. The minimal anomaly score is obtained for each set of centrality measures. The minimal anomaly score node ID based on closeness centrality measure is stored in $Min_{Closeness}$ and for Katz centrality measure is stored in Min_{Katz} .
7. Repeat steps 3 to 6 until all minimal anomaly scores node IDs are extracted from each community, respectively.
8. Localized node anomalies from communities are identified for each centrality measure.

4.2.3 Mathematical Explanation

Given a static, undirected graph $G = (V, E)$ where V denotes the set of nodes and E the set of edges, respectively. First, the graph G is partitioned using the Louvain community detection method [8]. Louvain's method is based on a heuristic of modularity maximization. This method works in two phases: Phase 1 assigns the nodes by local optimization to a particular community, and Phase 2 computes the maximum positive modularity gain of a node by moving it to all of its neighbouring communities; if there is no positive gain, the node remains in its community. The computation of modularity gain is as follows:-

$$\Delta Q = \frac{S_{j,in}}{2m} - \gamma \frac{\sum_{tot} S_j}{2m^2} \quad (4.1)$$

In Eqn. 4.1, j is the isolated node moving into a community C . Here, m is the size of the network, $S_{j,in}$ is the summation of the weights of the edges from node j to other nodes in the particular community C , S_j is the summation of the weights of the edges incident to node j , Σ_{tot} is the summation of the weights of the edges incident to nodes in the respective community C and γ is the resolution parameter.

The Anomaly scores of the node ID's based on the Closeness [21] and Katz [28] centrality measures are computed as follows:-

$$AnomNode(u)_{Closeness} = \frac{n-1}{\sum_{v=1}^{n-1} dist(v,u)} \quad (4.2)$$

In Eqn. 4.2, $dist(v, u)$ denotes the shortest path between node v and node u , and n is the number of nodes that takes to reach node u .

$$AnomNode(u)_{Katz} = \sum_{k=1}^{\infty} \sum_{v=1}^n \alpha^k (A^k)_{vu} \quad (4.3)$$

In Eqn. 4.3, A represents the adjacency matrix of graph G having eigenvalues λ . Here, k is the number of degree connections between nodes u and v . The α value is chosen in such a way that it is the reciprocal of the absolute value of the largest eigenvalue λ of A .

For each community C_i generated, where $i \in (1, 2, \dots, N)$ in Eqn. 4.4 and Eqn. 4.5:

$$Min_{Closeness} = minimum_{C_i}(AnomNode(u)_{Closeness}) \quad (4.4)$$

$$Min_{Katz} = minimum_{C_i}(AnomNode(u)_{Katz}) \quad (4.5)$$

In Eqn. 4.4 and Eqn. 4.5 the node ID having the minimum anomaly score is extracted from the respective communities.

Fig. 4.1 depicts the visualization of a network on the working mechanism of our proposed algorithm. Firstly, the network is partitioned into two communities after applying the Louvain algorithm. The Anomaly scores of node IDs are computed based on Eqn. 4.2 and Eqn. 4.3. The corresponding node IDs having minimum values in their respective communities are computed using Eqn. 4.4 and Eqn. 4.5. Two types of Anomaly detection are done based on the centrality

measures. Therefore, Node1 and Node6 are the anomalous nodes detected with respect to the two communities when Closeness centrality has been taken as an anomaly score, whereas, Node1 and Node4 are for Katz centrality, respectively.

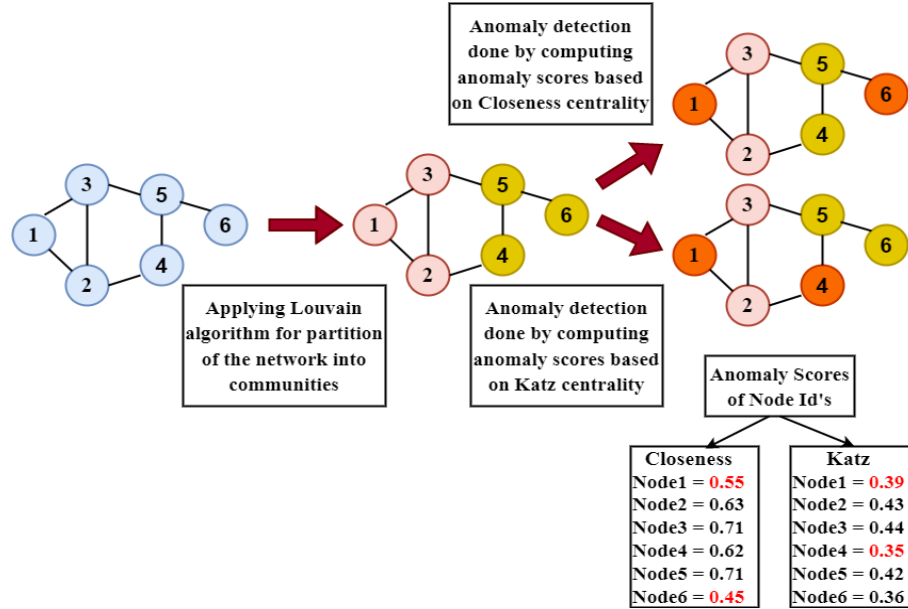


Figure 4.1: Visualization of the steps of our proposed method

4.3 Network Data Statistics

We have selected two different complex networks [49] and [23] to qualitatively assess the localized community-based node anomalies present in networks. The network data sets taken are static, unlabelled, and undirected.

Table 4.1: Properties of complex network data sets

Network data set	Number of Nodes	Number of Edges	Average Degree
Synthetic network	100	200	4.0000
Zachary's Karate Club network	34	78	4.5882

Table 4.1 depicts the features of the two complex networks based on the number of nodes, number of edges, and the average degree.

4.4 Experimental Results

We have performed experiments based on our proposed algorithm on the two network data sets taken for our empirical analysis. Also, we have included figures of the network data sets with respect to the communities and the respective localized community-based node anomalies identified in each of the complex networks.

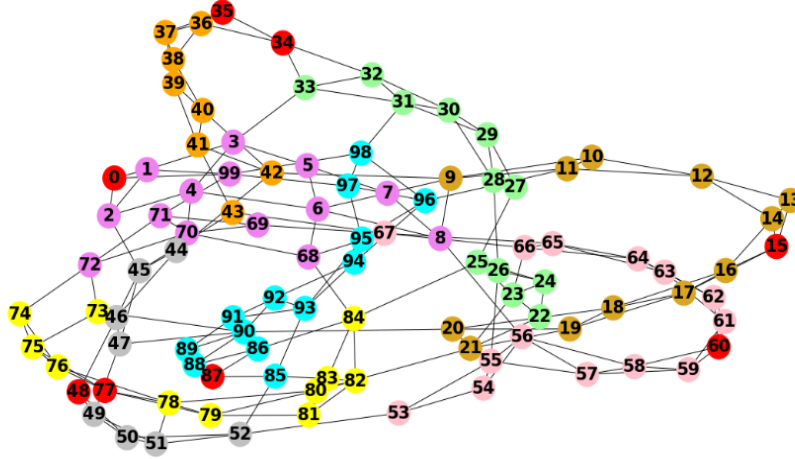


Figure 4.2: Synthetic network node anomalies (Closeness centrality)

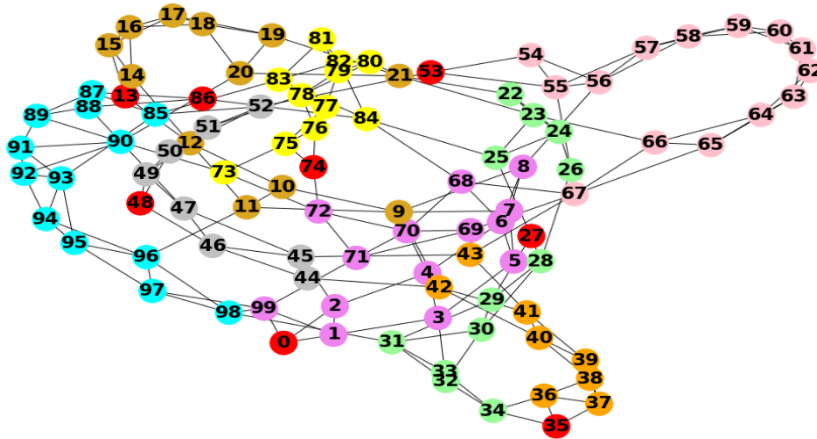


Figure 4.3: Synthetic network node anomalies (Katz centrality)

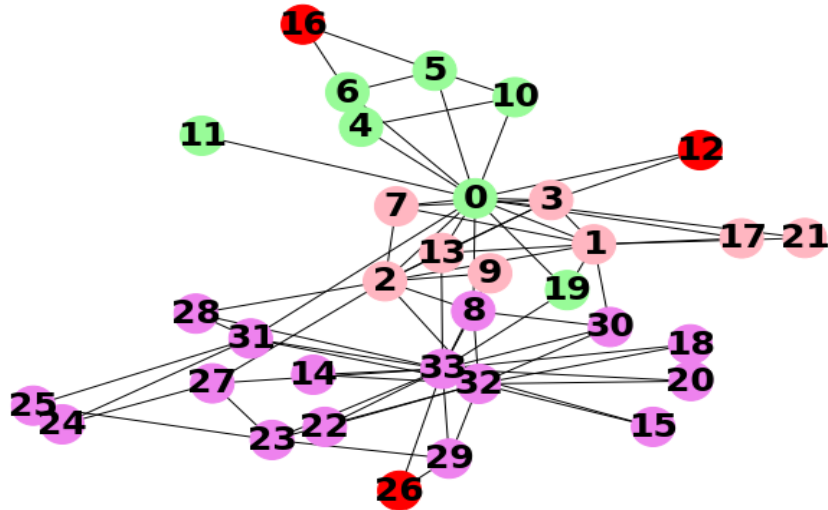


Figure 4.4: Zachary's Karate Club network node anomalies (Closeness centrality)

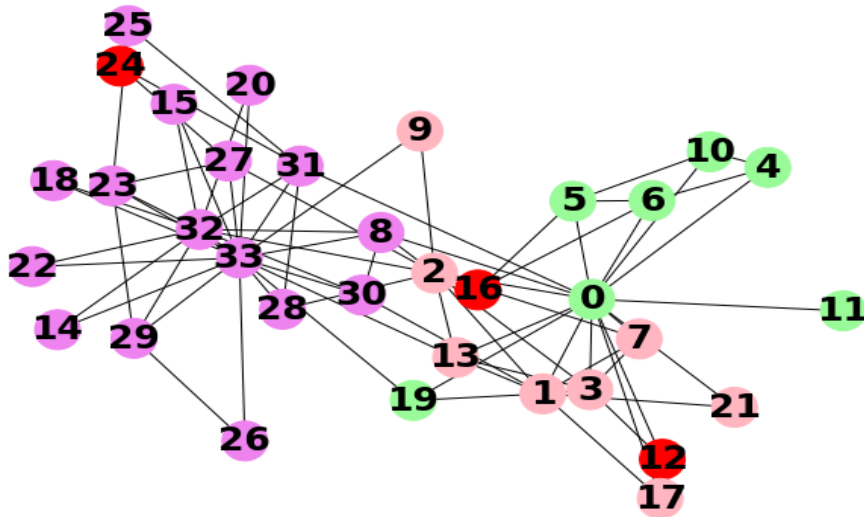


Figure 4.5: Zachary's Karate Club network node anomalies (Katz centrality)

We have demonstrated the experiments using our proposed algorithm where Fig. 4.2 and Fig. 4.3 shows the network illustration of the Synthetic network data set, which is partitioned into eight communities. Whereas, Fig. 4.4 and Fig. 4.5 depict the network illustration of Zachary's Karate Club network data set divided into three communities. For each network data set, we have given distinct colors to the communities and colored the anomalous node IDs as red.

Table 4.2: Results of the Proposed Algorithm

Network data set	Closeness centrality		Katz centrality	
	Communities	Anomalous node ID's	Communities	Anomalous node ID's
Synthetic network	8	0, 15, 34, 35, 48, 60, 77, 87	8	0, 13, 27, 35, 48, 53, 74, 86
Zachary's Karate Club network	3	12, 16, 26	3	12, 16, 24

Table 4.2 summarizes the results of our proposed algorithm based on the computation of the number of communities and anomalous node IDs extracted respectively to the particular communities on these two network data sets. Anomalous node IDs numbered 0, 35, and 48 are detected the same with respect to three of the communities based on the anomaly scores computed with respect to the two different centrality measures in the Synthetic network data set. Similarly, for Zachary's Karate Club network data set, the anomalous node IDs detected the same for two of the communities numbered 12 and 16. Though, the anomaly scores of the node IDs computed with respect to the two centrality measures generates different values.

Table 4.3: Synthetic network anomalous node's Anomaly Scores

$Min_{Closeness}$ (Anomalous node ID's)	Anomaly Score (Closeness centrality)	Min_{Katz} (Anomalous node ID's)	Anomaly Score (Katz centrality)
0	0.2106	0	0.0902
15	0.1755	13	0.0887
34	0.1784	27	0.0891
35	0.1615	35	0.0888
48	0.2071	48	0.0889
60	0.1713	53	0.0890
77	0.1864	74	0.0889
87	0.2143	86	0.0904

Table 4.4: Zachary's Karate Club anomalous node's Anomaly Scores

$Min_{Closeness}$ (Anomalous node IDs)	Anomaly Score (Closeness centrality)	Min_{Katz} (Anomalous node IDs)	Anomaly Score (Katz centrality)
12	0.3708	12	0.1161
16	0.2845	16	0.0907
26	0.3626	24	0.1102

Table 4.3 depicts the anomaly scores of anomalous node IDs based on: (i) Closeness and (ii) Katz centrality for each of the eight communities that are computed using the Louvain algorithm for the Synthetic network data set. Similarly, Table 4.4 shows the anomaly scores of anomalous node IDs for Zachary's Karate Club network data set. Anomalous node IDs are thus selected based on the nodes having minimum anomaly scores within the respective communities. The significance of the values depicts the variation of the Anomaly Scores of Node IDs in their respective communities based on Katz centrality and Closeness centrality.

4.5 Discussion

The experiments demonstrated gives an overview of how anomalous nodes within communities are detected in complex networks using our proposed algorithm. Our algorithm works in two phases based on finding communities in a network and anomaly detection within communities. Our findings are suggestive that localized node anomalies that are discovered by our proposed method are the node IDs that exhibit a certain dissimilarity from the other node members present within the particular community. Also, from the previous related work, we have seen and re-iterated, by our experiments too, that Katz centrality is a better measurement to find anomaly scores of the nodes within the communities rather than closeness centrality. Finding anomalous nodes exploits the underlying structure of the communities in complex networks. Certain anomalous node IDs are similarly identified as belonging to their communities when the two different centrality measures taken as anomaly scores are applied for two different network data sets; at the same time, other anomalous node IDs are differently detected.

4.6 Conclusion

We computed localized community-based node anomalies by partitioning a network into different communities. When two different centrality measures are computed, there is a variation in the anomalous node IDs extracted from the communities. Our results also establish that the Katz centrality measure, when taken as the Anomaly Score of nodes, is likely an effective method as it uses the eigenvector centrality approach rather than the Closeness centrality measure, which uses the shortest path approach amongst the node IDs. Katz's centrality effectively prunes the network. Our proposed method identifies locally anomalous nodes present within a particular community in a complex network. The proposed algorithm can be applied to various other real-world network data sets by scaling the number of nodes and edges to compute such localized node anomalies within communities. In the future, other non-overlapping community detection approaches can be used with these centrality measures. Hybridization approaches can also be applied for anomaly detection of large real-world networks.

5

Community Level Anomaly Detection

5.1 Introduction

In Chapter 4, we devised an algorithm to identify node-level anomalies in complex networks. Now, this Chapter explores community-level anomalies from a global perspective. We provided an extensive description of the methodology for detecting the prominent anomalies for domain-specific networks. A challenging area is to identify anomalous communities correctly. Therefore, the features of communities need to be extracted and represented properly. In Section 5.2, we outlined the phases for detecting anomalous communities. In Section 5.3, we described network datasets and in Section 5.4, we provided the description of the evaluation metrics. Section 5.5, we discussed the experimental results of the corresponding domain-specific networks based on the clustering outlier algorithms. Finally, in Section 5.6, we conclude this

work.

5.2 Anomalous Community Pattern Recognition

Identification of anomalous communities is crucial and can be well understood through community representation. Fraudulent communities can be well-masked within the network. So, to detect such anomalies, we need to preprocess the networks into a simpler format like the adjacency matrix. Large network representations are computationally costly. Therefore, it is imperative to design community matrix representation algorithms to include only the key features of the network in less space and time complexity. The traditional clustering-based outlier detection methods can be modified to work for community feature matrices. An example of anomalous communities can be a fraudulent group of users linked with similar fraudulent communities on social networks. In biological networks, the interaction of drugs or protein communities that are linked with a few similar communities but are quite distant from the rest of the communities can be identified as anomalous as they do not contribute much or rather quite insignificant in the network.

We covered the state-of-the-art algorithms in Chapter 2 (Section 2.3 and Section 2.4) related to community embedding techniques and community anomaly detection respectively. Community anomaly detection is mostly done on attributed networks. Real-world networks are unstructured, and therefore, networks that do not have inherent attributes should be analyzed for potential hidden community anomalies. It is still a nascent area to detect anomalous community anomalies based on community representation learning. we proposed a community feature matrix creation from the given large static network after detecting the communities using a non-overlapping community detection method and transformed it into a low-dimensional format by "T-distributed stochastic neighbour embedding (t-SNE)" [59]. Finally, we have taken three well-known algorithms (DBSCAN, CBLOF, SPECTRAL) and prefixed "AC" namely, AC-DBSCAN, AC-CBLOF and AC-SPECTRAL, for exploring anomalous communities from community matrix representation based learning of domain-specific network datasets. We have shown a schematic diagram for the phases of the community-level anomaly detection in Fig 5.1.

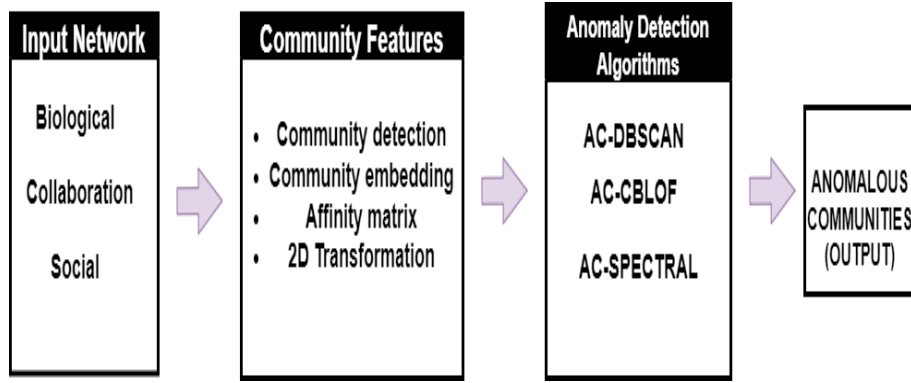


Figure 5.1: Schematic diagram depicting the workflow of Community Anomaly Detection

5.2.1 Problem Formulation

We formulated our problem to identify community anomalies for static undirected networks. Our problem is formulated in a three-fold mechanism:

1. Partitioning the network into communities based on label propagation algorithm.
2. Defining community embedded feature vectors of the network through community representation feature matrix.
3. Performing anomaly detection on community embedded feature vectors using clustering outlier detection algorithms to identify the anomalous communities.

5.2.2 Community Representation and Feature Engineering

Community detection methods are integral in extracting the communities of a network. To learn the nature of the communities, it is important to understand the intra-linkage and inter-linkage patterns of communities. For large networks, identifying anomalous communities becomes a humongous task. Therefore, it becomes crucial to embed the features of the community with respect to the inter-connections and intra-connections of nodes belonging to a variety of communities in a matrix format. Firstly, the distinct communities are identified by applying a non-overlapping community detection method (i.e., Label propagation method). Next, we proposed

a community-embedded feature matrix algorithm. The algorithmic steps are incorporated in Algorithm:2 for a simplified overview.

Algorithm 2 An adjacency matrix for embedding community feature vectors

```

1: procedure COMMATRIX( $G$ )  $\triangleright$  The adjacency community matrix, input is the network  $G$ 
2:    $C \leftarrow \text{label-propagation}(G)$   $\triangleright$  Community extraction applying Eqn 5.1
3:    $Com \leftarrow []$   $\triangleright$  Null community adjacency matrix
4:   for  $s$  in  $\text{Length}(C)$  do
5:     for  $t$  in  $\text{Length}(C)$  do
6:        $Link \leftarrow 0$ 
7:       for  $v1$  in  $C[i]$  do
8:         for  $v2$  in  $C[j]$  do
9:           if ( $G.HAS-EDGE(v1, v2) == \text{True}$ ) then  $\triangleright$  inter and intra-links
10:             $Link \leftarrow Link + 1$ 
11:          end if
12:        end for
13:      end for
14:       $Com[s][t] \leftarrow Link$ 
15:    end for
16:  end for
17:  return  $Com$   $\triangleright$  The community feature matrix is  $Com$ 
18: end procedure

```

We discussed about the Label propagation method [15] for extracting communities in Chapter:1. Label propagation labels the unlabelled network dataset to derive the communities. The mathematical formulation is given as:-

$$f(t+1) = \alpha S f(t) + (1 - \alpha) Y \quad (5.1)$$

In Eqn 5.1, S is the adjacency matrix, and $f(t)$ determines the function for labeling the nodes to place in a community at time t , which is an estimator factor. α is a tunable constant between 0 and 1. When $\alpha = 0$, the first term is ignored in Eqn 5.1 and for $\alpha = 1$ the second term is ignored. Y represents the assigned initial label for each unlabelled node.

We deduce two types of possible linkages from the embedded community feature matrix in a high-dimensional format. These two types are: "intra-community linkages" and "inter-community linkages". For a community matrix C of n -dimensions, the intra-community linkages are represented by the diagonal values, whereas the inter-community linkages are the non-diagonal values. Intra-community linkages are the nodes linked within their commu-

nity. Nodes interconnected with other nodes residing in different communities exhibit inter-community links. For larger networks, the size of communities grows invariably, leading to the increment of the cost of computation of the matrix.

We performed feature engineering using T-distributed stochastic neighbour embedding(t-SNE) [59] for 2D visualization of the network. As it is known that graph objects are high-dimensional data points, therefore t-SNE is a tool that can embed these graph objects in low-dimensional data points. The major role of this tool is to convert the similarities between the graph objects to joint probabilities. It operates on minimization of the Kullback-Leibler divergence amongst the computed joint probabilities. The cost function of t-SNE is not convex, i.e., different results are obtained on different initialization of hyperparameters. We transformed our high-dimensional feature community matrix into a low-dimensional one. The hyperparameters are tuned to specific values to obtain a 2D projection of the community matrix.

5.2.3 Community Anomaly Detection Algorithms

We have taken three clustering-based outlier algorithms for detecting anomalous communities. Clustering-based outlier algorithms assign cluster labels to the communities that are grouped. The cluster's label showing certain unexpected behaviour from the rest of the clusters are identified as anomalies. Cluster-based approaches determine the density of the communities aligned with each other. Clusters exhibiting high community link-ups are influential, and thus when the complex network evolves, similar communities attach themselves to the influential one. We have taken three baseline algorithms, namely, DBSCAN, SPECTRAL, and CBLOF, and customized them as AC-DBSCAN, AC-SPECTRAL, and AC-CBLOF for the implementation of community matrix representation for detecting anomalous communities in different domains of networks.

5.2.3.1 Algorithm 1: AC-DBSCAN

AC-DBSCAN is a variation of DBSCAN: "density-based spatial clustering" technique with noise [18] which is an unsupervised learning approach that identifies unique clusters based on high point density estimation factors. It explores clusters of various shapes and sizes from

heterogeneous datasets, which contains noise as anomalies. The algorithm uses two essential hyperparameters $minPts$ and $eps(\epsilon)$. $minPts$ determines the clustered number of data points that forms a dense region. $eps(\epsilon)$ is a distance metric to locate data points in the neighborhood area. It classifies points into core points, border points, and noise points. To detect anomalous communities, we pass the community feature matrix to locate regions of high density and low density. Low-density points are the anomalies. The algorithmic steps are incorporated in Algorithm:3 for a simplified overview.

Algorithm 3 AC-DBSCAN

Input: Community feature matrix.

Output: Anomalous communities as the noise graph objects.

1. Fit the community feature matrix obtained from Algorithm 2.
 2. Select community points that are reachable based on density w.r.t ϵ and $minPts$.
 3. For an affinity community matrix, if the community C_i is a core point, then a cluster is formed.
 4. If C_i is a border point, no other points are nearer to this point, then the next community element is visited.
 5. If C_i is neither a core nor a border point, then it is labelled as a noise point, i.e., a community anomaly.
 6. Iterate from Step 2 to 5 through the community affinity matrix row-wise until all the communities are processed, and the community anomalies are obtained.
-

AC-DBSCAN incorporates all the properties of DBSCAN, such as finding arbitrarily shaped clusters and the notion of noise. AC-DBSCAN operates on two major functionalities: density-reachability and connectivity amongst the community points. The crucial part of this method is hyperparameter estimation, i.e., $minPts$ and ϵ . We arbitrarily chose the value of $minPts$ as half of the number of dimensions of the community affinity matrix and $\epsilon = 300$, w.r.t. to the particular complex networks (exhibiting sparsity). A slight variation in the hyperparameters leads to a significant change in detecting anomalous communities. Therefore, choosing proper parameter values for domain-specific complex networks is important in the AC-DBSCAN method.

5.2.3.2 Algorithm 2: AC-CBLOF

AC-CBLOF method is a variation of Cluster-based Local Outlier Factor (CBLOF) algorithm [24] for analysis of community feature matrix. It is a proximity-based approach that reports a deviation degree of the communities with regard to the cluster centroid. In the traditional approach, CBLOF labels the points into smaller and larger clusters. Anomaly scores are computed depending on the size of the cluster and the distance to the larger cluster. AC-CBLOF fits the community matrix and detects anomalous communities. A simplified overview of the AC-CBLOF method is provided in Algorithm: 4.

Algorithm 4 AC-CBLOF

Input: Community feature matrix.

Output: Anomalous communities.

1. Fit the community feature matrix obtained from Algorithm 2.
 2. Extract the clusters and sort them in ascending order.
 3. For each community point C_i , a CBLOF is assigned.
 4. If C_i is assigned to a larger cluster, $CBLOF = \text{size-of-cluster} * \text{similarity between } C_i \text{ and the cluster}$.
 5. If C_i is assigned to a smaller cluster, $CBLOF = \text{size-of-cluster} * \text{similarity between } C_i \text{ and the larger cluster which is closest}$.
 6. Return the anomalous communities whose similarity deviates significantly large from all points.
-

The concept of AC-CBLOF revolves around the size of the cluster to give a local perspective of the outlier factors. AC-CBLOF is a quite flexible in hyperparameter estimation. It is a binary outlier detector for network datasets. The anomalies are detected based on *alpha* and *beta* parameters. It works well for sparse networks.

5.2.3.3 Algorithm 3: AC-SPECTRAL

We have taken the spectral clustering algorithm [53] and customized it as AC-SPECTRAL to identify the anomalous communities. Spectral clustering uses a graph-based metric to obtain the clusters from a given affinity matrix. The baseline method projects the dataset into a spectral embedding to convert the high-dimensional data to a low-dimensional one. Though, the

spectral clustering algorithm is not commonly used for detecting anomalies. The projection is made by calculating the graph Laplacian matrix. For the computation of graph Laplacian matrix, the degree of a node is calculated as shown in Eqn. 5.2 where W_{uv} is the edge between nodes u and v . A is the adjacency matrix, and the degree matrix D is computed by the formula in Eqn. 5.3. Finally, the Laplacian matrix can be obtained by Eqn. 5.4.

$$deg_u = \sum_{v=1 | (u,v) \in E}^n W_{uv} \quad (5.2)$$

$$D_{uv} = deg_u, u = v \quad (5.3)$$

$$D_{uv} = 0, u \neq v$$

$$L = D - A \quad (5.4)$$

AC-SPECTRAL determines anomalies based on the clusters. An overview of AC-SPECTRAL is provided in Algorithm: 5.

Algorithm 5 AC-SPECTRAL

Input: Community feature matrix.

Output: Anomalous communities belonging to the cluster.

1. Preprocess the community matrix from Algorithm 2 to a similarity graph.
 2. Project the community points C_i into a lower dimensional subspace by computing the graph Laplacian matrix from Eqn. 5.4
 3. C_i are assembled into specific clusters.
 4. Clusters are sorted in descending order.
 5. The lowest cluster size is taken as the set of anomalous communities.
-

AC-SPECTRAL is effective for sparse complex networks. In this, the important hyper-parameter is $n - clusters$ i.e., for choosing the appropriate number of clusters. We arbitrarily chose $n - clusters = 2$ to obtain the normal and anomalous clusters. The *affinity* parameter is set to the nearest-neighbors distance.

5.3 Network Data Statistics

We have taken seven benchmark complex networks (discussed in Chapter 3) to detect the domain-specific anomalous communities, which are static, undirected, and unweighted. Table 5.1 depicts the seven benchmark complex network datasets description. The largest network is the Function-function network, whereas the smallest one is the Social-hamsterster network.

Table 5.1: Statistics of complex networks

Network	Domain	Nodes	Edges
Drug-target	Biological	7,341	15,138
Function-function	Biological	46,027	1,06,510
Astro-Physics collaboration	Collaboration	18,772	1,98,110
Condense-matter collaboration	Collaboration	23,133	93,497
Facebook-pages-company	Social	14,113	52,310
Facebook-pages-tvshow	Social	3,892	17,262
Social-hamsterster	Social	2,426	16,630

5.4 Evaluation Metrics

To evaluate the algorithms, we have chosen two of the well-studied evaluation metrics to determine how the clusters are formed and whether the anomalies are detected correctly or not. These metrics are useful for analyzing the performance of unsupervised methods.

1. **Silhouette score**- The Silhouette score [56] is a measurement to compute the correctness of the clusters formed with respect to the clustering technique. It gives a value between -1 to 1. If the Silhouette score value is 1, the clusters are dense and distinctively separated. If it is 0, then the means of the clusters are not so dense, and the distance between them is insignificant; if the value is -1, then the cluster centroids are assigned wrongly.

$$S = (b_1 - a_1) / \max(a_1, b_1) \quad (5.5)$$

Eqn. 5.5 shows the calculation of the Silhouette coefficient, where a_1 is the average

"intra-cluster distance", i.e., the average distance computed for each point in the cluster, and b_1 gives the "inter-cluster distance" (the average distances amongst all clusters). This score works quite effectively for unlabelled datasets.

2. **Calinski-harabasz index (CH)** score is also known as "variance ratio criterion" [9].

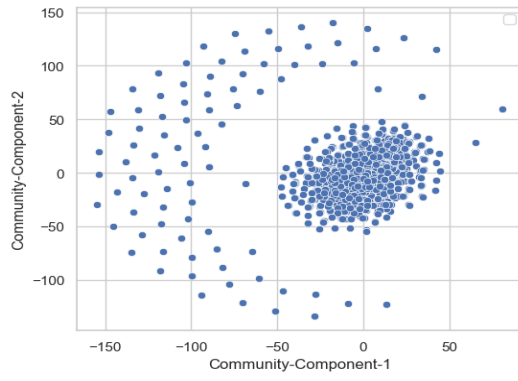
It can be used to validate the methods whose ground truth labels are not given, i.e., unlabelled datasets. *CH* index computes the cluster's cohesiveness (similarity of points to it's cluster) compared with other clusters in separation. Cohesion indicates the distance of the points from the centroid of its cluster, and separation is the distance of all the cluster centroids from the global centroid. *CH* index for P number of clusters on the dataset points $DP=[d_1, d_2, \dots, d_N]$ is computed as:-

$$\begin{aligned}
 Separation &= \frac{\sum_{p=1}^P n_p ||c_p - c||^2}{P - 1} \\
 Cohesion &= \frac{\sum_{p=1}^P \sum_{i=1}^{n_p} ||d_i - c_p||^2}{N - P} \\
 CH &= \frac{Separation}{Cohesion}
 \end{aligned} \tag{5.6}$$

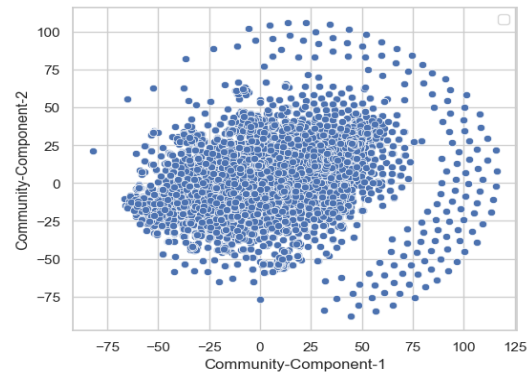
In Eqn. 5.6, n_p is the set of data points and c_p is centroid of p^{th} cluster. The global centroid is c and N is the total number of points, respectively. High values of *CH* indicate the clusters are formed correctly. The inter-cluster distance needs to be large while the intra-cluster distance should be small, in order to achieve well-defined clusters.

5.5 Experimental Results

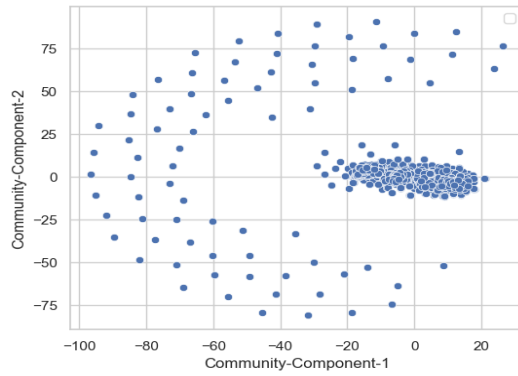
We demonstrated the experiments based on the three algorithms for the seven benchmark network data sets which are taken for our empirical analysis. Also, we included t-SNE visualizations of the network datasets and provided a detailed comparative study of the three algorithms based on the Silhouette score and Calinski-harabasz (*CH*) index.



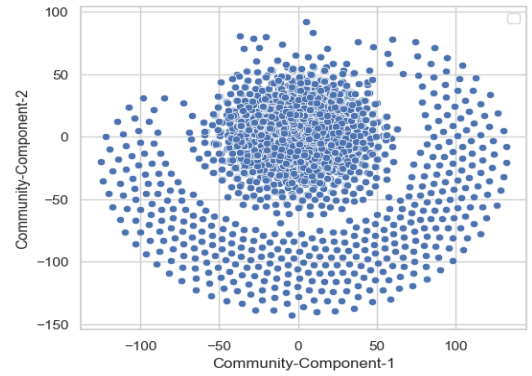
(a) Drug-target



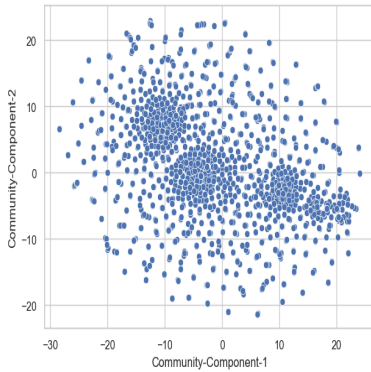
(b) Function-function



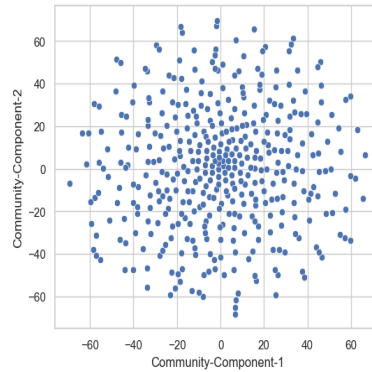
(c) Astro Physics Collaboration



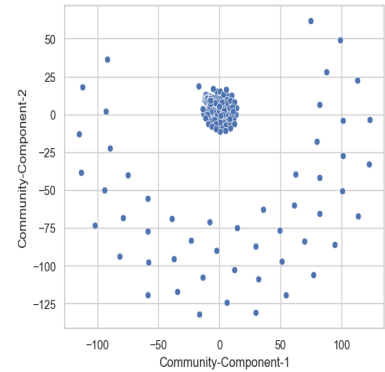
(d) Condense Matter Collaboration



(e) Facebook-pages-company



(f) Facebook-pages-tvshow



(g) Social-hamsterster

Figure 5.2: t-SNE 2D Visualization of the respective Communities in the Complex Networks

Fig. 5.2 depicts the t-SNE visualization of community points in 2D form. The t-SNE visualization shows the scattering of the community points with respect to the joint probabilities of community-component1 and community-component2.

Table 5.2: Evaluation of algorithms based on identified anomalous communities in complex networks

Network	Communities	AC-DBSCAN	AC-CBLOF	AC-SPECTRAL
Drug-target	983	17	99	342
Function-function	5001	14	500	15
Astro-Physics collaboration	1040	42	104	567
Condense-matter collaboration	3195	51	320	189
Facebook-pages-company	1308	30	131	154
Facebook-pages-tvshow	408	18	41	79
Social-hamsterster	256	4	26	85

Table 5.2 depicts the number of communities detected by the Label-propagation method and anomalous communities detected by the respective algorithms. AC-SPECTRAL detects the highest number of anomalous communities in most complex networks. In the case of Function-function and Condense-matter collaboration networks, AC-CBLOF generates the highest number of community anomalies.

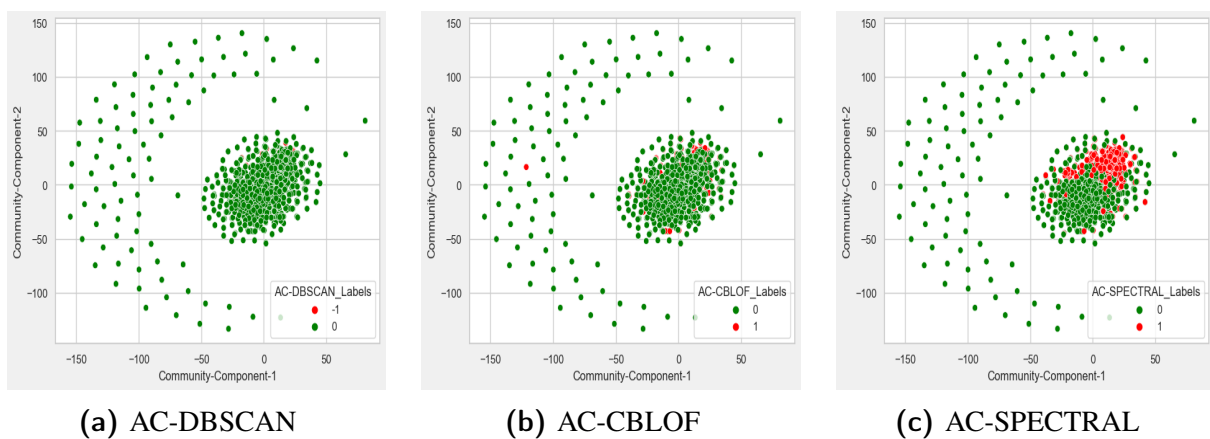


Figure 5.3: Snapshots of anomalous/normal communities of Drug-Target Network

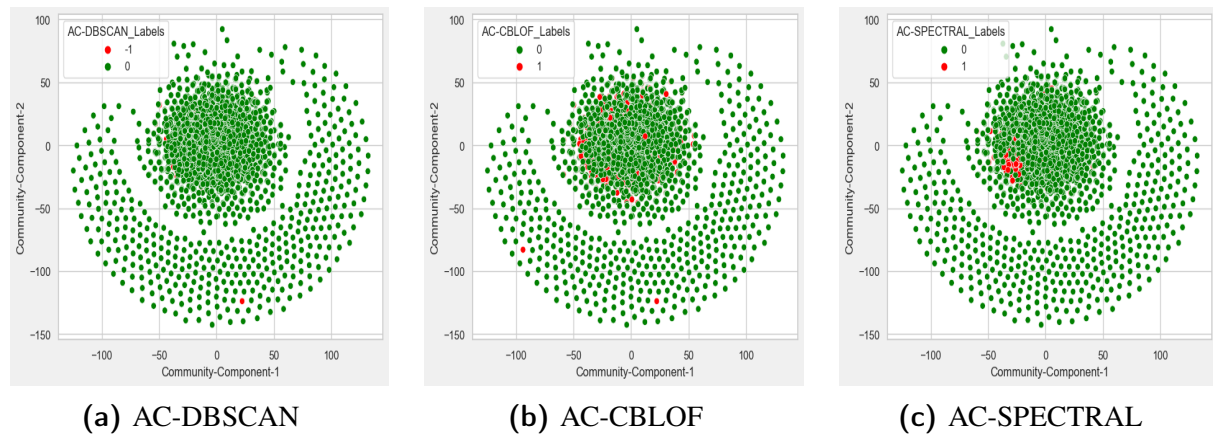


Figure 5.4: Snapshots of anomalous/normal communities of Condense-matter collaboration Network

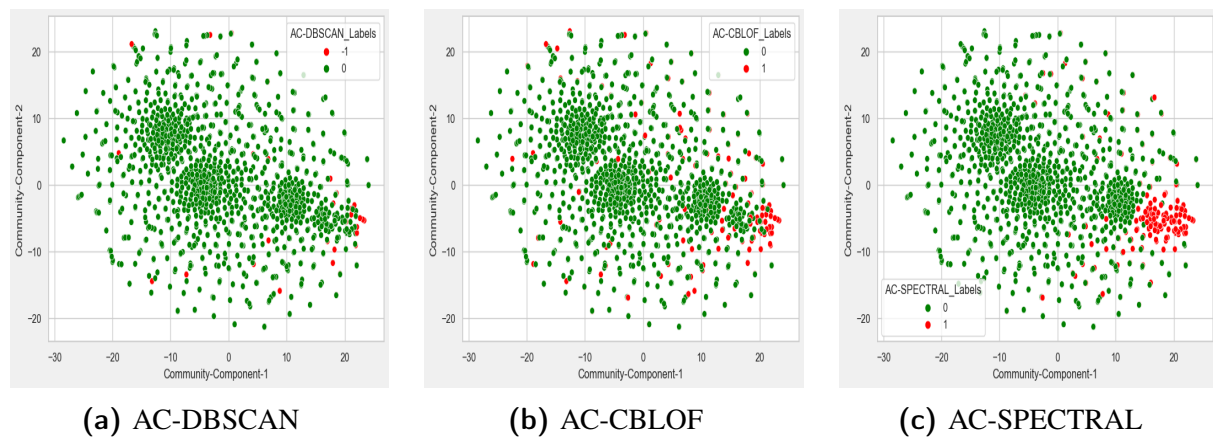


Figure 5.5: Snapshots of anomalous/normal communities of Facebook-pages-company Network

We visualized Figs. 5.3, 5.4, and 5.5 (one each for three domains of network) that demonstrate the anomalous communities detected by the three algorithms with respect to the target complex networks. AC-DBSCAN generates two cluster labels: -1: "Anomalous communities" and 0: "Normal communities". In the case of AC-CBLOF and AC-SPECTRAL, the cluster labels are defined as 1: "Anomalous communities" and 0: "Normal communities". The red colored points depict the anomalous communities. Further, it is seen, that the networks with a large number of communities exhibit compactness amongst the communities.

Drug-target networks find the communities of the interaction of drugs and genes that might not be feasible after a period. In a similar fashion, the Condense-matter collaboration network detects anomalous communities based on the communities of authors that have less collaboration with other community authors. In the context of the Facebook-pages-company networks,

the communities of pages that are less visited or the mutual likes are lesser detected. During the evolution of complex networks, such types of communities disappear. Therefore, it is significant to determine such communities.

Table 5.3: Evaluation of algorithms based on Silhouette Score in complex networks

Network	AC-DBSCAN	AC-CBLOF	AC-SPECTRAL
Drug-target	0.95	0.78	0.25
Function-function	0.99	0.79	-0.73
Astro-Physics collaboration	0.92	0.81	-0.05
Condense-matter collaboration	0.91	0.75	0.41
Facebook-pages-company	0.94	0.80	0.72
Facebook-pages-tvshow	0.88	0.79	0.47
Social-hamsterster	0.96	0.80	0.35

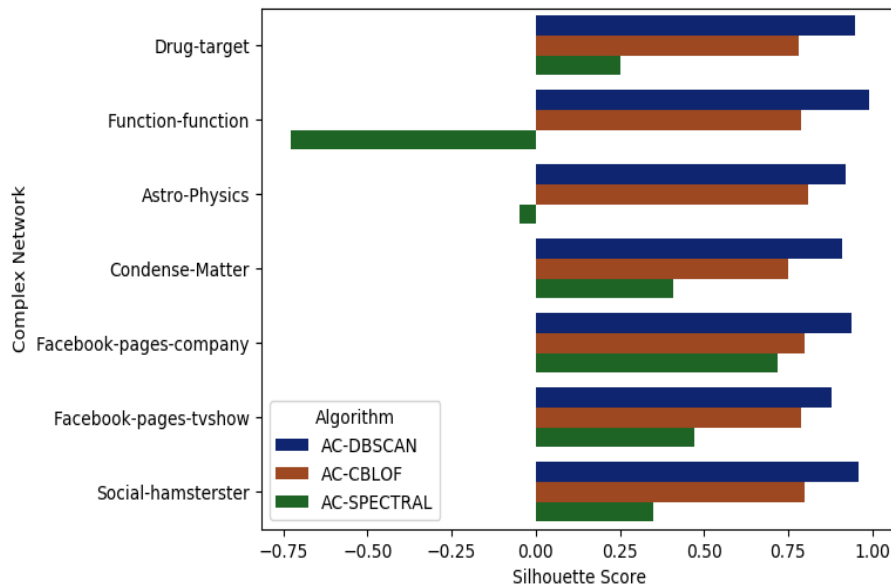


Figure 5.6: Comparative Evaluation of Algorithms based on Silhouette score

In Table 5.3, we computed the Silhouette score of each algorithm. AC-DBSCAN outperforms the other two algorithms, depicting the correctness of the generated clusters. AC-SPECTRAL poorly detects the clusters of Function-function and Astro-Physics collaboration

networks, as their Silhouette score is negative. AC-DBSCAN effectively detects the correct anomalies. We visualized the performance of the algorithms with respect to the Silhouette score in Fig. 5.6.

Table 5.4: Evaluation of algorithms based on CH index in complex networks

Network	AC-DBSCAN	AC-CBLOF	AC-SPECTRAL
Drug-target	78.53	12.49	2.83
Function-function	399.00	9.80	0.03
Astro-Physics collaboration	26.17	9.88	0.92
Condense-matter collaboration	93.71	20.31	9.39
Facebook-pages-company	50.92	11.24	9.35
Facebook-pages-tvshow	23.29	9.59	4.13
Social-hamsterster	86.06	9.78	2.21

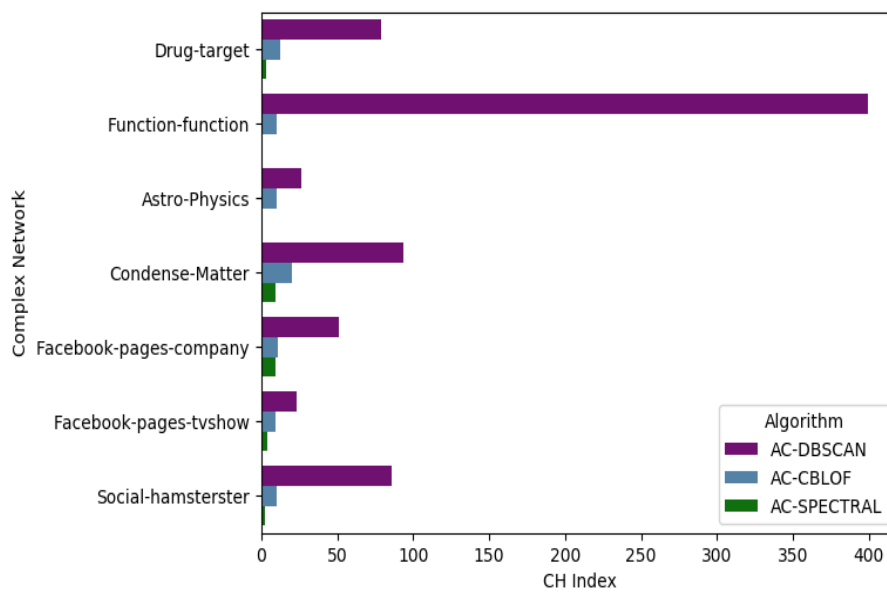


Figure 5.7: Comparative Evaluation of Algorithms based on CH Index

In Table 5.4, the CH Index is calculated for each algorithm. AC-DBSCAN shows remarkable performance in terms of the CH Index, as it gives a much larger value. The performance of AC-CBLOF is moderate. We have shown the comparative evaluation of the algorithms in Fig 5.7 by a grouped bar chart.

5.6 Conclusion

In this chapter, we designed a community embedding technique to convert complex network community information into a community affinity matrix. Then, we performed outlier detection by customizing three clustering algorithms to understand the nature of communities that fall into either of the two classes normal or anomalous. Experimental results for evaluating the algorithms based on Silhouette score and CH index show that AC-DBSCAN performs more efficiently than AC-CBLOF and AC-SPECTRAL. The highest number of community anomalies is detected by AC-SPECTRAL, but AC-DBSCAN generates correct clusters and identifies the correct set of anomalous communities. In the case of Function-function and Astro-Physics collaboration networks, AC-spectral performance is quite poor. For future work, the performance of AC-SPECTRAL can be improved by choosing a different number of clusters to identify the correct pattern of anomalous communities.

6

Conclusion and Future Work

6.1 Work Summary

In this dissertation, we studied topological and structural anomalies present in a complex network. Our work is divided in detecting two types of anomalies: Node anomalies and Community anomalies. We addressed the *Research question 1* in Chapter 4 (Section 4.2) and *Research question 2* in Chapter 5 (Section 5.2) by formulating problem definitions for each respectively. We explored and visualized the characteristics of complex networks with respect to their domains. Summarizing our works:-

1. We designed a localized community-based anomaly detection algorithm to identify node anomalies inside communities on the Synthetic and Zachary's Karate club networks. In this algorithm, the communities are partitioned by a community detection method, and

then two well-known centrality metrics are applied to get different sets of node anomalies within communities. The goal is to get an insight into the variation of node anomalies for different centrality measurements.

2. For understanding the community attributes of a network, we devised a community feature matrix that behaves like an adjacency matrix for communities. The community matrix is dissected to understand its patterns (*"intra-community and inter-community links"*). The complex network is preprocessed into a community affinity matrix to analyze the community anomalies effectively.
3. We customized three of the unsupervised clustering anomaly detection algorithms to work on three different domains of complex networks. These algorithms detect community anomalies by labelling them into clusters. We evaluated the performance based on certain metrics to determine the correctness of cluster formation. Lastly, we provided visualizations of *"normal/anomalous"* communities on the chosen set of complex networks.

6.2 Directions for Future Work

In previous chapters, we systematically outlined the related work and proposed algorithms for node-level and community-level anomaly detection. Here, we provide a few recommendations for future direction:-

- For node-level anomaly detection, we proposed a Localized community-based node anomaly detection algorithm in Algorithm 1 applying the Louvain algorithm for partitioning the network. Other overlapping and non-overlapping community detection algorithms can be used that can generate hidden node anomalies, and their performance can be assessed.
- The community embedding matrix proposed in Algorithm 2 can be modified as the network scales in size, including only the community feature vectors that are of significance in the network using deep learning approaches.

- Another future work can be to devise multi-view clustering outlier detection methods to obtain multiple network-level anomalies in different views of the network. AC-SPECTRAL algorithm performs poorly for certain networks like function-function network and Astro-Physics collaboration network. Therefore, this algorithm can be enhanced by tuning the number of clusters generated and correctly identifying the anomalies for application-specific networks.
- Proper evaluation metrics for anomalous network pattern recognition by using unsupervised algorithms is a research area that is still unexplored. Strong evidence of correctly identified anomalies is required. The metrics should be flexible enough to interpret the anomalies at each network level. The evaluation measures should justify with solid explanations whether a network object is an anomaly or not.
- Our work focused on identifying node and community anomalies on unlabelled static networks. Motivated by recent advancements, it is suggestive to use dynamic network settings where the network evolves with time. The algorithms can be modified to provide snapshots of node and community anomalies for dynamic networks.

List of Publication

In Conference Proceedings

1. Trishita Mukherjee and Rajeev Kumar. Localized Community Based Node Anomalies in Complex Networks. In Proc. *11th Int. Conf. Soft Computing for Problem Solving* (SocProS). May 2022. Springer

References

- [1] Leman Akoglu, Mary McGlohon, and Christos Faloutsos. Anomaly detection in large graphs. Technical Report CMU-CS-09-173, CMU, 2009.
- [2] Leman Akoglu, Mary McGlohon, and Christos Faloutsos. Oddball: Spotting anomalies in weighted graphs. In *Proc. Pacific-Asia Conf. Knowledge Discovery and Data Mining*, pages 410–421. Springer, 2010.
- [3] Leman Akoglu, Hanghang Tong, and Danai Koutra. Graph-based anomaly detection and description: a survey. *Data Mining & Knowledge Discovery*, 29(3):626–688, 2015.
- [4] Brigham S Anderson, Carter Butts, and Kathleen Carley. The interaction of size and density with graph-level indices. *Social Networks*, 21(3):239–267, 1999.
- [5] Albert-László Barabási. Network science. *Philosophical Trans. Royal Society A: Mathematical, Physical and Engineering Sciences*, 371(1987):20120375, 2013.
- [6] Yoshua Bengio, Pascal Lamblin, Dan Popovici, and Hugo Larochelle. Greedy layer-wise training of deep networks. *Advances in Neural Information Processing Systems*, 19, 2006.
- [7] Christopher M Bishop. Mixture models and the em algorithm. *Microsoft Research, Cambridge*, 2006.
- [8] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. Fast unfolding of communities in large networks. *Journal Statistical Mechanics: Theory & Experiment*, 2008(10):P10008, 2008.
- [9] Tadeusz Caliński and Jerzy Harabasz. A dendrite method for cluster analysis. *Communications in Statistics Theory and Methods*, 3(1):1–27, 1974.
- [10] Sandro Cavallari, Erik Cambria, Hongyun Cai, Kevin Chen-Chuan Chang, and Vincent W Zheng. Embedding both finite and infinite communities on graphs [application notes]. *IEEE Computational Intelligence Magazine*, 14(3):39–50, 2019.
- [11] Varun Chandola, Arindam Banerjee, and Vipin Kumar. Anomaly detection: A survey. *ACM Computing Surveys*, 41(3):1–58, 2009.
- [12] Lijun Chang, Wei Li, Lu Qin, Wenjie Zhang, and Shiyu Yang. pSCAN: fast and exact structural graph clustering. *IEEE Trans. Knowledge and Data Engineering*, 29(2):387–401, 2017.
- [13] Liangchen Chen, Shu Gao, and Baoxu Liu. An improved density peaks clustering algorithm based on grid screening and mutual neighborhood degree for network anomaly detection. *Scientific Reports*, 12(1):1–14, 2022.

- [14] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proc. 22nd ACM SIGKDD Int Conf. Knowledge Discovery and Data Mining*, pages 785–794, 2016.
- [15] Gennaro Cordasco and Luisa Gargano. Community detection via semi-synchronous label propagation algorithms. In *IEEE Int. workshop Business Applications of Social Network Analysis (BASNA)*, pages 1–8. IEEE, 2010.
- [16] Alexandre d’Aspremont, Laurent Ghaoui, Michael Jordan, and Gert Lanckriet. A direct formulation for sparse pca using semidefinite programming. *Advances in Neural Information Processing Systems*, 17, 2004.
- [17] Sergey N Dorogovtsev, Jos FF Mendes, and Alexander N Samukhin. Size-dependent degree distribution of a scale-free growing network. *Physical Review E*, 63(6):062101, 2001.
- [18] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proc. KDD*, volume 96, pages 226–231, 1996.
- [19] Leonhard Euler. The seven bridges of königsberg. *The World of Mathematics*, 1:573–580, 1956.
- [20] Rodrigo Francisquini, Ana Carolina Lorena, and Mariá CV Nascimento. Community-based anomaly detection using spectral graph filtering. *arXiv preprint arXiv:2201.09936*, 2022.
- [21] Linton C Freeman. Centrality in social networks conceptual clarification. *Social Networks*, 1(3):215–239, 1978.
- [22] Jing Gao, Feng Liang, Wei Fan, Chi Wang, Yizhou Sun, and Jiawei Han. On community outliers and their efficient detection in information networks. In *Proc. 16th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, pages 813–822, 2010.
- [23] Michelle Girvan and Mark EJ Newman. Community structure in social and biological networks. *Proc. National Academy of Sciences*, 99(12):7821–7826, 2002.
- [24] Zengyou He, Xiaofei Xu, and Shengchun Deng. Discovering cluster-based local outliers. *Pattern Recognition Letters*, 24(9-10):1641–1650, 2003.
- [25] Thomas J Helling, Johannes C Scholtes, and Frank W Takes. A community-aware approach for identifying node anomalies in complex networks. In *Proc. Int. Conf. Complex Networks & Applications*, pages 244–255. Springer, 2018.
- [26] Keith Henderson, Brian Gallagher, Lei Li, Leman Akoglu, Tina Eliassi-Rad, Hanghang Tong, and Christos Faloutsos. It’s who you know: graph mining using recursive structural features. In *Proc. 17th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, pages 663–671, 2011.
- [27] Tsuyoshi Idé, Aurelie C Lozano, Naoki Abe, and Yan Liu. Proximity-based anomaly detection using sparse structure learning. In *Proc. SIAM Int. Conf. Data Mining*, pages 97–108. SIAM, 2009.

- [28] Leo Katz. A new status index derived from sociometric analysis. *Psychometrika*, 18(1):39–43, 1953.
- [29] Jon Kleinberg. The small-world phenomenon: An algorithmic perspective. In *Proc. 32nd Annual ACM Symposium Theory of Computing*, pages 163–170, 2000.
- [30] Rajeev Kumar and Nilanjan Banerjee. Multiobjective network topology design. *Applied Soft Computing*, 11(8):5120–5128, 2011.
- [31] Shay Lapid, Dima Kagan, and Michael Fire. Co-membership-based generic anomalous communities detection. *arXiv preprint arXiv:2203.16246*, 2022.
- [32] Vito Latora and Massimo Marchiori. Efficient behavior of small-world networks. *Physical Review Letters*, 87(19):198701, 2001.
- [33] Jure Leskovec, Jon Kleinberg, and Christos Faloutsos. Graph evolution: Densification and shrinking diameters. *ACM Trans. Knowledge Discovery from Data (TKDD)*, 1(1):2–es, 2007.
- [34] Jure Leskovec and Rok Sosič. Snap: A general-purpose network analysis and graph-mining library. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 8(1):1–20, 2016.
- [35] Fanzhen Liu, Zhao Li, Baokun Wang, Jia Wu, Jian Yang, Jiaming Huang, Yiqing Zhang, Weiqiang Wang, Shan Xue, Surya Nepal, et al. eRiskCom: an e-commerce risky community detection platform. *The VLDB Journal*, pages 1–17, 2022.
- [36] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. Isolation forest. In *Proc. 8th Int. Conf. Data Mining*, pages 413–422. IEEE, 2008.
- [37] Jalil Jabari Lotf, Mohammad Abdollahi Azgomi, and Mohammad Reza Ebrahimi Dishabi. An improved influence maximization method for social networks based on genetic algorithm. *Physica A: Statistical Mechanics & its Applications*, 586:126480, 2022.
- [38] Xiaoxiao Ma, Jia Wu, Shan Xue, Jian Yang, Chuan Zhou, Quan Z Sheng, Hui Xiong, and Leman Akoglu. A comprehensive survey on graph anomaly detection with deep learning. *IEEE Trans. Knowledge and Data Engineering*, 2021.
- [39] Basim Mahmood and Mafaz Alanezi. Structural-spectral-based approach for anomaly detection in social networks. *Int. Journal Computing and Digital Systems*, 10(1):343–351, 2021.
- [40] Sagar Maheshwari Marinka Zitnik, Rok Sosi and Jure Leskovec. Biosnap datasets: Stanford biomedical network dataset collection, Aug 2018.
- [41] Benjamin A Miller, Michelle S Beard, Patrick J Wolfe, and Nadya T Bliss. A spectral framework for anomalous subgraph detection. *IEEE Trans. Signal Processing*, 63(16):4191–4206, 2015.
- [42] Emmanuel Müller, Patricia Iglesias Sánchez, Yvonne Mülle, and Klemens Böhm. Ranking outlier nodes in subspaces of attributed graphs. In *Proc. IEEE 29th Int. Conf. Data Engineering Workshops (ICDEW)*, pages 216–222. IEEE, 2013.

- [43] Mark EJ Newman. The structure and function of complex networks. *SIAM Review*, 45(2):167–256, 2003.
- [44] Erdős Paul and Rényi Alfréd. On random graphs I. *Publicationes Mathematicae (Debrecen)*, 6:290–297, 1959.
- [45] Bryan Perozzi and Leman Akoglu. Discovering communities and anomalies in attributed graphs: Interactive visual exploration and summarization. *ACM Trans. Knowledge Discovery from Data (TKDD)*, 12(2):1–40, 2018.
- [46] Tahereh Pourhabibi, Kok-Leong Ong, Booi H Kam, and Yee Ling Boo. Fraud detection: A systematic literature review of graph-based anomaly detection approaches. *Decision Support Systems*, 133:113303, 2020.
- [47] Günther R Raidl. A unified view on hybrid metaheuristics. In *Proc. Int. Workshop on Hybrid Metaheuristics*, pages 1–12. Springer, 2006.
- [48] Krishna Raj PM, Ankith Mohan, and KG Srinivasa. Small world phenomena. In *Practical Social Network Analysis with Python*, pages 57–85. Springer, 2018.
- [49] Ryan Rossi and Nesreen Ahmed. The network data repository with interactive graph analytics and visualization. In *Proc. 29th AAAI Conf. Artificial Intelligence*, 2015.
- [50] Martin Rosvall and Carl T Bergstrom. Maps of random walks on complex networks reveal community structure. *Proc. National Academy Sciences*, 105(4):1118–1123, 2008.
- [51] Benedek Rozemberczki, Ryan Davies, Rik Sarkar, and Charles Sutton. Gemsec: Graph embedding with self clustering. In *Proc. IEEE/ACM Int. Conf. Advances in Social Networks Analysis and Mining*, pages 65–72, 2019.
- [52] Soma Saha, Rajeev Kumar, and Gyan Baboo. Characterization of graph properties for improved Pareto fronts using heuristics and EA for bi-objective graph coloring problem. *Applied Soft Computing*, 13(5):2812–2822, 2013.
- [53] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.
- [54] Hiroaki Shiokawa, Yasuhiro Fujiwara, and Makoto Onizuka. Scan++ efficient algorithm for finding clusters, hubs and outliers on large-scale graphs. *Proc. VLDB Endowment*, 8(11):1178–1189, 2015.
- [55] David B Skillicorn. Detecting anomalies in graphs. In *Proc. IEEE Intelligence and Security Informatics*, pages 209–216. IEEE, 2007.
- [56] Anja Struyf, Mia Hubert, and Peter Rousseeuw. Clustering in an object-oriented environment. *Journal of Statistical Software*, 1:1–30, 1997.
- [57] Michael PH Stumpf and Carsten Wiuf. Sampling properties of random graphs: the degree distribution. *Physical Review E*, 72(3):036118, 2005.
- [58] Camila PS Tautenhain and Mariá CV Nascimento. SpecRp: A spectral-based community embedding algorithm. *Machine Learning with Applications*, page 100326, 2022.

- [59] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal Machine Learning Research*, 9:2579–2605, 11 2008.
- [60] Stijn vanDongen. A cluster algorithm for graphs. *Report Information Systems*, 10(R0010), 2000.
- [61] Dmitry Vengertsev and Hemal Thakkar. Anomaly detection in graph: unsupervised learning, graph-based features and deep architecture. Technical report, Dept. Comput. Sci., Stanford Univ., Stanford, CA, USA, 2015.
- [62] Duncan J Watts. Networks, dynamics, and the small-world phenomenon. *American Journal of Sociology*, 105(2):493–527, 1999.
- [63] Duncan J Watts and Steven H Strogatz. Collective dynamics of ‘small-world’ networks. *Nature*, 393(6684):440–442, 1998.
- [64] Xiaowei Xu, Nurcan Yuruk, Zhidan Feng, and Thomas AJ Schweiger. Scan: a structural clustering algorithm for networks. In *Proc. 13th ACM SIGKDD Int. Conf. Knowledge Discovery & Data Mining*, pages 824–833, 2007.
- [65] Zhe Yao, Philip Mark, and Michael Rabbat. Anomaly detection using proximity graph and pagerank algorithm. *IEEE Trans. Information Forensics and Security*, 7(4):1288–1300, 2012.
- [66] Vincent W Zheng, Sandro Cavallari, Hongyun Cai, Kevin Chen-Chuan Chang, and Erik Cambria. From node embedding to community embedding. *arXiv preprint arXiv:1610.09950*, 2016.