ANOMALY DETECTION IN ONLINE SOCIAL NETWORKS: USING DATA-MINING TECHNIQUES AND FUZZY LOGIC

Reza Hassanzadeh
BEng., MSc

Submitted in fulfilment of the requirements for the degree of
Doctor of Philosophy

School of Electrical Engineering and Computer Science
Faculty of Science and Engineering
Queensland University of Technology
November 2014
Keywords

Anomaly Detection
Clustering-Based Model
Distribution-Based Model
Expectation Maximization
Fuzzy Clustering
Fuzzy Inference Engine
Fuzzy Logic
Gaussian Mixture Model
Graph-Based Anomaly Detection
Membership Function
Online Social Networks
Orthogonal Projection
Power-Law Regression Model
Predictive Model
Abstract

The Online Social Networks (OSNs), which captures the structure and dynamics of person-to-person and person-to-technology interaction, is being used for various purposes such as business, education, telemarketing, medical, entertainment. This technology also opens the door for unlawful activities. Detecting anomalies, in this new perspective of social life that articulates and reflects the off-line relationships, is an important factor as they could be a sign of a significant problem or carrying useful information for the analyser.

Two types of data can be inferred from OSNs: (1) the behavioural data that considers the dynamic usage behaviour of users; and (2) the structural data that considers the structure of the networks. These two types of data can be modelled by graph theory in order to extract meaningful features which can be analysed by appropriate techniques. Existing anomaly detection techniques using graph modelling are limited due to issues such as time and computational complexity, low accuracy, missing value, privacy, and lack of labelled datasets. To overcome the existing limitations, we present various hybrid methods that utilise different types of structural input features and techniques.

We present these approaches within a multi-layered framework which provides the full requirements needed for finding anomalies in online social networks data graph, including modelling, algorithms, labelling, and evaluation.
In the first layer of the proposed framework, we model an online social network with graph theory and compute the various graph features for the nodes in the graph. The second layer of the framework includes our methods which tackle the problem of anomaly detection in online social networks from different angles: distance-based, distribution-based, and clustering-based. We use fuzzy logic to define the boundaries of the anomalies as they can be treated as a multiple-valued logic problem in which we have a degree of truth rather than as only two possible values (normal or abnormal). The third layer of our framework is for evaluating the proposed methods using three different and popular OSNs.

The experiment results show in general that (1) a combination of orthogonal projection and a clustering algorithm can improve the accuracy of the distance-based method, and (2) in terms of increasing accuracy, using fuzzy based clustering shows better results compared to using hard portioning ones. The reason behind the outperformance of the proposed fuzzy-based clustering method is that instances can be members of more than one cluster, with different levels of certainty. This contrasts with hard partitioning algorithms such as k-means in which any instances can belong to only a single cluster. This means that the fuzzy nature of friendship relations is lost during clustering, which affects the quality of detecting anomalies within the OSNs data. Moreover, experiments show the distribution-based method outperforms the accuracy among all other methods, because of the ability to find the natural relationship between instances with the expectation-maximization algorithm and describe the fuzziness of the instances with fuzzy logic. The evaluation results are consistent among the three different real-life datasets.
Table of Contents

CHAPTER 1: INTRODUCTION ........................................................................................................... 1
  1.1 MOTIVATION ......................................................................................................................... 1
  1.2 PROBLEM STATEMENT .......................................................................................................... 8
  1.3 RESEARCH QUESTIONS AND OBJECTIVE ........................................................................ 11
  1.4 CONTRIBUTION TO THE BODY OF KNOWLEDGE ............................................................ 12
  1.5 OUTLINE OF THE THESIS .................................................................................................... 16

CHAPTER 2: LITERATURE REVIEW ............................................................................................... 19
  2.1 ANOMALY DETECTION ......................................................................................................... 19
    2.1.1 Anomaly Detection Techniques ...................................................................................... 20
      2.1.1.1 Supervised Anomaly Detection ............................................................................... 22
      2.1.1.2 Semi-supervised Anomaly Detection Techniques .................................................... 26
      2.1.1.3 Unsupervised Anomaly Detection Techniques ....................................................... 26
    2.1.2 Reporting Anomaly Detection ....................................................................................... 28
    2.1.3 Summary ......................................................................................................................... 28
  2.2 ANOMALY DETECTION AND ONLINE SOCIAL NETWORKS .................................................. 29
    2.2.1 Online behaviours.......................................................................................................... 30
    2.2.2 Type Of Anomalies ........................................................................................................ 34
    2.2.3 OSNs Anomaly Detection Challenges .......................................................................... 36
      2.2.3.1 Labelled Dataset .................................................................................................... 37
    2.2.4 Anomaly Detection In Graph-Based Data .................................................................... 38
      2.2.4.1 Anomaly In Static Large Data Graph .................................................................... 40
    2.2.5 Summary ......................................................................................................................... 48
  2.3 CLUSTERING ALGORITHMS ............................................................................................... 49
    2.3.1 Semi-Unsupervised Clustering ...................................................................................... 51
    2.3.2 Unsupervised Clustering ............................................................................................... 52
    2.3.3 Fuzzy Clustering ............................................................................................................ 52
    2.3.4 Summary ......................................................................................................................... 53
  2.4 FUZZY LOGIC ....................................................................................................................... 54
  2.5 RESEARCH GAP .................................................................................................................. 56
  2.6 SUMMARY ............................................................................................................................. 58

CHAPTER 3: MULTI-LAYER FRAMEWORK ...................................................................................... 61
  3.1 FRAMEWORK OVERVIEW ................................................................................................... 62
  3.2 LAYER-ONE: PRE-PROCESSING, MODELLING, IDENTIFYING Egonets, AND SUPER-Egonets .... 64
    3.2.1 Modelling OSNs Using Graph Theory .......................................................................... 65
    3.2.2 Features Extraction ....................................................................................................... 67
      3.1.1.1 Online Social Network Characteristics ................................................................. 69
      3.1.1.2 Centrality Metrics .................................................................................................. 72
      3.1.1.3 Community Detection ......................................................................................... 77
      3.1.1.4 Cliqueness and Starness ..................................................................................... 79
  3.3 LAYER-TWO OVERVIEW: ANOMALY DETECTION ALGORITHMS .................................. 79
    3.3.1 Layer-Two (a): Distance-Based Anomaly Detection Using Graph Metrics ............ 81
    3.3.2 Layer-Two (b): Distribution-Based (Statistical-Based) Anomaly Detection Using Graph Metrics ........................................................................................................... 82
    3.3.3 Layer-Two (c): Clustering-Based Approach Using Graph Metrics ....................... 83
CHAPTER 4: ANOMALY DETECTION METHODS ................................................. 97

4.1 INTRODUCTION ......................................................................................... 97
4.2 DISTANCE-BASED APPROACH USING GRAPH METRICS AND ORTHOGONAL PROJECTION ............................................................... 100
  4.2.1 Method Overview .............................................................................. 101
  4.2.2 Input–Computing Graph Metrics ............................................................. 102
  4.2.3 Compute Regression Model ................................................................... 104
  4.2.4 Computing The Distance From Regression Model ................................. 106
  4.2.5 Compute Orthogonal Projection .............................................................. 107
  4.2.6 Fuzzy C-Means (FCM) Clustering ......................................................... 110
4.3 DISTRIBUTION-BASED (STATISTICAL-BASED) APPROACH USING GRAPH METRICS ................................................................. 112
  4.3.1 Method Overview .............................................................................. 113
  4.3.2 Input–Local Graph Properties ............................................................... 115
  4.3.3 Clustering Preliminary Anomaly Score With Unsupervised Learning .......... 115
  4.3.4 Classification Using Fuzzy Inference Engine ........................................... 119
4.4 CLUSTERING-BASED ANOMALY DETECTION IN ONLINE-SOCIAL-NETWORK GRAPHS .......................................................... 124
  4.4.1 Method Overview .............................................................................. 126
  4.4.2 Input To Algorithm ............................................................................. 128
  4.4.3 Finding Cluster Number Using GMM-EM ............................................ 128
  4.4.4 Clustering Using Fuzzy C-means (FCM) ................................................ 129
  4.4.5 Representing Clusters With Fuzzy Inference Engine ............................. 131
4.5 SUMMARY .................................................................................................. 133

CHAPTER 5: EXPERIMENTS AND DISCUSSIONS ............................................. 137

5.1 FRAMEWORK ........................................................................................... 138
  5.1.1 Distance-Based Approach ................................................................. 140
    5.1.1.1 Experiment Design ...................................................................... 140
    5.1.1.2 Power-Law Regression Method Results ......................................... 143
    5.1.1.3 Orthogonal Projection Method Results ........................................... 144
  5.1.2 Distribution-Based Approach .............................................................. 145
    5.1.2.1 Experiment Design ...................................................................... 145
    5.1.2.2 Distribution Method Results ......................................................... 148
  5.1.3 Clustering-Based Approach ................................................................. 151
    5.1.3.1 Experiment Design ...................................................................... 152
    5.1.3.2 Clustering Method Results ........................................................... 153
5.2 EXPERIMENT RESULTS DISCUSSION ..................................................... 155
  5.2.1 Power-Law Degree And Normal Instances Distribution ......................... 156
  5.2.2 STRENGTHS AND SHORTCOMINGS OF EACH METHOD ................. 161
    5.2.2.1 Power-Law Regression Method ..................................................... 161
    5.2.2.2 Orthogonal Projection Method ....................................................... 161
    5.2.2.3 Distribution Method ..................................................................... 162
    5.2.2.4 Clustering Method ......................................................................... 163
  5.2.3 PERFORMANCE COMPARISONS ........................................................ 163
    5.2.3.1 Dealing With Anomaly Detection Challenges ................................ 164
    5.2.3.2 Distance-Based Method vs Clustering-Based Method ....................... 165
    5.2.3.3 Distribution-Based Method vs Clustering-Based Method .................. 167
    5.2.3.4 Proposed Methods vs Benchmarking ............................................ 168
    5.2.3.5 Effectiveness Of Clustering And Orthogonal Projection ................... 168
List of Figures

Figure 1. Outline of Framework ........................................... 12
Figure 2. Boundaries of Anomalies using Fuzzy Logic ...................... 13
Figure 3. Main Elements Associated with an Anomaly Detection Technique ...................... 20
Figure 4. Anomaly Detection Techniques (Chandola, et al., 2009) ...................... 22
Figure 5. Type of Anomalies in Online Social Network (Akoglu, et al., 2010) ...................... 35
Figure 6. Clustering Process ........................................... 50
Figure 7. Proposed Framework ........................................... 63
Figure 8. The “Friends of Friends are Often Friends” Pattern Network ...................... 70
Figure 9. Near-Star Topology ........................................... 70
Figure 10. Near-Clique Topology ........................................... 71
Figure 11. Local Patterns ........................................... 71
Figure 12. Full Star and Full Clique ........................................... 76
Figure 13. Labelling Procedure ........................................... 89
Figure 14. The Threshold Finding Algorithm ........................................... 94
Figure 15. Stars and Cliques ........................................... 99
Figure 16. Layer 2 of Framework: Distance Based Method ...................... 100
Figure 17. Steps to Detect Anomalies Using Distance-Based Approach ...................... 101
Figure 18. Modelling Points using Power Law Regression Line ...................... 108
Figure 19. Orthogonal Projection of Points on Power-Law Regression Line ...................... 108
Figure 20. Layer 2 of Framework: Distribution Based method ...................... 112
Figure 21. Steps Required in Distribution-Based Anomalies Detection ...................... 114
Figure 22. Observed Data Points ........................................... 117
Figure 23. Applying EM on Observed Data Points ........................................... 118
Figure 24. Number of Component vs Log Likelihood for Three Datasets ...................... 119
Figure 25. Fuzzy Inference Engine ........................................... 123
Figure 26. Layer 2 of Framework: Clustering Method ...................... 124
Figure 27. Steps Required in Clustering-Based Algorithm ...................... 127
Figure 28. Implementation of Fuzzy C-Means (FCM) Algorithm ...................... 132
Figure 29. Layer 3 of Framework-Evaluation ........................................... 137
Figure 30. A Network with Degree of Eight ........................................... 139
Figure 31. Power-Law Regression Model ........................................... 142
Figure 32. F-Score for Power-Law Regression Method ........................................... 143
Figure 33. F-Score for Orthogonal Projection Method ........................................... 144
Figure 34. Number Components vs Log Likelihood ........................................... 146
# List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table 1</td>
<td>Full Star Analysis</td>
<td>76</td>
</tr>
<tr>
<td>Table 2</td>
<td>Full Clique Analysis</td>
<td>77</td>
</tr>
<tr>
<td>Table 3</td>
<td>Summary of Methods</td>
<td>80</td>
</tr>
<tr>
<td>Table 4</td>
<td>Dataset Details</td>
<td>85</td>
</tr>
<tr>
<td>Table 5</td>
<td>Statistical Information of Generated Egonets</td>
<td>86</td>
</tr>
<tr>
<td>Table 6</td>
<td>Statistical Information</td>
<td>149</td>
</tr>
<tr>
<td>Table 7</td>
<td>Normal and Anomaly Distribution</td>
<td>157</td>
</tr>
<tr>
<td>Table 8</td>
<td>Facebook Result</td>
<td>158</td>
</tr>
<tr>
<td>Table 9</td>
<td>Flickr Result</td>
<td>159</td>
</tr>
<tr>
<td>Table 10</td>
<td>Orkut result</td>
<td>160</td>
</tr>
</tbody>
</table>
List of Abbreviations & Symbols

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABC</td>
<td>Average Betweenness Centrality</td>
</tr>
<tr>
<td>aScore</td>
<td>Anomaly Score</td>
</tr>
<tr>
<td>BC</td>
<td>Betweenness Centrality</td>
</tr>
<tr>
<td>CbLOF</td>
<td>Cluster-Based Local Outlier Factor</td>
</tr>
<tr>
<td>Com</td>
<td>Community</td>
</tr>
<tr>
<td>Deg</td>
<td>Degree</td>
</tr>
<tr>
<td>DNODA</td>
<td>Direct Neighbour Outlier Detection Algorithm</td>
</tr>
<tr>
<td>E</td>
<td>Edge</td>
</tr>
<tr>
<td>Egonet</td>
<td>Ego Network</td>
</tr>
<tr>
<td>EM</td>
<td>Expectation-Maximization</td>
</tr>
<tr>
<td>ESU</td>
<td>Enumerate Subgraphs</td>
</tr>
<tr>
<td>FCM</td>
<td>Fuzzy C-means</td>
</tr>
<tr>
<td>FIS</td>
<td>Fuzzy Inference System</td>
</tr>
<tr>
<td>FN</td>
<td>False Negative</td>
</tr>
<tr>
<td>FP</td>
<td>False Positive</td>
</tr>
<tr>
<td>FPR</td>
<td>False Positive Rate</td>
</tr>
<tr>
<td>G</td>
<td>Graph</td>
</tr>
<tr>
<td>GBAD</td>
<td>Graph-Based Anomaly Detection</td>
</tr>
<tr>
<td>GLODA</td>
<td>Global Outlier Detection Algorithm</td>
</tr>
<tr>
<td>GMM</td>
<td>Gaussian Mixture Model</td>
</tr>
<tr>
<td>HMRF</td>
<td>Hidden Markov Random Field</td>
</tr>
<tr>
<td>kNN</td>
<td>k-Nearest Neighbour</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Full Form</td>
</tr>
<tr>
<td>---------------</td>
<td>-----------</td>
</tr>
<tr>
<td>LOF</td>
<td>Local Outlier Factors</td>
</tr>
<tr>
<td>MDL</td>
<td>Minimum Descriptive Length</td>
</tr>
<tr>
<td>MFs</td>
<td>Membership Functions</td>
</tr>
<tr>
<td>N</td>
<td>Node (Vertex)</td>
</tr>
<tr>
<td>NMI</td>
<td>Normalised Mutual Information</td>
</tr>
<tr>
<td>OLS</td>
<td>Ordinary Least Squares</td>
</tr>
<tr>
<td>OSNs</td>
<td>Online Social Networks</td>
</tr>
<tr>
<td>P</td>
<td>Precision</td>
</tr>
<tr>
<td>R</td>
<td>Recall</td>
</tr>
<tr>
<td>RAND-ESU</td>
<td>Randomly Enumerate Subgraphs</td>
</tr>
<tr>
<td>RSVM</td>
<td>Robust Support Vector Machines</td>
</tr>
<tr>
<td>SNA</td>
<td>Social Network Analysis</td>
</tr>
<tr>
<td>SSR</td>
<td>Sum of the Squared Residuals</td>
</tr>
<tr>
<td>SVMs</td>
<td>Support Vector Machines</td>
</tr>
<tr>
<td>TN</td>
<td>True Negative</td>
</tr>
<tr>
<td>TPR</td>
<td>True Positive Rate</td>
</tr>
<tr>
<td>V</td>
<td>Vertex (Node)</td>
</tr>
</tbody>
</table>
Symbol | Description
---|---
\( \mathcal{G} = (\mathcal{V}, \mathcal{E}) \) | graph with set \( \mathcal{V} \) of vertices and set \( \mathcal{E} \) of edges
\( \mathcal{V}_{\text{egonet}_i} \) | set of nodes existing in egonet \( i \)
n | size of \( \mathcal{G} \)
\( \zeta^{(i)}_c \) | clique anomaly score for user \( i \)
\( \zeta^{(i)}_s \) | star anomaly score for user \( i \)
\( \zeta^{(i)} \) | logical disjunction of \( \zeta_s \) and \( \zeta_c \)
\( \mu_x \) | mean of cluster \( x \)
\( \mathbf{U} \) | matrix of the Fuzzy membership \((m_{id})_{n*c}\)
\( \beta \) | fuzziness index
\( \mu_d \) | mean for Gaussian \( d \) in mixture model
\( d^2(\zeta^{(i)}, \mu_j) \) | distance between \( i \)th data and \( j \)th cluster mean \( \mu \)
\( F_j \) | Fuzzy covariance
\((m_{id})_{n*c}\) | Fuzzy membership
c | number of cluster
\( \varepsilon \) | threshold
\( \text{Mx} \) | Membership Function \( x \)
\( \text{S} \) | Substructure
\( \text{DL} \) | Description Length
\( \text{R2} \) | Coefficient of Determination
\( E(\cdot) \) | Expected value
\( y = Cx^\theta \) | Regression model where \( \theta \) is the power-law exponent and \( C \) is a coefficient
$\tau, \text{TrShld}$: Threshold

$\text{Max Iter}$: Maximum Iteration

Prs: Precision

Rec: Recall

$\psi_{xm}$: Set of mean

$\phi$: A vector of unknown parameters for GMM

$\Sigma_d$: Covariance for latent variable $d$

$\mu_x, \sigma_x, \alpha_x, \beta_x, \gamma_x$: Fuzzy Membership Function Parameters
Statement of Original Authorship

The work contained in this thesis has not been previously submitted to meet requirements for an award at this or any other higher education institution. To the best of my knowledge and belief, the thesis contains no material previously published or written by another person except where due reference is made.

Signature: ___

Date: _____05/11/14________________________
Acknowledgments

Foremost, I would like to express my sincere gratitude to my advisors specially Associate Professor Rich Nayak for her continuous support, patience, motivation, enthusiasm and insightful comments. Her magnificent guidance helped me in all the time of research and writing of this thesis. I would also like to thank Dr. Douglas Stebila for his wonderful commitments, encouragement, and constructive comments. Finally thanks to Dr. Laurianne Sitbon for her help and motivation, Professor Acram Taji, Mrs. Noor Ifada, Dr. Gavin Shaw, Mr. Daniel Emerson, Mr. Paul Westall, Ms. Mary Finch, and my group members. It would not have been possible to finish my study without the help and support of the kind people around me.

This thesis is dedicated to my family…
Chapter 1: Introduction

This thesis is concerned with designing and developing a graph-theory-based framework and algorithms to detect anomalies in online social network data graphs.

1.1 MOTIVATION

In our everyday life, anomaly detection techniques are used explicitly or implicitly to detect divergences from what is normal or expected. Your neighbours can identify a possible thief by seeing the unusual behaviour of a stranger around your house. Using anomaly detection techniques such as clustering, different applications are able to discover uncommon patterns (Fawcett & Provost, 1999). Banks can find fraudulent activities by looking at uncommon spending patterns (Bolton & Hand, 2002). Network intrusion detection techniques (Brahmi, Yahia, & Poncelet, 2010) have been developed to find a possible attack on the computer network by comparing the normal traffic signature with the incoming traffic.

Research on anomaly detection, which dates back to the 20th century, was initiated by the statistics community. The anomaly concept varies according to the data domain it has been applied to. For instance, Hawkins (1980) characterises an outlier as “an observation that deviates so much from other observations as to
arouse suspicion that it was generated by a different mechanism”. Barnett and Lewis (1984) indicate that “an outlier is one that appears to deviate markedly from other members of the sample in which it occurs”. Johnson and Wichern (2002) defines an outlier as “an observation in a data set which appears to be inconsistent with the remainder of that set of data”. Essentially, the anomaly is perceived as an outlier. In the same way, online social networks (OSNs) analysts can find any unusual patterns which can lead to identifying any useful information about suspect users or illegal activities. For instance, any quantitative or qualitative features of a user behaviours in online social networks that are inconsistent with the rest of users can be considered anomalies (Faloutsos, 2014).

These simple definitions of anomaly is technically very challenging as several factors should be considered. Chandola (2009) described some of them as follows:

- Defining an accurate boundary between normal and anomalous behaviour is not possible. It is hard to distinguish instances sitting close to the boundary between a normal or anomalous instance.

- Defining a normal behaviour is complicated, especially when anomalies come from malicious actions. Anomalies usually adjust themselves to normal behaviour so anomalous observations cannot be distinguished.

- The current notion of normal and anomalous behaviour in several application domains might not work in the future as the concept of anomalousness continues to evolve with changes caused, for instance, by emerging technology.
• The unique definition of an anomaly is not possible as it depends on application domains. As a result, developed techniques in different domains are not easily cross-domain transferrable and need to be adapted.

• Noisy data is often likely to be similar to real anomalies, so it is hard to differentiate and eliminate noise from the data set.

These challenges make the most of the anomaly detection techniques limited to solving a particular formulation of the problem. It is very hard to solve an anomaly detection problem in a general form. Therefore, an anomaly detection technique needs to be developed and customised for a specific application by adopting notions from different disciplines such as statistics, machine learning, and data mining. Depending on applications and their limitations, we need to find a suitable approach to formulate the problem. Suitability of anomaly detection techniques for an application depends on the nature of the input data (e.g. discrete, continuous), the type of anomaly (e.g. point: if individual instance can be spotted as anomaly respect to the rest, contextual: an instance is anomalous in a specific context), the availability of labelled data, and the constraints and requirements that come from the application domain. This thesis mitigates the aforementioned challenges by using a fuzzy hybrid approach. For instance, to overcome the definition of an accurate boundary between normal and anomalous behaviour, we employ fuzzy logic in our approaches. Moreover we adapt machine learning techniques to the domain of OSNs to alleviate the cross-domain transferrable problem.
Online social networks provide online hangout spaces for everyone, especially young adults aged between 18 to 24, who makes up 75% of the people using online social networks (Papacharissi, 2010). They use this technology to socialise with interested friends and acquaintances, and to share information, photos, and videos. This powerful phenomenon, which captures the structure and dynamics of person-to-person and person-to-technology interaction, is being used for various purposes such as business, education, telemarketing, medical, entertainment and illicit activities. This technology also opens the door for unlawful activities. The increasing use of online social networks for committing illegal activities (Choo, 2009) presses authorities to find solutions for securing normal users. Analysing user behaviour to identify anomalies in this new perspective of social life that articulates and reflects the off-line relationships is now demanding. This emerging need is based on assumptions that: (1) the user behaviour and network pattern patterns carry useful information for the social network analysers; and (2) the patterns can be linked to unlawful activities, such as cyber-attacks and identification of intruders (Eberle & Holder, 2007). Detecting anomalies is an important factor in OSNs as these could be a sign of a significant problem or of carrying useful information for the analyser. For instance, an uncommon friendship pattern such as star topology in online social networks could be related to a celebrity or influential person. This kind of information can be used by financial companies for advertising their products in the influential person network. Identifying meaningful patterns and modelling them are considered to be important tasks by authority (government) and analytical studies.
The well-known anomalous topologies such as star and clique are used by existing approaches as a ground-truth for detecting anomalies. Faloutsos (2014) and Akoglu, McGlohon, and Faloutsos (2010) modelled online social networks with graph theory and characterise outliers (minority) as star or near-star, clique or near-clique, heavy vicinity, and dominant edge. Our experiments, applied on three different and popular online social networks datasets such as Facebook, Orkut, and Flickr (Cha, Mislove, Adams, & Gummadi, 2008; Mislove, Marcon, Gummadi, Druschel, & Bhattacharjee, 2007; Viswanath, Mislove, Cha, & Gummadi, 2009), also confirm that the majority follows the pattern of “friends of friends are often friends” and the minority (anomalous) follows either the “cliques or near-cliques” pattern (all the neighbours connected) or the “stars or near-star” pattern (mostly disconnected). Following the anomaly definitions (Akoglu, et al., 2010; Chandola, et al., 2009; Tong & Lin, 2011), these two types of patterns (clique and star) can be linked to anomalies in social networks, as only a minor population shows this distinct behaviour.

The quantitative structural features of online social networks such as relationship, in/out degree, betweenness centrality and community topology can be best represented as a fuzzy variable. Therefore they can be treated as a multiple-valued logic problem in which we have a degree of truth rather than only two possible values. For instance, how many friends should a user have to be considered a social or influential person? Or how much topology of a user network should be similar to a star or clique topology before being considered to be an anomaly? It is not accurate to use two-level logics such as binary to describe these kinds of characteristics. In reality the characteristics such as
“influentialness”, friendship, starness, cliqueness, and community are matter of degree and are relative. To be considered influential, a user must have at least a certain number of connections/friends; however, that number cannot be fixed. They can have overlap with the other sets in contrast to the binary. These properties of online social networks emphasise the need of fuzzy methods in order to tackle the problem of anomaly detection.

Two types of graph data can be collected from online social networks: (1) behavioural data that consider the dynamic usage behaviour of users; and (2) structural data that consider the structure of the user network graph. For instance, behavioural data can refer to analysing user behaviour with respect to the amount of time spent online or on chatting. Structural data include information of a user network’s topology in terms of number of connections and the characteristics of connections. The structural data are more valuable as they include properties of a graph that are not prone to being fabricated or denied by the users in any online social networks. The behavioural data are heavily dependent on technology and these kinds of data are not reliable. For instance these days many people, most of the time, are online due to the cheap Internet facility and new technology such as smart phones. The other example is chatting which now is using as a main way of communication. Therefore structural-based techniques which work on structural data can be more reliable and be a good candidate in detecting anomalies in OSNs.

Limited work has been done on applying structural anomaly detection techniques to online social networks due to issues such as accuracy, computational complexity, privacy, lack of labelled datasets and lack of sufficient
information (Akoglu, et al., 2010; Limsaiprom & Tantatsanawong, 2010). These limitations lead to lack of customised anomaly detection techniques for online social networks. Methods for finding outliers in structural data can be divided into distance-based, distribution-based, and clustering based. These methods do not work well if datasets include fuzzy instances (e.g. topological similarity) or sparse instances (e.g. connectivity matrix). Storing and manipulating sparse data face time and space complexity issues. Fuzzy instances need to be treated with multi-levels logic in order to achieve a better accuracy. Moreover, the existing works on structural-based techniques are not fuzzy-based and suffer from missing outliers in sub-networks with a high number of nodes. Existing methods are not specifically focused on the online social network and also do not consider the fuzzy characteristics of objects under investigation.

To overcome such these limitations, this thesis presents, firstly, work on structural-based anomaly detection methods as they employ users’ network topology meta-data which cannot be fabricated, impersonated and denied by the users; on the other hand, inputs to the behavioural-based methods are not easily available and also can be impersonated. The processes required such as gathering accurate data for behavioural techniques are technology dependent and not easy to develop as new technologies emerge quickly. Secondly, to improve accuracy, we present various hybrid methods within a multi-layer framework that utilise the discrete and continuous types of input data. The hybrid methods include various combinations of the distance-to-regression model, the orthogonal projection, clustering, the statistical model, and fuzzy logic methods.
1.2 PROBLEM STATEMENT

A common approach to identifying anomalous objects, known as supervised learning or classification, is to learn from training datasets which include normal and/or abnormal instances to make a model. The abnormal instances then can be identified if they significantly differ from the model. This needs a rich dataset in terms of proper labelling to make an accurate prediction. However, in many cases such as online social networks, the process of finding or making a labelled dataset is expensive and time consuming, and often impossible due to the nature of datasets such as privacy (Akoglu, et al., 2010; Bouguesa, 2011; Hu, Mac Namee, & Delany, 2008; Limsaiprom & Tantatsanawong, 2010).

Clustering, another common approach to identify anomalies, is an unsupervised technique used for categorising similar data instances into groups. Data instances are assumed to be normal if they fit in large and dense clusters, and to be anomalies if they fit in small or sparse clusters. Clustering is usually performed in an unsupervised way without utilising any a-priori knowledge. In this thesis, we start with no a-priori knowledge of what is normal and what is abnormal. To identify normal and anomalous users, we look at the behaviours followed by the majority and minority of users. Behaviour exhibited by the majority defines normal; by the minority defines anomalous. In unsupervised anomaly detection approaches the aim is to cluster similar objects; however semi-supervised ones are interested in determining which cluster has accommodated more anomalous objects. Our approach to this problem is to use both unsupervised and semi-supervised techniques in a hybrid way within a multi-layer framework.
Detecting outliers in the structural data of online social networks such as the links that they have established with other users in the network (friendship) is the specific problem which we consider in this research. While users can hide their identity by supplying false information and can deceive analysts, analysing certain types of metadata such as user connections topology can help to spot anomalies more accurately. This metadata can be modelled as a graph in which nodes represent people and edges represent the links. The edges connect nodes/people using a range of relationships such as friendship, affiliation, family and many others.

During the course of this research, we have attempted to improve the accuracy of existing algorithms in detecting structural-based anomalies either including the “clique or near-clique” pattern (all the neighbours connected) or the “star or near-star” pattern (mostly disconnected). Previous works (Akoglu, et al., 2010; Gupta, Jing, Xifeng, Cam, & Jiawei, 2013; Shrivastava, Majumder, & Rastogi, 2008; Tong & Lin, 2011) have established that these two types of patterns can be linked to abnormalities in the network, particularly in online social networks. For instance the online social network patterns leading up to, during and after the 9/11 terrorist attack (Akoglu, et al., 2010) took on the topology of either a clique or a star topology. In the first example, it means all the members involved in the attack have connections to each other. In the second example, a user connects to others indiscriminately, without any direct connections between the targeted users. Both patterns contrast with the most common pattern (friends of friends are often friends) and can be considered as anomalies.
From a technical point of view the aim of this research is to introduce novel methods and features for overcoming the limitations of existing outlier detection algorithms in online social networks, and to improve the accuracy. This is done by taking advantage of graph theory to model OSNs and extract new suitable features, of fuzzy logic to deal with fuzziness of structural behaviours in OSNs, and of various hybrid methods within a multi-layer framework to improve accuracy. The hybrid methods include the combinations of the distance to regression model, orthogonal projection, clustering, and the statistical model that utilise different types of input data, such as discrete and continuous.

More specifically, In order to overcome the limitation of existing methods by improving detection accuracy, this thesis develops methods based on three different well known machine-learning models: distribution, clustering, and distance. These different models are employed to deal with the different types of graph metrics generated from modelling the social network data as graphs. For instance, in the distribution-based approach, this research is interested in accuracy improvement by a combination of Gaussian Mixture Models and fuzzy logic for continuous domain data. The fuzziness characteristic of instances is the missing point in the existing methods. The use of fuzzy logic allows the handling of instances with different levels of uncertainty. The clustering-based approach, which uses discrete domain data, aims to improve and adapt the fuzzy c-means clustering method using maximum likelihood estimation and fuzzy logic. In the distance-based approach the target is to accurately enhance the OddBall method (Akoglu, et al., 2010) by using a power-law regression model as well as the proposed method of orthogonal projection of instances on the regression model.
1.3 RESEARCH QUESTIONS AND OBJECTIVE

This thesis provides an automatic process of anomaly detection in graph data generated from online social networks. Identification of the research gaps in anomaly detection in online social networks has led to the formulation of the following research questions.

- What are the new features to select from the graph modelling the online social network data in order to represent anomalies and to get better insight into discover anomalies?

- How can fuzzy-based machine learning techniques are developed to detect anomalies and increase accuracy using the selected features of online social networks?

- How a multi-layered framework should be used for analysing and evaluating the proposed methods using unlabelled datasets with semi-supervised learning approaches?

The proposed research aims to employ the use of graph theoretical modelling and data-mining techniques in order to improve the accuracy of the anomaly detection techniques in graph data such as that found in online social networks. The objective of this research is to propose novel hybrid data-mining based approaches within a multi-layer framework to find anomalies in structural data of online social networks.
1.4 CONTRIBUTION TO THE BODY OF KNOWLEDGE

This research introduces the proposed approaches within a multi-layered framework and the notion of using the unlabelled dataset of online social networks to detect anomalies. We explore, develop, and test different approaches to the problem within the proposed framework. The approaches fall in the category of unsupervised outlier detection and semi-supervised learning. We demonstrate that these novel techniques can most accurately identify anomalous users. The multi-layered framework shown in Figure 1 provides for the full requirements needed for finding anomalies in online social networks data graph. These include modelling, algorithms, labelling, and evaluation. In the first layer of the proposed framework, we model an online social network with the graph theory and extract the various new graph features for the nodes in the graph. These new features include betweenness centrality, average betweenness centrality, starness degree, cliqueness degree and the community cohesiveness of a user’s local network. These features are used as inputs or framework baseline data for all the developed methods in this research.

![Figure 1. Outline of Framework](image-url)
The second layer of our framework includes our proposed methods for tackling the problem of anomaly detection in online social networks from different angles using different inputs. Distance-based, distribution-based, and clustering-based are three angles that we tackle in this layer. We use fuzzy logic to define the boundaries of anomalies as they can be treated as a multiple-valued logic problem in which we have a degree of truth rather than only two possible values (normal or abnormal), as shown in Figure 2.

These methods are able to overcome these existing problems of anomaly detection techniques: (1) missing anomalies for sub-networks with a high number of nodes and edges, (2) considering anomalous instances, which sit far from the anomalous instances, which sit far from the regression model, normal in distance-based methods, and (3) missing fuzzy nature of online social network during detecting process. All of these can affect the quality of detecting anomalies within the online social network data.

Figure 2. Boundaries of Anomalies using Fuzzy Logic
The third layer of our framework is allocated to labelling a subset of dataset and evaluating the proposed methods. Using the labelled data is an important step in the evaluation process. However, labelled datasets in online social networks is hard to get access to, due to privacy. A labelled dataset is often prepared manually by a human expert using visualisation techniques (Chandola, et al., 2009). In most cases generating labels for normal behaviour is easier, compared to getting a labelled set of anomalous behaviour, because of their dynamic nature. In our case we try to have different types of anomalous data to make sure we can evaluate our approach more accurately.

Given a set of labelled data, we can then evaluate the proposed methods by computing for each metric a threshold that minimises the number of false positives and false negatives, and finally by comparing the results to state of the art methods as a benchmark. We apply this framework to datasets from three different and popular online social networks (Facebook, Orkut, and Flickr) to determine which of the proposed methods are best suited for identifying outliers in the real world. We find that our proposed distribution-based method is more accurate than existing others methods such as the OddBall (Akoglu, et al., 2010).

In general the research gaps from the literature review are identified as:

- Low accuracy of detecting anomalous behaviours in online social networks and absence of customised methods for OSNs;
- Using technology dependent methods in terms of how users’ usage patterns are mined;
• Using the binary logic problem in which only two possible values
  (normal or abnormal) are considered.

These shortcomings are overcome by:

• Using hybrid methods based on graph metrics, a power-law regression model, orthogonal projection, and clustering algorithms;

• Using structural features which are not deniable by users as inputs to the proposed methods;

• Using fuzzy logic to solve a multiple-valued logic problem such as anomaly.

The main contributions are summarised below:

  o Developed distance-based anomaly detection methods using orthogonal projection, a clustering algorithm, and new graph metrics and definitions such as average betweenness centrality, and community cohesiveness;

  o Designed a framework to evaluate the proposed approaches.

  o Developed Distribution-based anomalies detection methods using EM-Gaussian Mixture Model algorithm, graph, two anomaly scores for starness and cliqueness, and a combination of Gaussian Mixture Model and fuzzy logic as a novel method to differentiate between normal and anomalous instances;

  o Used fuzzy linguistic and quantitative variables to symbolise uncertainty such as friendship relations in online social networks.
 Developed Clustering-based anomalies detection methods using the natural underlying relationship between instances to define the number of clusters automatically, and a fuzzy membership function to define the boundary of anomalies.

1.5 OUTLINE OF THE THESIS

This thesis comprises six chapters. Following this introductory chapter, the literature review, Chapter 2, provides a summary of related work in the field of online social networks, anomaly detection techniques, clustering algorithms, fuzzy logic, and unlabelled datasets. Three main categories of methods based on distance, distribution, and clustering are discussed, including the advantages and disadvantages of each method. Online social network modelling and analysis methods are also discussed.

Chapter 3, the multi-layer framework, gives details of our proposed multi-layer framework steps: extracting the new graph-base features, designing our methods, and evaluating these methods. It uses meaningful features that represent the network in order to detect anomalies. The details about the experimental setup, data format, and statistics of the dataset are presented. The benchmarking methods are introduced and mechanisms on how they work are explained.

Chapter 4, the proposed approaches, presents the developed anomaly detection methods. The proposed approaches use three different concepts distance-based, distribution-based and clustering-based. The findings of these
methods, as well as new features selection, were presented respectively in the following papers according to the underlying concepts:

Distance-based:


Distribution-based:


Clustering-based:


Chapters 5, presents analysis of the result of experiments applied on three different real-life datasets.

Chapter 6, the conclusion, summarises the research findings and proposes future work.
Chapter 2: Literature Review

In Chapter 1, we introduced the research challenge of anomaly detection in online social networks. We noted that anomaly detection is an increasingly important aspect of this new emerging technology. In this chapter we review the literature related to the research problem of detecting anomalies in OSNs.

2.1 ANOMALY DETECTION

Anomaly detection is an important problem that has been researched within diverse research areas as shown in Figure 3. Chandola, et al. (2009) categorises it in research areas and application domains. They include bank, fraud, industrial damage, image processing insurance, critical systems, health care, military and online social networks. Anomaly detection, also called outlier detection, refers to detecting patterns which do not comply with accepted behaviours. Detecting anomalies is an important factor in a domain as it could be either a sign of a significant problem or carrying useful information for the analyser. For instance, an uncommon friendship pattern such as star topology in online social networks could be related to a celebrity or influential person (Zhang et al., 2013). This kind of information can be used by a financial company for advertising their products in the influential person’s network.
In this section we review anomaly detection challenges, anomaly detection techniques, evaluation methods, and generating the labelled data for evaluation. We also study key assumptions used by each technique to determine which techniques are suitable for online social networks.

2.1.1 ANOMALY DETECTION TECHNIQUES

As shown in Figure 4, anomalies can be categorised into three classes: point anomalies, contextual anomalies, and collective anomalies (Chandola, et al., 2009). A point anomaly refers to detecting an anomalous data instance with regard to the remainder of the data. A contextual anomaly, also called a conditional anomaly (Song, Wu, Jermaine, & Ranka, 2007), refers to a data
instance which is considered anomalous in a specific context, but not in others. For instance, dropping temperature in June is not normal in many countries but it is normal in others. In a contextual anomaly data instance is defined by using contextual attributes and behavioural attributes (Ahmad et al., 2013; Bogdanov, Busch, Moehlis, Singh, & Szymanski, 2013; Chen, Wu, Srinivasan, & Zhang, 2013; O'Banion & Birnbaum, 2013; Xia, Ribeiro, Chen, Liu, & Towsley, 2013). Contextual anomalies have been generally investigated in time-series data (Salvador, Chan, & Brodie, 2004) and spatial data (Shekhar, Lu, & Zhang, 2001). A collective anomaly refers to detecting anomaly techniques that call a collection data instance anomalous with regard to the whole data set. The data instances themselves may perhaps not be anomalies if they do not occur together as a group. For example, a sequence of events such as buffer-overflow, ssh, and ftp could be a sign of a typical Web-based attack by a remote machine. Collective anomalies have been investigated for graph data (Noble & Cook, 2003). In summary, point anomalies are used for all datasets and collective anomalies are used only for related data instances.

This research focuses on point anomalies for which different techniques such as supervised, rule-based, semi-supervised, unsupervised, and statistical anomaly detection have been used (Chandola, et al., 2009; Guthrie, Guthrie, Allison, & Wilks, 2007; Hodge & Austin, 2004; Noble & Cook, 2003; Patcha & Park, 2007). The techniques can also be sub-divided into different groups including classification-based, nearest neighbour-based, clustering-based, statistical, information theoretic, and spectral.
2.1.1.1 SUPERVISED ANOMALY DETECTION

In the supervised mode, anomaly detection techniques assume that there exists a training data set in which instances have been labelled into normal and anomaly classes. Any unobserved data instance is analysed by the predictive model, which is built by this approach of deciding whether it is normal or not. A supervised anomaly detection approach has two main problems. First, the total numbers of anomalous instances are usually much less than the number of normal instances in the training dataset. These issues have been significantly investigated in data mining and machine learning literature (Chawla, Japkowicz, & Kotcz, 2004; Joshi, Agarwal, & Kumar, 2001, 2002; Phua, Lee, Smith, & Gayler, 2010; Zellag & Kemme, 2014). Second, obtaining descriptive and precise labels is tricky, particularly for the anomaly class. Inserting synthetic anomalies into a normal data set in order to get a labelled training data set has been suggested by some researchers (Abe, Zadrozny, & Langford, 2006; Patcha & Park, 2007).
These supervised anomaly detection methods can further be divided into two groups: multi-class and one-class, according to Chandola, et al. (2009). Multi-class methods deal with training data that contains labelled instances belonging to several normal classes (Barbara, Wu, & Jajodia, 2001; De Stefano, Sansone, & Vento, 2002; Gogoi, Borah, & Bhattacharyya, 2011). If the classifier cannot classify a test instance as normal, it will be classified as anomalous. Some techniques link a confidence score to the classifier’s prediction in order to classify the test instance as normal or anomalous. These techniques use a one-class classification algorithm in order to learn a discriminative boundary around the normal instances (Roth, 2005, 2006; Schölkopf, Platt, Shawe-Taylor, Smola, & Williamson, 2001; Zhong, Khoshgoftaar, & Seliya, 2007). If a test instance could not fit in the learned boundary, it is considered as anomalous. The one-class classification algorithms are neural networks-based, Bayesian networks-based, support vector machines-based, and rule-based to build classifiers.

Neural networks can be used in both multi-class and one-class anomaly detection techniques. A multi-class technique uses neural networks in two phases, learning and detecting. In the learning phase, a neural network learns the various normal classes by using information which came from the normal training data. In the detecting phase, if a test instance provided as an input is rejected by the neural network, it is an anomaly otherwise; it is normal (De Stefano, et al., 2002). One-class technique uses replicator neural networks to detect an anomaly (Hawkins, He, Williams, & Baxter, 2002).

A multi-class technique also uses Bayesian networks to detect anomalies. Bayesian networks are used by a univariate categorical data set to estimate the
posterior probability of observing a class label from normal class labels and the anomaly class label set for a given test data instance. Laplace Smoothing is utilised to smooth the zero probabilities, particularly for the anomaly classes (Chan, Mahoney, & Arshad, 2003). A number of variants of the basic technique have been offered for detecting the anomaly in some domains such as network (Sebyala, Olukemi, & Sacks, 2002), and text data (Baker, Hofmann, McCallum, & Yang, 1999). This basic technique is based on independency between the different attributes. A number of the basic technique variations use complicated Bayesian networks to catch the conditional dependencies among the various attributes (Das & Schneider, 2007; Janakiram, Reddy, & Kumar, 2006).

To detect anomaly in the one-class setting, support vector machines (Ratsch, Mika, Scholkopf, & Muller, 2002; Vapnik, 2000) have been used. The basic technique decides whether each test instance belongs to the learned region or not. If a test instance belongs to the learned region, it is called normal; otherwise it is called anomalous. The basic technique has also been developed to detect anomalies in temporal sequences (Ma & Perkins, 2003). Robust support vector machines (Song, Hu, & Xie, 2002) have been employed to detect intrusion (Hu, Liao, & Vemuri, 2003).

Rule-based anomaly detection techniques which belong to the supervised group differentiate between normal and anomalous behaviour of a system, based on the rules which have been learned by the technique during the training phase (Duffield, Haffner, Krishnamurthy, & Ringberg, 2009; Wong, Moore, Cooper, & Wagner, 2002). If a test instance does not follow the rules, it will be marked as an anomaly. Rule-based techniques could be applied in both multi-class and one-
class settings. A basic multi-class technique in the learning phase uses a rule learning algorithm such as RIPPER or Decision Trees to learn rules from the training data (Lee & Stolfo, 2000). They use support and confidence as a measure for evaluating the generated rules. The rule $X \Rightarrow Y$ has support $s$ if $s\%$ of occurrences in dataset $D$ contain $X\cup Y$. The rule $X \Rightarrow Y$ has confidence $c$ if $c\%$ of occurrences in dataset $D$ contain $X$ also contain $Y$. Confidence is calculated by dividing the number of training instances which are accurately classified by the rule by the total number of training instances covered by the rule. In the matching phase, the basic multi-class technique matches each test instance with a suitable rule. The test instance anomaly score is calculated from the inverse of the confidence associated with the suitable rule. One class uses association rule mining to generate rules from the data in an unsupervised mode (Agrawal & Srikant, 1995; Guo & Li, 2008; Qin & Hwang, 2004; Tandon & Chan, 2007; Yairi, Kato, & Hori, 2001).

The statistical-based techniques in general can be categorised under unsupervised or supervised learning, depending on the underlying concepts they use. Statistical anomaly detection techniques are built on the assumption that normal data instances happen in high probability regions of a stochastic model, whilst anomalies happen in the low probability regions of the stochastic model (Qayyum, Islam, & Jamil, 2005; Stibor, Timmis, & Eckert, 2005). There are two main statistical anomaly detection techniques: parametric and nonparametric. Both of them have been applied to fit a statistical model. The supervised parametric and nonparametric techniques learning include regression model-based, Histogram-Based and Kernel Function-Based (Chandola, et al., 2009).
2.1.1.2 SEMI-SUPERVISED ANOMALY DETECTION TECHNIQUES

The semi-supervised anomaly detection techniques assume that there is a training subset in which instances have been labelled for only the normal class (Blanchard, Lee, & Scott, 2010; Laleh & Azgomi, 2009; Moonesignhe & Tan, 2006; Noto, Brodley, & Slonim, 2010; Su & Tsai, 2011). For instance, in spacecraft fault detection (Fujimaki, Yairi, & Machida, 2005), an accident indicates an anomaly which is difficult to model. The classic method used by these techniques is to build a model for the normal behaviour, and use the model to detect anomalies. A limited number of techniques, such as the negative selection algorithm, operate based on the availability of only the anomaly instances for training (Dasgupta & Majumdar, 2002). Difficulty in finding a training dataset that includes all possibilities of anomalous behaviours causes these techniques to be uncommon.

2.1.1.3 UNSUPERVISED ANOMALY DETECTION TECHNIQUES

In the unsupervised mode, anomaly detection techniques do not need training data. This type of technique, which is widely used, implicitly assumes that normal instances are more common than anomalies in the data. In these clustering-based methods (Abe, et al., 2006; Chimphlee, Abdullah, Sap, & Noor, 2005; Tan, Steinbach, & Kumar, 2005; Yalamanchili, Jain, & Parekh, 2009), the false alarm rate will be increased if this assumption is not accurate. Several semi-supervised techniques can work in unsupervised mode by employing an unlabelled dataset sample as training datasets if the number of anomalies was very
few compared to the whole dataset (Basu, Bilenko, & Mooney, 2004; Guthrie, et al., 2007).

Clustering-based techniques are developed by the following concepts. One group assumes that normal data instances fit in a cluster; anomalies do not fit in any cluster, they appear as outliers. The other group assumes that whilst normal data instances sit by their nearest cluster centroid, anomalies are far from their nearest cluster centroid. The third group assumes data instances are normal if they fit in large and dense clusters, and anomalies if they fit in small or sparse clusters.

The nearest neighbour techniques are based on the assumption that the normal data instances happen in close-together neighbourhoods, while anomalies happen distant from their closest neighbours. These techniques need a distance or similarity measure which is defined between two data instances. Distance or similarity can be calculated in several ways. For instance, Euclidean distance is a common choice for continuous attributes; however other measures can be used (Tan, et al., 2005) as well. Using distance to $k^{th}$ nearest neighbour, and using relative density, are two methods which are employed by nearest neighbour-based anomaly detection techniques (Boriah, Chandola, & Kumar, 2008). Relative density assumes that an instance which is in a neighbourhood with low density is an anomaly and an instance which is in a dense neighbourhood is normal. The difference between clustering-based and nearest neighbour-based techniques is in the way they operate. In the clustering-based algorithm each instance is evaluated based on its cluster although they sit far from the centre of the cluster but in neighbour-based each instance is evaluated based on its local neighbourhood. There are some methods using both concepts togethers.
The statistical based techniques which can be categorised under unsupervised techniques include the Gaussian model-based and mixture of parametric distributions-based (Chandola, et al., 2009). These parametric anomaly detection techniques assume that normal instances happen in the high probability regions whilst anomalies happen in the low probability regions of the stochastic model (Qayyum, et al., 2005; Stibor, et al., 2005).

2.1.2 REPORTING ANOMALY DETECTION

The way the anomalies are reported is one of the key aspects in designing an anomaly detection technique. Scores and labels are the two ways that an anomaly can be reported (Chandola, et al., 2009). The scores-based technique creates a list of ranked anomalies as an output. Analysts use the list to determine the anomalies by using a cut-off threshold or choosing the top few anomalies from the list. In the labels-based technique, all test instances are labelled as normal or anomalous using an anomaly technique. This does not let the analyst use a domain specific threshold to pick the most relevant anomalies. However, this can be done implicitly through parameter selection within a technique.

2.1.3 SUMMARY

Each type of anomaly detection technique previously listed has its own strengths and shortcomings. Nearest neighbour and clustering-based techniques face problems detecting anomalies if high dimension datasets are being used
(Chandola, et al., 2009), due to the difficulties of calculating the distance in high dimensions datasets for detecting anomalies. For complex datasets, classification-based techniques would perform better if normal and anomalous instance labels were available. However, imbalanced spreading of the normal and anomalous labels over datasets causes a considerable difficulty for these techniques. With low-dimensional datasets, statistical techniques perform well if the assumptions are met.

We will model an OSN with graph theory; therefore from the nature of the input data perspective our unlabelled data instances are described by vertices (nodes) and edges, with edges connecting the vertices together. Vertices represent users and edges match to friendship links. Moreover, data graph modelled in this thesis can be considered as a low dimensional dataset. Based on these characteristics, we find that semi-supervised techniques, which are also able to deal with an unlabelled dataset, are suitable for our purpose. For instance, other approaches such as unsupervised algorithms suffer from unavailability of ground-truth for training leads to inefficiency of detecting anomalies (Aggarwal, 2013). To overcome this problem, the process of labelling datasets using visualisation methods is utilised. The generated labelled data are used as a training model in a semi-supervised way. More details about our proposed algorithm for detecting anomalies are described in Chapter 4.

2.2 ANOMALY DETECTION AND ONLINE SOCIAL NETWORKS

Online Social networks, a web-based service, allow users to: (1) create a public or semi-public profile within the network; (2) add a list of other users and
their shared connection to their profiles; and (3) view and navigate their list of links and also the links built by others users within the system (Boyd & Ellison, 2008). In social networks individuals are interdependent, and their behaviour is based on these interdependencies. Analysing online social networks has become a popular research area for practitioners and theorists interested in study of behaviours as the growing literature in this area shows (McGloin & Kirk, 2010; Wang & Lu, 2013). Online social networks are seen as rich sources of useful information by various application domains including business, education and social. Significant knowledge from social networks can be obtained by using these analysis techniques to detect unusual behaviours for future study. Identifying meaningful patterns and modelling them are considered as important tasks by authority (e.g. governments) and analytical studies.

2.2.1 ONLINE BEHAVIOURS

Online behaviours refer to habitual actions formed in cyberspace which can be a reflection of offline activities (Tang, 2012). Computer mediated communication via online social networks plays a key role in this kind of behaviour. Therefore, it is important first to understand the nature of the medium and its implications. A social network can be seen as a computer mediated communication service that allows people to create a profile (public or semi-public), share and establish connection among each other, and view and navigate through their list of connections (Centola, 2010). It also allows users to post comments, to provide private messaging, to share video and photo, and to have instant messaging (Boyd & Ellison, 2007).
A profile is a unique page that shows personal information which is obtained by asking a series of questions such as age, location, and interests (Sundén, 2003). The posted information can be defined to be public, semi-public, or private while they also contain true or false information. The false information in the profile can be a result of precaution or misleading (Choo, 2009). This facility in addition to the absence of physical interaction with unknown persons requesting friendship allows online predators and criminals to ply their trade.

This phenomenon enables users to have “latent ties” and connect to individuals who have never met or it would not otherwise be made offline (Haythornthwaite, 2005). Latent ties are connections that technically exist but have not yet been active. These kinds of connections can be built automatically by computer software or other means such as registering in an institute e-mail system or accepting friendship from someone just met. Using online social networks, these latent ties can be changed to strong ties such as friendship. Regardless of type of connections the established connections between users of OSNs called friends, contacts, fans, links, or followers can be categorised in one-directional or bi-directional ties (Boyd, 2006). These connections cannot be denied or fabricated by users even if they want to do so. Therefore they are the most reliable information which can be used for analysing user behaviours in online social networks.

Navigating through someone’s articulated list of friends is one of the important services within online social networks. The list of friends on most OSNs could be viewed by users who have an active account. Many new connections are established by using this feature. The previous research has
shown that the connection data can be clustered using the information such as age, work, nationality, ethnic, religious, sexual orientation, educational level, political, or other identity-driven categories (Thelwall, 2008).

Users behaviours in online social networks are categorised into passive (no shows), active (inviters, linkers), newcomers, onlookers, cliquers, star, and mix-n-mingler behaviours (Rozen, Askalani, & Senn, 2012 ). Passive users have low levels of trust to OSNs and join online social networks only because of curiosity, friend persistence, and protect themselves by limiting their engagement (Kumar, Novak, & Tomkins, 2010). These users do not log in to a social network very often and have no interest in making anyone aware of their activities or interests (Thelwall, 2008). Although they do not trust any social networks, they use posted information as a source of updating.

Sparks users are the most active users of social media. Inviters are kind of active users who are interested in articulating their offline communities into online, and actively involved in inviting and encouraging their friends to join the online services. They are most influential users in the culture of the network (Gupta, Sycara, Gordon, & Hefny, 2013). These users with high level of trust to OSNs easily share their personal information, traverse through the network, and freely distribute their personal details. An inviter sends requests to non-members to join online social networks. Linkers are users who connects themselves to the other users (Kumar, et al., 2010) play a huge role in the growth of the online social network by actively contributing to make connections to other users. They are the most active and influential users within their groups and communities.
They look at social networks as one of the effective and immediate self-expression method.

Newcomers’ users are kinds of passive users of a social network who joined recently to online social network. Onlooker users are active but by traveling through network and sharing almost no personal information. Cliques who tend to be influential among their direct neighbours, are categorised in this group (Subbian, Sharma, Wen, & Srivastava, 2013). A linker sends requests to existing users including inviters. Stars are users who discriminately connect to the other users with almost no connections between them. Mix-n-minglers users are those ones who often share and interact with a various community. According to a recent study, the typical user, for instance, in Facebook can be categorised as moderately-active user in terms of sending requests, posting content, and liking the content of their friends (Hampton, Goulet, Marlow, & Rainie, 2012). It means the average user receives more requests and contents than they send. Female, old and unmarried users are more susceptible to be influenced compared to male, young, and married users respectively (Aral & Walker, 2012).

Developing trust is one of the most important problems within online activities (Choo, 2009). The anonymity of identification is used by predators to develop trust and intimacy much faster online than in face-to-face relationships. Online predators are expert in finding vulnerable targets by collecting personal information, searching within profiles, playing as a trusted party, and using fake identities. The online predators can be categorised into three groups. First, privacy predators (Choo, 2009; Hitchcock, 2007), these groups use OSNs to breach the privacy of the other users. For instance, they might use public Twitter feed to
track down where someone has lunch with someone else. Second, sexual predators (Berson, 2003; Wolak, Finkelhor, Mitchell, & Ybarra, 2010), they use OSNs to obtain sexual contact with another person in a predatory manner. Third, financial predators (Bapna, 2003), they use OSNs to obtain the financial information such as bank account or offer fake money-making schemes.

2.2.2 TYPE OF ANOMALIES

Akoglu, et al. (2010) and Faloutsos (2014) modelled online social networks with graph theory. Faloutsos (2014) suggests to use EigenSpokes, the highest magnitude projection along the singular vector, to spot anomalies. These are those who use similarity between induced sub-graphs and near-cliques topology as a method of identifying anomaly. Faloutsos shows that cliques and star topology can potentially relate to suspicious activities in financial predators (e.g. on-line buyer-and-seller settings), Facebook, and twitter-like networks. Akoglu, et al. (2010) also characterises outliers as star or near-star, clique or near-clique, heavy vicinity, and dominant edge, as shown in Figure 5. The star or near-star topology is a kind of graph networks whose nodes are connected to a central node like a hub. The hub provides a shared point for the other nodes, which have no or minimum connections with each other. The star concept can be related to online social networks: a user connects to the other users indiscriminately as there are no connections between the connected users and the central one. The clique or near-clique topology (or a complete subgraph) is a subset of a graph network in which every two or more nodes are connected by an edge. This concept in online social networks refers to groups of people, all of whom know each other. Heavy
vicinity and dominant edges are as concern to user networks which have considerably high total edge weight compared to the number of edges. For instance, in online social networks, if a user receives a large number of messages from a single source (user), this can be considered as heavy vicinity. As this research does not focus on weighted graphs the heavy vicinity and dominant edge topologies are not considered. In this context, abnormality is defined as uncommon behaviour patterns that are not represented by the majority in a training dataset. The majority of users in online social networks show the pattern of “friends are often friends”. In off-line social networks (real society), from a sociological perspective, it is likely that two friends of a person are also friends of each other (Zuckerman & Jost, 2001). This phenomenon is also true for online social networks (Subrahmanyam, Reich, Waechter, & Espinoza, 2008; Zuckerman & Jost, 2001).

![Figure 5.Type of Anomalies in Online Social Network (Akoglu, et al., 2010)](image)

Star or near-star  Clique/near-Clique  Heavy vicinity  Dominant edge
2.2.3 OSNS ANOMALY DETECTION CHALLENGES

Although outlier detection concept seems very simple, it is a difficult problem to solve especially in online social networks when we are dealing with human behaviors. The challenges include low accuracy, time and computational complexity, unlabelled datasets, privacy, temporal velocity, lack of availability of sufficient information, and no consistent definition for an anomaly across different online social networks (Gjoka, Kurant, Butts, & Markopoulou, 2010; Gross & Acquisti, 2005; Hodge, 2006; Kumar, et al., 2010; Ugander, Karrer, Backstrom, & Marlow, 2011). Finding a labelled dataset in any social network is difficult due to the privacy of users. None of the online social network providers are keen to make their users’ data available to the public due to the legal issues. Lack of sufficient information such as exchanged messages between users in existing datasets makes it much harder for anomaly detection techniques. Moreover, OSNs providers, over time, keep adding new features and capabilities to their network to respond to new demands. This brings new challenges for outlier detection techniques in order to achieve their goal. In addition, different structures and purposes of existing online social networks bring inconsistency between definitions of anomalies that is not easy to overcome by a generalised approach.

Analysis of an online social network’s graph can be constituted in two categories: local and global (Newman, et al., 2002). This structural division is important as it can be used as an analytical leverage to get inside user behaviours. The local view of user networks concentrates on extracting rules about users’ behaviour; the global view focuses on generalising the extracted rules as patterns for differentiating users’ online behaviour. For instance, the local and global
views of users’ friendship networks can be utilised to generate usage patterns. According to the anomaly definition (Chandola, et al., 2009), if the usage pattern of a user follows the common usage pattern defined by the global view, the user behaviour can be classified as “normal”, otherwise it can be called “anomalous”.

2.2.3.1 LABELLED DATASET

The accuracy of any anomaly detection technique depends on the datasets used for training. Without a good training dataset it is hard to build an effective model to detect anomalies accurately. A good training dataset requires the existence of real different examples of the normal and anomalous instances which have been labelled properly. However in many cases such as online social networks finding the real labelled datasets is very hard due to the privacy of users. The important factor is to have the relations between users in datasets in order to build egonet and computing the related graph metrics. For instance, finding a dataset, that includes the other datasets and labelling, specifically datasets with pre-established labelling of anomalies, such as viral marketing/attack or criminal records, is difficult to get. In these cases a labelled dataset is often prepared manually by a human expert using visualisation techniques (Bloedorn et al., 2001; Chandola, et al., 2009; Jyothsna, Prasad, & Prasad, 2011; Liu, Li, Lee, & Yu, 2004; Sims, Sinitsyn, & Eidenbenz, 2013; Singh & Upadhyaya, 2012). In the labelling process, visualisation software such as R, or NodeXL are used to view users networks, which are selected randomly, to decide whether they are anomalies. The ground-truth for this process is based on “friends of friends are often friends” pattern which is followed by the majority. The variations of the
minor patterns such as star and clique which can lead to anomalies, according to
the literature, are used as the ground-truth to generate labels for anomalous users.

Generating labels for normal behaviour data instances is easier in
comparison to generating labels for anomalous instances because of their dynamic
nature. Anomalies keep evolving over time in order to adapt to new circumstances
with a general tendency toward hiding in normal looking activities.

2.2.4 ANOMALY DETECTION IN GRAPH-BASED DATA

Graph theory is widely used to represent complex structures that are
difficult to model (Gunasekara, Mehrotra, & Mohan, 2013; He & Zheng, 2013).
There are many relational data that can be modelled by graph theory. They
includes Internet, food web, blog networks, biological network, protein-protein
interaction, power grid, online social, and dating networks (Newman, Watts, &
Strogatz, 2002). Graphs provide a powerful representation of co-dependent
instances with robust connections between nodes. Graph theory has been used
widely to model online social networks (Kumar, et al., 2010; Snijders, 2011;
Xiang, Neville, & Rogati, 2010).

Graph-based anomaly detection algorithms (Aggarwal & Yu, 2001; Akoglu,
Tong, Vreeken, & Faloutsos, 2012; Breunig, Kriegel, Ng, & Sander, 2000;
Chandola, et al., 2009; Chaudhary, Szalay, & Moore, 2002; Das & Schneider,
2007; Ghoting, Parthasarathy, & Otey, 2008; Müller, Assent, Iglesias, Mülle, &
Böhm, 2012; Müller, Schiffer, & Seidl, 2010; Papadimitriou, Kitagawa, Gibbons,
& Faloutsos, 2003; Xiang, et al., 2010; Ye, Parthasarathy, & Tatakonda, 2011) can
be categorised into two groups based on the types of graph they used: static or dynamic (Chakrabarti, Faloutsos, & McGlohon, 2010). These algorithms are able to work with simple and attributed (node-/edge-labelled) graphs to detect anomalies. Extracting effective and meaningful features are the key steps for static-based algorithms which work on simple graphs (Henderson et al., 2011). The effective features (e.g. number of edges and node, total weight) should be able to generate a model, to be computed fast, and to be interpreted easily. The static-based algorithms include distance-based (model-based), density-based, and clustering-based. The obtained models learned by static-based algorithms are accurate as they have access to all the data for a reasonable period of time and have no time restriction to build the models. However they are prone to miss new kinds of anomaly if they are not shown in the examined datasets (Ye, et al., 2011). This weakness is not important when we deal with a local network of users (egonet) as they are most unlikely to have major topological changes during a short period of time (Hu & Wang, 2009).

Dynamic-based algorithms concentrate on the sequences of time-evolving graphs events under time series drift concepts which include graph distance changes (e.g. features, structures), or connectivity (e.g. phase transition) (Bay & Pazzani, 1999; Gama, Medas, Castillo, & Rodrigues, 2004). A time series drift is a sequence of events, computed at consecutive points in time spaced at time intervals. In these algorithms the graph similarity over time is computed in order to find any unusual changes. The scalability of these kinds of algorithms is always the problem. These thesis focuses on static-based algorithms as the changes are slow in a user network’s topology in online social networks.
2.2.4.1 ANOMALY IN STATIC LARGE DATA GRAPH

Methods for finding outliers in large data graph can be divided into different concepts based on density (Papadimitriou, et al., 2003; Sharma, Ram, & Singh, 2011), distance (Knorr & Ng, 1997; Knorr, Ng, & Tucakov, 2000; Knox & Ng, 1998), distribution (Barnett & Lewis, 1984; Solberg & Lahti, 2005), depth (Dang & Serfling, 2010; Johnson, Kwok, & Ng, 1998), and clustering (Duan, Xu, Liu, & Lee, 2009; Jain & Dubes, 1988; Jain, Murty, & Flynn, 1999; Tan, et al., 2005).

The density-based approach which was introduced by Breunig, et al. (2000), depends on the local density of the neighbourhood and uses LOF (local outlier factor) to detect outliers. Based on its computation formula, higher values of LOF correspond to outliers. LOF is based on a concept of a local density with respect to k-nearest neighbours. If the local density of an instance is lower compared to the local density of neighbours it can refer to an outlier. While standard density-based clustering algorithms (Agrawal, Gehrke, Gunopoulos, & Raghavan, 1998; Ester, Kriegel, Sander, & Xu, 1996; Wang, Yang, & Muntz, 1997) use two parameters including minimum number of objects required to form a cluster (MinPts) and volume, LOF uses only MinPts as parameter. In LOF the parameters identify a density threshold for objects to be clustered. LOF may suffer from detecting small outlying clusters as the MinPts has to be as large as the size of the clusters. Papadimitriou, et al. (2003) introduces Local Correlation Integral (LOCI) to overcome these problem by using probabilistic reasoning. Papadimitriou presents multi-granularity deviation factor (MDEF) as a new approach to deal with local density variations in the feature space to detect both isolated outliers,
and outlying clusters. LOCI identify an object outlier if its MDEF value deviate more than three standard deviations from the local averages.

The distribution-based approaches use standard distribution models as a metric to differentiate between normal and outliers (Hawkins, 1980). They call an object abnormal if it falls out of the model (Papadimitriou, et al., 2003). In this statistical approach outliers are defined based on the probability distribution. Distribution-based approaches follow the view that distribution of anomalies is different from distribution of normal instances which include majority. These methods use a model to estimate the distribution parameters of normal instances. They define instances anomalies if they are unlikely to be modelled by the parameters generated for the normal instances distribution.

In depth-based methods data is organised into layers according to some definition of depth such as depth contours. Depth-based methods assign a depth to each point in 2-d dataset and expect that outliers go to the centre of contours or shallow layers (Johnson, et al., 1998). These methods are weak in dealing with high dimensional data. Clustering-based approaches call an object anomaly if it fits objects to a cluster dominated by anomalies. Distance-based methods call an object \( \alpha \) in a data set \( P \) an outlier if at least a fraction \( \beta \) of the objects in \( P \) are more than \( \tau \) far away from \( \alpha \). They do not work well if a dataset includes sparse area (Breunig, et al., 2000). However using direct neighbours of users for analysis we can overcome the sparsity weakness of these powerful approaches.

Depth-based approaches categorise data objects in layers (e.g. contour) in the data space with the assumption that shallow layers are more likely to fit anomalies than the deeper layers. A main property of these kinds of approaches is
based on scaling-invariant of depth. To perform reliable detection, it is important that the instances with different depths can be detectable even under variance in layer scale. These instances should lie on the deeper layers which usually are the centre of each contour.

Graph anomaly detection techniques which are based on hypergraphs, can also be used in behavioural data (O'Madadhain, Hutchins, & Smyth, 2005; Silva & Willett, 2008). Hypergraphs is an important extension of graphs which allows edges to connect more than two vertices simultaneously. For instances for detecting anomalous meeting in a social network graph modelling, statistical analysis such as the distribution of meetings, and a variational Expectation-Maximization algorithm are used.

These aforementioned techniques are used in different domain applications such as games, signal processing with their own limitations with regard to the application. In extending these techniques of outlier detection to OSNs, issues such as accuracy arise. Therefore this research which develops these techniques to the OSN area concentrates on accuracy improvement using fuzzy logic.

2.2.4.2 GRAPH MINING ALGORITHMS

Graph mining algorithms are used to find clusters, patterns, classes and trends among given graph (Aggarwal & Wang, 2010; Ding, Katenka, Barford, Kolaczyk, & Crovella, 2012; Jimeng, Huiming, Chakrabarti, & Faloutsos, 2005; Tong & Lin, 2011). The algorithms (Apriori-style, HSIGRAM, VSIGRAM) looking for patterns can extract frequent patterns among several graphs or within a
single graph (Kuramochi & Karypis, 2001). The Apriori algorithm mines a linear set of items to find associations between them that makes these algorithms suitable for frequent sub-graph mining. Clustering algorithms can find clusters based on distance or similarity between edge labels (values) or the underlying structure of nodes (Flake, Tarjan, & Tsioutsiouliklis, 2004; Karger, 1999). Classification algorithms can classify a given graph using label propagation if there is a subset of nodes in a graph which are labelled, or graph classification if there is a subgraph in a graph dataset which are labelled (Kashima & Inokuchi, 2002; Kondor & Lafferty, 2002; Rossi, McDowell, Aha, & Neville, 2012). The algorithms used in these methods specifically need labelled edges as inputs and in many application domains such as OSNs this kinds of data is not available.

Noble and Cook (2003) introduced two techniques to look at the problem of graph-based anomaly detection using the anomalous substructure and anomalous sub-graph. These techniques are part of the Subdue algorithm (Holder & Cook, 2009) which is for detecting frequent patterns (substructures) within given graphs. Subdue uses Minimum Description Length (MDL) to compare the graph by replacing each substructure with a vertex which representing that substructure (Rissanen, 1989). Subdue uses this heuristic: let $G$ be entire graph, let $S$ be the substructure, let DL be description length (number of bits needed to encode data), and $F1(S,G) = DL(G|S) + DL(S)$ where, $DL(G|S)$ is the DL of $G$ after compressing it using $S$, and $DL(S)$ is the description length of the substructure. The best substructure is the one that minimises $F1$. Given database $D$ and set of models for $D$, minimum description length selects model $M$ that minimises $L(M) + L(D|M)$, where $L(M) = \text{length in bits: description of model } M$ and
Chapter 2: Literature Review

$L(D|M) = \text{length in bits: data, encoded by } M \ (e.g. = a_1X + a_0 \ OR \ a_0X^9 + ... + a_1X + a_0)$. Substructure-based anomaly detection techniques typically scan the whole graph to find the abnormal substructures. The problem is that infrequent substructures such as very large ones are expected to repeat rarely, although they are not abnormal. To overcome this problem, Noble and Cook (2003) suggest to use the maximum values of $F1(S, G)$ as an anomaly score.

Davis et al. (2011) present the YAGADA algorithm to search labelled graphs for anomalies using both structural data and numeric attributes. Their approach is based on discretisation (Fayyad & Irani, 1993; Kontkanen & Myllymäki, 2007) which assigns a categorical label $q_0$ to normal value and outlierness $q_i$ to all others $i$. They use different outlierness scores such as model fitting (Gaussian Mixture Model (GMM)), k-nearest neighbour (kNN) distance, cluster-based and Local Outlier Factors (CbLOF)) (He, Xu, & Deng, 2003). Another algorithm which takes numeric edge weights into consideration is OddBall (Akoglu, et al., 2010). It uses egonets to detect anomalous vertices with respect to their immediate neighbourhood; however, it ignores the label of vertex and edge. Therefore it cannot detect the weighted anomalous substructures.

Online social networks which are modelled by graph theory can include subgraphs (communities), different shapes of topology such as star and clique. Within the social context, community refers to the fact that people naturally tend to form groups, based on their interest inside work environment, family, trust and friends (Fortunato, 2010; Nguyen, Alim, Yilin, & Thai, 2013). Community outliers (Akoglu, Tong, Meeder, & Faloutsos, 2012; Bhat & Abulaish, 2013; Gao et al., 2010) are an important aspect of anomaly detection. Finding community
outliers in a network can be categorised into global or local. If only an object’s global information such as a user’s number of friends or number of interacted messages, is used without its network structure (e.g. friendship), the identified outliers are called global outliers and is called global outlier detection algorithm. If we only look at local information (i.e. neighbouring nodes information) we call them local outliers. Gao, et al. (2010) calls this method direct neighbour outlier detection algorithm. If we only take links into account to detect outliers, the method is called structural outlier detection algorithm (Xu, Yuruk, Feng, & Schweiger, 2007). It uses links to split data into communities and then detect outliers in each community. Detecting communities within direct neighbours can be done by simpler ways such as computing clique which is less computationally expensive and shows better results for a big data graph.

Based on online social network characteristics such as small world network, community of friends of friends can show a lot of information about a user under investigation. People naturally tend to form communities based on their similarity and common interests. This behaviour is true in online social networks (Yang & Leskovec, 2012; Yang, Guo, & Ma, 2010). The information which can be extracted from communities’ structure is useful to analyse the behaviour of a user and can lead to identifying anomalous behaviour.

Some clustering algorithms (Li, Nie, Lee, Giles, & Wen, 2008; Wang, Mohanty, & McCallum, 2006; Yang, Jin, Chi, & Zhu, 2009) use both data and link information to cluster objects, however, they assume that there are no outliers in dataset. Gao, et al. (2010) proposed a probabilistic model for community outlier detection integrating objects’ information and network structure. An Objects’
information is modelled by a mixed model in which K components is used to model normal community behaviour and one uniform component for outliers. A latent variable is used by the mixture model for each object to indicate its community. They use hidden Markov random field (HMRF) on the latent variable to incorporate links into the model applied on the information object. The problem with using this method for online social networks is the lack of object information which is hard to get due to privacy.

Eberle and Holder (2007), who worked on fraud detection, define anomaly within labelled graph as: “A graph substructure $S'$ is anomalous if it is not isomorphic to the graph’s normative (common) substructure $S$, but is isomorphic to $S$ with $x\%$ where $x$ is the percentage of vertices and edges that has to be changed to make $S'$ isomorphic to $S$ “. Their approach called GBAD (Graph-Based Anomaly Detection) which is based on SUBDUE graph-based knowledge discovery method and Minimum Description Length (MDL) heuristic (Grünwald, Myung, & Pitt, 2005). GBAD is divided into three separate algorithms: GBAD-MDL, GBAD-P and GBAD-MPS. The GBAD-MDL algorithm uses MDL to find the most common substructure in a graph, and look for similar patterns in the whole graph. The GBAD-P algorithm is a probabilistic model which uses MDL to find the common substructure in a graph. The GBAD-P finds the anomalies that happen as anomalous extensions of a common pattern. The GBAD-MDL algorithm finds the anomalies that are similar to the best substructure and differ only by the labels of a few nodes or edges. The GBAD-MPS algorithm uses MDL and cost transformer values to find the common substructure and determine how many changes are needed to match the best substructure. Applying GBAD to a
large graph data such as online social networks suffers from time computational complexity as they use an isomorphic concept to find anomalies.

Each user network (egonet) has a “hub” vertex representing the user itself, along with a vertex for each friend connected to the hub. To find an anomalous user network we are interested in determining how unusual each distinct user network is (Noble & Cook, 2003). The Subgraph-based anomaly detection concept is the centre of our proposed approach in this research. However, we are looking for the degree of anomalousness rather than the frequency of patterns repeated in graph data.

This section was about finding clusters, patterns, classes and trends among a given graph. The frequent patterns can be extracted among several graphs or within a single graph. The clusters can be found based on distance or similarity between edge labels (values) or the underlying structure of nodes. Using edge’s label propagation, the classification algorithms can classify a subset of nodes in a given graph. Davis, et al. (2011) present an algorithm to search labelled graphs for anomalies using both structural data and numeric attributes. The algorithms, which are used in these methods, specifically need labelled edges as inputs. However, in many applications domain such as OSNs these kinds of information is not available.

Noble and Cook (2003) used Subdue algorithm which is for detecting frequent patterns (substructures) within given graphs. Subdue uses Minimum Description Length (MDL) to compare the graph by replacing each substructure with a vertex representing that substructure. These techniques usually probe the entire graph to find the abnormal substructures. In addition to time complexity of
these techniques for a large graph such as OSNs, those infrequent substructures such as very large ones are expected to repeat rarely, although they are not abnormal. The other problem is that the fuzziness of anomalous behaviours was not considered in these methods.

2.2.5 SUMMARY

Limited work has been done on finding anomalies in online social network due to low accuracy, complexity, privacy issues, lack of labelled datasets and lack of sufficient information. Accuracy, an important performance measure in any anomaly detection technique, becomes one of the key measures in this thesis.

Graphs are widely used to represent complex structures that are difficult to model. Graph-based anomaly detection algorithms can be categorised in two groups based on static and dynamic graphs. The static-based algorithms include distance-based (model-based), density-based, and clustering-based. The accuracy of the obtained models learned by static-based algorithms is more accurate as they have access to all data for a reasonable period of time and have no time restriction. However they are prone to miss new kinds of anomaly if they are not shown in the examined datasets. This weakness can be overcome by using a local network of users as they are most unlikely to have major topological changes during a short period of time. Moreover, the existing static-based algorithms have not been developed or customised for OSNs, in which users’ behaviours are seen as fuzzy variables. Dynamic-based algorithms focus mainly on the sequences of time-evolving graphs events under time series drift concepts. In these algorithms the graph similarity over time is computed in order to find any unusual changes. The scalability of these kinds of algorithms is always the problem. This thesis
concentrates on static-based algorithms as the changes are slow in a user network’s topology in online social networks.

The Subdue algorithm uses MDL to replace each substructure with a vertex representing it, for detecting anomalies within data graph. These techniques usually probe the entire graph to find such abnormal substructures. In addition to time complexity, infrequent substructures such as the very large ones are considered as anomalies because they are rarely repeated.

User’s behaviours in online social networks have been categorised in the literature as passive, active, and moderate. The typical user for instance in Facebook was a categorised as moderately active user in terms of sending requests, posting content, and liking the content of their friends. In the literature, online social networks were modelled with graph theory, characterising outliers as star or near-star, clique or near-clique, heavy vicinity, and dominant edge. From a sociological view online social behaviours can be a reflection of offline behaviours.

2.3 CLUSTERING ALGORITHMS

Clustering algorithms divide data into groups called subsets or clusters or categories. A cluster can be expressed as a group of data with high internal homogeneity and high external disjunction (Jain & Dubes, 1988). Clustering algorithms are used for studying unlabelled data by either establishing a set of different collections or building a hierarchical structure. The clustering objective is to find hidden data structures to approximate characterisation of given samples.
The number of clusters can be given to algorithms or can be found by the algorithms automatically (Cherkassky & Mulier, 2007). Clustering algorithms development processes includes several steps such as pre-processing, algorithm design, and evaluation. These processes are defined as shown in Figure 6 by Xu and Wunsch (2005).

![Figure 6. Clustering Process](image)

As defined by Jain, et al. (1999), Bishop (1995), and Webb (2003), in clustering algorithms feature selection concerns finding unique features and feature extraction concerns transforming existing features to new practical features. Feature selection plays an important role in improving the quality and the effectiveness of the generated clusters. The features should be able to differentiate instances, to resist to noises, and to be simple. To adapt or design an appropriate clustering algorithm it is crucial first to understand the characteristics of problems. The second step is to find a suitable similarity measure and an objective function (Kleinberg, 2003). The base of all similarity measures is on grouping similar instances together. However, there is no ubiquitous similarity measure that can be
applied to all problems. Therefore, different problems need different customised similarity measures.

Cluster validation is another important step in which objective evaluation strategies should be taken to provide users a degree of confidence about the generated clusters. Cluster validation can help to find the right number of clusters, to verify the meaningfulness of the clusters, and to find a suitable algorithm for our problem. There are three different evaluation tests: external indices, internal indices, and relative indices (Bandyopadhyay & Maulik, 2001; Bezdek & Pal, 1998; Chen, 2009; Halkidi, Batistakis, & Vazirgiannis, 2002a, 2002b; Leung, Zhang, & Xu, 2000; Levine & Domany, 2001). The external indices are derived from prior information on the data, the internal indices are based on the clustering structure came from the original data, and the relative indices use both the internal and the external indices. A user interpretation of the generated clusters is the last step of clustering evaluation process which provides deep insights to the original data. In some cases additional analyses and experiments are needed to confirm reliability of discovered knowledge.

2.3.1 SEMI-UNSUPERVISED CLUSTERING

Clustering is traditionally seen as unsupervised learning. It is usually used when no labelled data or any other information is available to find proper connections between instances and predefined classes. However, recently new clustering algorithms use some kind of supervision to adjust generated clusters (Grira, Crucianu, & Boujema, 2004) so they can be called semi-supervised or
constraints. In some cases, in addition to similarity knowledge used by unsupervised clustering we have limited extra prior information such as small labelled datasets which can be utilised to adjust and improve the clustering output. However, this extra information is not enough to be used by supervised learning methods such as classification. This process is called the semi-supervised clustering method.

### 2.3.2 UNSUPERVISED CLUSTERING

Unsupervised clustering techniques that do not need training data are widely used (Luo, Wang, & Zhang, 2003). These kinds of techniques implicitly assume that normal instances are more common than anomalies in the test data. In these methods, the false alarm rate will be increased if this assumption is not accurate. Several semi supervised techniques can work in an unsupervised mode by employing an unlabelled dataset sample as a training datasets if the number of anomalies is very small compare to the whole datasets (Guthrie, et al., 2007).

### 2.3.3 FUZZY CLUSTERING

Fuzzy clustering is a clustering technique in which a dataset is grouped into different clusters where every data point belongs to every cluster to a certain degree (Gath & Geva, 1989; Xue, Shang, & Feng, 2010). For example, a certain data point that lies close to the centre of a cluster will have a high degree of membership to that cluster and another data point that lies far away from the centre of a cluster will have a low degree of membership to that cluster.
2.3.4 SUMMARY

Clustering techniques are widely used in anomaly detection. This type of technique implicitly assumes that normal instances are more commonly present than anomalies in the test data (Guthrie, et al., 2007). They have been divided into three categories: unsupervised, semi-supervised, and fuzzy, based on the assumptions used to form clusters (Chandola, et al., 2009). With semi-supervised clustering methods, if a new instance does not belong to any of the clusters or if it is not close to any cluster, it is considered to be an anomaly. For unsupervised clustering methods, post-processing is needed after clustering in order to detect an anomaly. This post-processing includes determining the size of the clusters and the distance from each cluster’s centroid.

Clustering algorithms identify an instance as an anomaly if (1) it does not fit into any cluster, or (2) it belongs to a small cluster, or (3) it belongs to a low density cluster, or (4) it is distant from all other instances within the same cluster. The unsupervised and semi-supervised algorithms assume that normal data instances fit in a cluster and anomalies appear as outliers. They also assume that the normal data instances appear cohesively in their nearest cluster centroid and the anomalies appear a long way from their nearest cluster centroid. Moreover they assume that the data instances are normal if they fit into large and dense clusters, and that they are anomalies if they fit into small or sparse clusters. The fuzzy algorithms suppose that all data instances can belong to all clusters with different degrees of membership (the degree of being correctly fitted in each cluster). Normal instances have the highest membership degree and anomalies have the lowest.
Based on some clustering techniques’ drawbacks, it is better to combine clustering techniques with other techniques such as fuzzy logic to achieve better performance for anomaly detection in OSNs. This combination becomes important when dealing with the fuzzy nature of the data found in online social networks data. These are three main drawbacks of clustering-based techniques: (1) they are not optimised to find anomalies because the main objective of a clustering algorithm is to find clusters; (2) if the anomalies in the data form clusters by themselves, these techniques will not be able to detect them; the problem will be worse if the distances between any two instances becomes quite similar; (3) clustering algorithms may not produce any meaningful clusters to extract rules in order to detect similar patterns.

The existing fuzzy clustering algorithms can generate only a matrix of fuzzy membership degree. This can be used as an output to show the degree of belonging or membership of each instance to the generated clusters. However, in order to employ each cluster’s membership degree to generate rules using a fuzzy inference engine, these clusters need to be represented by a fuzzy membership function.

2.4 FUZZY LOGIC

Anomalies can be treated as a multiple-valued logic problem in which we have a degree of truth rather than only two possible values (i.e., "anomaly" and "normal"). One of the most popular logics in dealing with this kind of problem is fuzzy logic (Zadeh, 1965). Fuzzy logic or fuzzy inference system (FIS) is based
on approximation instead of typical predicate logic. This is adopted by researchers in several areas such as intrusion detection, clustering, financial fraud detection (Cornelis, Lu, Guo, & Zhang, 2007), and networks traffic (Goudarzi & Hassanzadeh, 2006). Fuzzy logic techniques have been employed in the computer security field since the early part of 90’s (Idrsi & Shanmugam, 2006). For instance, applying fuzzy logic to the intrusion detection area provides flexibility to the uncertain problem of intrusion detection and allows better complexity for these techniques. By using fuzzy logic the problem of a sharp boundary between anomalies and normal instances are solved.

A fuzzy inference system, also called a fuzzy rule-based system or fuzzy model, is composed of three blocks: (1) Fuzzifier, which determines degree of membership, transforms the crisp inputs (distinct or exact inputs) into linguistic variable inputs using membership functions (MFs); (2) Inference Engine using rules expressed in the form of IF-THEN statements; and (3) Defuzzifier that transforms the linguistic variable outputs into crisp output. An instance can be an anomaly or normal if its anomaly score falls into a certain intervals. These intervals are represented by membership functions. This semi-supervised technique differentiates between normal and anomalous instances by rules which have been generated by the fuzzy inference engine using membership functions (MFs). In fuzzy logic, there are different shapes of membership functions such as triangular, trapezoidal, piecewise linear, Gaussian, or singleton.

The reason for using fuzzy logic is the fuzzy nature of the problems. For instance, defining the boundary between normal and anomalous instances can be considered as a problem with the fuzzy nature. Therefore, we propose to use fuzzy
logic to interpret and discover the degree of abnormality of a user network based on the lower and upper bound of a given anomaly score. The use of fuzzy logic allows us to handle the fuzzy features of online social networks such as community, type of relationship, and user network’s topology.

2.5 RESEARCH GAP

Methods for finding outliers in data graphs can be divided into distance-based, distribution-based, and clustering based. The mechanisms of these approaches were reviewed in order to develop novel methods for detecting anomalies in online social networks. Distance-based methods call an object anomaly if they sit far away from model. Distribution-based use distribution models to call an object abnormal if it falls out of the model. Clustering-based approaches call an object an anomaly if it fits objects to a cluster dominated by anomalies. These methods do not work well if datasets include fuzzy or sparse instances. Storing and manipulating sparse data face time and space complexity issues. Moreover, fuzzy instances need to be treated with multi-levels logic in order to achieve a better performance in terms of accuracy. These weaknesses can be overcome by using fuzzy logic and selecting features from the direct neighbours of users (egonet) instead of the whole graph.

To avoid the sparsity, such as the other egonet-based approaches, the proposed methods are applied only on the direct neighbours of a user rather than applying on all the users’ networks. Therefore, for finding anomalies, we use the extracted features from users’ egonets in order to discover common patterns, and
generate a model which can alleviate the sparsity problem that exists in current algorithms.

In order to achieve better accuracy, anomalies score need to be computed under uncertainty, for instance, how much a detected pattern is similar to real anomalies. Fuzzy set theory lends itself well to these kinds of problems. A fuzzy logic-based conceptual framework, which is devised for multi-levels prediction, is used to generate rules for detecting anomalies. The scoring used in the proposed algorithms and prediction mechanisms is represented by linguistic variable to overcome uncertainty.

The well-known anomalous topologies such as star and clique are used by existing approaches as a ground-truth for detecting anomalies. These connection-based methods rely merely on the connections between users local network. Based on the limitations of these methods, the best strategy is to combine detection techniques in a hybrid way with multi-levels logic such as fuzzy theory. By using a fuzzy-based hybrid approach we can achieve prediction accuracies that are comparable to or even better than existing approaches. In order to leverage the research gaps, the proposed methods need to model online social networks using graph theory and directly apply to the proposed methods.

From the literature review, the following research gaps, which are based on the structural approaches, are identified as follows:

- Lack of customised structural-based anomaly detection techniques for online social networks due to issues such as complexity, low accuracy, privacy, lack of labelled datasets and lack of sufficient information. The
majority of work focuses on behaviour-based techniques, which consider the dynamic usage behaviour of users, are technology dependent. The existing works on structural-based techniques suffer from low accuracy or missing outliers in subnetworks with a high number of nodes and edges. To improve the accuracy of the previous methods, first, we decide to develop the new structure-based methods as the structural meta-data cannot be denied by the users. Second, we employ a hybrid approach that utilise different types of techniques such as distribution, clustering, and fuzzy logic.

• Lack of fuzzy-based structural anomaly detection techniques for online social networks which can handle the problems of fuzzy behaviours and characteristics such as relationship and community. The quantitative structural features of online social networks can be best represented as a fuzzy variable. Therefore they can be treated as a multiple-valued logic problem in which we have a degree of truth rather than only two possible values.

2.6 SUMMARY

This thesis addresses the problem of anomaly detection in online social network. This chapter covers the topics including anomaly detection challenges and techniques, online social network analysis, community detection, clustering techniques, and fuzzy logic. These topics play important roles in anomaly detection as well as in the proposed approaches. The chapter begins with anomaly
detection definition and techniques, advantages and disadvantages of each technique, existing challenges, type of data, and methods for reporting anomalies. Recent works on anomalies detection are categorised into different groups based on the techniques being adapted. Different groups use different assumptions for detecting anomalies. The literature shows that most anomaly detection techniques are made for a specific application by adapting and customising existing machine learning techniques. Limited work has been done on finding anomalies in online social network due to low accuracy, complexity, privacy issues, lack of labelled datasets, and lack of sufficient information. Accuracy is an important performance measure in any anomaly detection technique which becomes one of the key measures in this thesis.

In the literature online social networks were represented with graph theory in which the outliers are identified as star or near-star, clique or near-clique, heavy vicinity, and dominant edge. Anomaly detection algorithms need a model (e.g. normal patterns in the data) to compute an anomaly score of a user on the basis of the deviations from the model. For instance, the model can include a Gaussian mixture model or a regression-based model, or a statistical-based model (Aggarwal, 2013). All the models have their own assumptions regarding normal behaviour. The anomaly score is calculated based on how well instances fit into the generated model. The choice of the model is important as an inaccurate model can lead to weak outcomes. For instance a regression-based model may not work accurately if there is a strong relationship between independent instances that leads them to be clustered. In reality, the choice of the model often depends on the expert’s understanding of the problem, and the application using the model.
Deviation from the model can be varied for different applications domains. Therefore, the best model for a particular dataset can be built after adjusting for and evaluating the different properties of that domain.

In this Chapter, online social network characteristics, user behaviours, anomaly detection challenges, type of possible anomalies, labelling problem, and modelling were also discussed. User’s behaviours in online social networks were categorised as passive, active, and moderate. According to the literature, the typical user for instance in Facebook was categorised as moderate active user in terms of sending requests, posting content, and liking the content of their friends. From a sociological view in offline and online social networks, it is likely that two friends of a user are also friends of each other.
Chapter 3: Multi-Layer Framework

This chapter is concerned with the research design employed in developing the proposed anomaly detection methods in this thesis. This research introduces a multi-layered framework and the proposed methods which are applied to the datasets generated from online social networks in order to detect anomalies.

An object is, intuitively, an anomaly if it varies considerably from its neighbours. Based on definitions of “varies from neighbours” and neighbourhood, and whether or not the variation is substantial, we will have different sets of anomaly definitions. Different anomaly definitions need to have different techniques to deal with them. Therefore some of the well-known techniques based on distance, distribution, and clustering are developed. Detecting outliers using the structural connectivity of users’ network is the specific problem which we study in this research. While users can hide their identity by supplying false information and can deceive analysts, analysis of users’ connectivity can help to spot anomalies more accurately. We explore, develop, and test different novel approaches to the anomaly detection problem within the proposed framework. These methods belong to the category of statistical and semi-supervised learning methods. The remainder of this chapter is organised as follows. The basis for each layer of the framework including dataset characteristics and evaluation measures is discussed. The description of each layer presents a quick overview of the concepts used and shows the overall view of the methodology.
3.1. FRAMEWORK OVERVIEW

The first layer of the proposed framework shown in Figure 7 is about pre-processing and modelling an online social network’s data. Data collection processes are usually faced with out of control issues which result in having noisy data such as missing values, wrong data combinations, and duplicate data. These noisy and unreliable data instances can mislead a training process to produce a training set during the knowledge discovery phase. The data pre-processing steps we took comprised normalisation, cleaning, transformation, integration, reduction, feature extraction and selection. We model the three real-life datasets using various graph theory properties. The appropriate features are then extracted for each node in the graph. These features are used as input data for all the methods developed in this research. The second layer of the framework includes various methods that handle the problem of anomaly detection in online social networks differently. Distance-based, distribution-based, and clustering-based are three concepts that are used in this framework. These methods approach the anomaly detection problem from different angles in order to achieve a better accuracy and simplicity. In the first method our focus is on the differences between the observed value and the predicted value generated by a power-law regression model as well as by orthogonal projection. The objective of this approach is to improve the accuracy of the OddBall method (Akoglu, et al., 2010) and overcome its limitation. To do that, new graph metrics (average betweenness centrality), a power-law regression model, the distance (residual) to the model, the orthogonal projection of points on the model and a clustering algorithm are introduced to model and find anomalies. Distance to the model shows how far a point sits from
the model. Based on this distance an anomaly score for an instance is computed in order to verify whether the instance is anomalous.

In the second method, we introduce new graph metrics (starness, cliqueness) that are continuous variables. We use a distribution-based approach to find common patterns which can lead to the detection of anomalies. The objectives of a statistical model are to improve the accuracy of the first method and to deal with

Figure 7. Proposed Framework
new metrics. In this approach, we use the Gaussian mixture model and fuzzy logic to differentiate between anomalous and normal instances.

In the third method, the concentration is on the employment of a clustering-based approach to the graph metrics, introduced in the first approach with the objective of increasing overcoming the limitation such as false alarm. In this data mining technique we utilise fuzzy clustering to locate anomalies. The membership degrees generated by the fuzzy clustering are then represented by fuzzy a membership function to define the boundaries of anomalies.

3.2. LAYER-ONE: PRE-PROCESSING, MODELLING, IDENTIFYING EGONETS, AND SUPER-EGONETS

This section looks at the pre-processing of the graph data collected from three real-life datasets including Facebook, Flikr, and Orkut. These datasets have been previously studied in literature (Cha, et al., 2008; Mislove, et al., 2007; Viswanath, et al., 2009). Each dataset represents relationships that exist in the social network between nodes in the form of a list of adjacencies. These datasets, collected using crawling techniques (Cha, et al., 2008; Mislove, et al., 2007; Viswanath, et al., 2009), show different characteristics that have been previously generated by other researchers for measuring and analysing online social networks (Mislove, et al., 2007).

This layer is concerned with extracting the topology and structure of each user network and representing them in the form of a graph. This is one of the most time-consuming tasks of this research, as it often needs several iterations
involving human interference. The key purpose of this layer is to provide a suitable representation of online social network data that enables the proposed algorithms to detect anomalies. The first step includes extracting and building each user network (egonet), which needs a lot of pre-processing steps. It needs to parse the given dataset to find all directed neighbours of a user and to convert this to a graph structure. Examining the generated graphs shows several unconnected and isolated users that should be identified and removed from the data as they have no effect in processing. In some cases when needed by the proposed methods, data transformation is used to improve performance. For instance, normalisation is used in the distribution-based method to bring all anomaly scores into the same range. Moreover, in the distance-based approach, orthogonal projections of instances are used to improve the quality of the clustering algorithm. The orthogonal projections of instances are calculated using minimum distance to the curve. The outputs points are the closest points identified along the curve to each of the points. These points were mapped to the given curve in terms of the minimum (Euclidean, 2-norm) distance to the curve.

3.2.1. MODELLING OSNS USING GRAPH THEORY

Given an OSN, we model it as a graph $G = (V, E)$ and build a sub-graph for each node so the local features can be extracted in order to represent the node. The graph $G$ consists of a set $V$ of vertices (nodes) and a set $E$ of edges ($E$) where each $e \in E$ is an unordered pair of distinct vertices. Vertices and edges represent users and their relationship respectively. We model each user as a single node (ego), its 1-level neighbourhood (an egonet) and its 2-level neighbourhood (a
super-egonet). For sampling we randomly walk through given network graph data and from an initial starting node. To make an egonet it travels over the network node by node to discover the entire direct neighbours of the current node. In the next step, we go through all the direct neighbour nodes to discover their friends to make a super-egonet.

Formally, for a node \( j \) in a graph \( \mathcal{G} = (\mathcal{V}, \mathcal{E}) \) where \( j \in \mathcal{V} \), and \( \mathcal{V}^{\text{egonet}_j} = \{j, j_1, j_2, \ldots, j_k\} \), \( j_1, \ldots, j_k \) are the direct neighbors of node \( j \), the egonet and super-egonet are defined as follows:

\[ \text{egonet}_j \text{ is the subgraph of } \mathcal{G} \text{ induced by } \mathcal{V}^{\text{egonet}_j}; \text{ that is, it includes all edges from the original graph between this subset of vertices. In other words, an egonet is a sub-graph in } \mathcal{G} \text{ that consists of the user’s node as the central node and all of its direct neighbours connections.} \]

Super-egonet is defined as the user’s egonet and the egonets of all its neighbours. For node \( j \), super-egonet \( j \) = \( \{\text{egonet}_j, \text{egonet}_{j_1}, \ldots, \text{egonet}_{j_n}\} \). A super-egonet is the sub-graph consisting of the ego, all of its immediate neighbours, and all neighbour of neighbours and their edges. The \( \text{deg} \) is a function returning the number of edges in an egonet, that is, the number of edges within the egonet connecting the nodes together.

Each user is represented by extracted features that their egonet and super egonet reveal. The formal definition of the extracted features and required definition can be found in the following sections.
3.2.2. FEATURES EXTRACTION

Most of the time all attributes used for building a model may not make a meaningful contribution: this can result in having an inaccurate model due to the presence of redundancy and interdependency (Cao, Shen, Sun, Yang, & Chen, 2007). Noise coming from irrelevant attributes can increase the size of the model and the time and space complexity of the algorithmic processing. On the other hand, it might be possible to have a dataset in which there is a set of attributes with the same underlying feature. Bringing these same features together in the training phase can skew the algorithm and decrease the model accuracy. To deal with noise, underlying correlation, high dimensionality, and the effectiveness of the model, we propose feature selection (selecting the maximum related attributes) and extraction (integrating attributes into a new reduced group of features (Blum & Langley, 1997; Huan & Lei, 2005); sometime only one or two features).

Social network components include users, friendship ties, communities, and the information passing through them. OSN analysis methods have been commonly used in identifying relationships between social entities, as well as patterns and implications of these relationships (Wasserman & Faust, 1994). Some important attributes that can be modelled include transitivity, user’s network information (e.g. number of nodes and edges; betweenness centrality), structural connectivity such as presence of star and clique subnetworks, community cohesiveness, and isolated users. Some of these features have been investigated by previous researchers and have been introduced as metrics for detecting anomalies. The graph-theory-based model provides a structural and conceptual view of a social network for formal study. A statistical theory model is
used to understand network characteristics by analysing theoretical propositions against the network. This theory can be accepted or rejected by the information obtained from the dataset.

In this thesis, we analyse and model online social networks characteristics using a combination of graph and statistical theory. We extract meaningful structural features using graph theory. We use statistical theory to analyse minority and majority such as starness, cliqueness, and “friends of friends are often friends”. Our hypothesis is based on the fact that the majority of users in online social networks follow the “friends of friends are often friends” pattern.

The features we deal with include betweenness centrality, average betweenness centrality, number of nodes and edges, community cohesiveness, starness degree, and cliqueness degree which can help to detect anomalies within the graph data. For instance, measures such as betweenness centrality have been used for finding the central users in social networks graphs (Kas, Wachs, Carley, & Carley, 2013; Pfeiffer III & Neville, 2011). It shows the level of control over the flow of information and relationships. Nodes with higher centrality can be called the prominent role players in the network. Betweenness centrality, when compared to the other, such as degree and closeness centrality, are much more informative. Degree centrality computes the number of connections and closeness centrality shows how quickly a user can travel through the network.
3.1.1.1 ONLINE SOCIAL NETWORK CHARACTERISTICS

The common parameters in any OSNs include degree distribution, neighbourhood, connectivity, shortest paths, clustering coefficients, shared neighbours, and betweenness centrality. One of the common measurements of every online social networks topology is to look at their degree distributions. Online social networks are also called power-law networks or scale-free networks. It means new nodes tend to connect to the nodes with large degree. The majority of social networks in nature have a power-law degree distribution: nodes with small degrees are most frequent. In online social networks people also tend to be assertive in their relationships with others; for example, high degree nodes tend to be connected to high degree nodes in the network (Newman, 2003).

Analysis levels of OSNs include micro-level (e.g. users), meso-level (e.g. organisation), and macro-level (e.g. large-scale and complex networks) analysis (Kadushin, 2012; Riketta & Nienaber, 2007; Strogatz, 2001). At the micro level each user egonet, including users and their direct neighbours, is analysed. Our experiments, as well as other research, show that majority of users follow the “friends of friends are often friends” pattern (Akoglu, et al., 2010). A minority of users follow either the “cliques or near-cliques” pattern (all the neighbours connected) or the “stars or near-star” pattern (mostly disconnected) as shown in Figure 8, Figure 9 and Figure 10. Akoglu, et al. (2010) verified these hypotheses against different datasets such as Auth2Conf (#N=421k, #E=1M), Postnet (#N=223k, #E=217k), Don2Com (#N=1.6M, #E= 2M), and Enron(#N=36K, #E=183K).
Figure 8. The “Friends of Friends are Often Friends” Pattern Network

Figure 9. Near-Star Topology
Figure 8 shows the normal pattern in most OSNs: “friends of friends are often friends”, and also shows different clusters using the circles. The connections (edges) between them are used as a similarity measure which put them together in
a cluster. The measure of similarity is the existence of the edge between the friends of a user. In other words two friends can be in the same cluster if there is an edge between them. Figure 9 shows the star pattern, which is considered as an anomaly in most OSNs. In this kind of anomaly a user indiscriminately connects to the other users without any connections among them. Figure 10 shows the clique pattern, which also reflects anomaly behaviours in most OSNs. In this kind of topology all friends of a user are connected to each other.

Local patterns in an egonet, as shown in Figure 11, refers to the pattern of each user network, showing the direct friends and which of these friends have connection to each other. The local patterns can give us a good and reliable picture of network topology types at a global level. Statistical analysis of the local patterns facilitates a comprehensive understanding of the patterns, followed by statistical analyse of the majority in order to generate rules. These generated rules are then used for identifying the anomalies in datasets. Using local patterns to locate influential entities within online social networks (OSNs) is one of the key concepts of finding anomalous topology such as the star (Aral & Walker, 2012). For instance a company looking to sell their products using OSNs usually targets influential individuals in the network and gives them free samples. This viral marketing tries to convince other users to buy their products by using the relationships power of influential entities (Zhang, Zhu, Wang, & Zhao, 2013).

3.1.1.2 CENTRALITY METRICS

Centrality of users is not an individual attribute; instead it is concerned with relations with the other users. The overall view of the online social networks
structure suggests that the different levels of centrality are a result of variations in the patterns of connections among the users. High centrality of a user in a network usually comes with high degree and high betweenness properties of the network. In the star and clique topologies, centrality of the user under investigation tends to be higher and lower values respectively.

Different measures have been used to characterise the centrality of a node to describe the node behaviours. Closeness centrality provides a global view about a node in the network, while betweenness centrality is defined with reference to the local view of a node. Among these measures, betweenness centrality is able to identify nodes that are highly interactive with other nodes in terms of flowing information.

In this research, we utilise betweenness centrality (Eq. 3.1), a measure of user centrality, as an extracted feature to modelling uncommon behaviours such as starness and cliqueness. The reason behind this is that it takes into account the global as well as the local features of a network. Local features concentrate on extracting rules within egonets; global features focus on generalising the extracted rules as pattern rules to differentiate people’s online behaviour. For instance, the local and global view of user’s friendship topology can be utilised to generate a model which can characterise the user relationship. According to the anomaly definition (Chandola, et al., 2009), if the friendship pattern of a user follows the common pattern defined by the global features, it is called “normal”, otherwise “anomalous”. The betweenness centrality ($\mathcal{B}_t$) of a node in a graph is the number of shortest paths, between all the pairs of nodes within that graph, that go through that node. In the social network theory context, users differ in their impact on the
overall network. The influential nodes are identified by betweenness centrality (Freeman, 1979; Friedkin, 1991), a measure of how many shortest paths pass through the node, which indicates how highly or densely connected they are to their neighbours.

**Definition 1 (Betweenness centrality).** The betweenness centrality of a vertex \( \hat{j} \in V(G) \) is

\[
\mathcal{B}_t = \sum_{s \neq \hat{j} \neq d} \psi_{\hat{j}}^{sd} / n_{sd} \quad \hat{j}, s, d \in V
\]  

(3.1)

where \( \psi_{\hat{j}}^{sd} \) is the number of shortest paths between nodes \( s \) and \( d \) passing through node \( \hat{j} \), and \( n_{sd} \) is the total number of shortest paths from \( s \) to \( d \) within the egonet. Each new edge defining a new shortest path will reduce the \( \mathcal{B}_t \) of the central node.

**Average Betweenness Centrality** is computed as the proportion of all geodesics (shortest path) that include the node under study with the number of nodes within an egonet. This measure takes the geodesics path of each node into account in order to provide a degree of starness and cliqueness of a given egonet. It is worth mentioning that the average betweenness centrality of an egonet is decreased as new edges are added. The range of value of these metrics depends on the topology of user’s network. For instance, the higher value of this metric for the central node within a network shows better distribution of connections among the other nodes.

**Definition 2 (Average Betweenness Centrality).** The average betweenness centrality of \( egonet_{\hat{j}} \) is:
\[ \sigma_{j}^{cc} = \frac{BC_j + \sum_{j=1}^{n} BC_j}{n}, \quad \text{where} \quad n = |\mathcal{V}_{\text{egonet}_j}|, j \in \mathcal{V}(\mathcal{G}) \quad (3.2) \]

**Definition 3 (Closeness Centrality).** The closeness centrality of \( \mathcal{V}_{\text{egonet}_j} \) is:
the inverse of the distance of each node to every other node in an egonet.

\[ \sigma_{j}^{cc} = \left[ \sum_{j=1}^{g} d(n_i, n_j) \right]^{-1}, \quad \text{where} \quad g = |\mathcal{V}_{\text{egonet}_j}|, n_x \in \mathcal{V}(\mathcal{G}) \quad (3.3) \]

**Definition 4 (Eigenvector centrality).** The eigenvector centrality of \( \mathcal{V}_{\text{egonet}_j} \) is a measure of the centrality of a node in an egonet which assumes that edges to high-scoring nodes contribute more to the score of the node under investigation. Relative scores are allocated to all nodes in the egonet.

\[ \sigma_{j}^{ec} = \mu \sum_{i=1}^{n} a_{ij}x_i, \quad \text{where} \quad a_{ij} \in A = \text{similarity matrix}, \quad n = |\mathcal{V}_{\text{egonet}_j}|, x \text{ is corresponding eigenvector} \quad (3.4) \]

**Definition 5 (Clustering coefficient).** The clustering centrality of \( \mathcal{V}_{\text{egonet}_j} \) is a measure of how much nodes in an egonet tend to cluster together.

\[ \sigma_{j}^{clc} = \frac{2e}{n(n-1)}, \quad \text{where} \quad e = |\mathcal{E}_{\text{egonet}_j}|, n = |\mathcal{V}_{\text{egonet}_j}| \quad (3.5) \]

Betweenness centrality and the clustering coefficient work similarly in terms of finding “one’s friends are also friends of each other” patterns. Betweenness centrality for a node is defined to be the shortest path. Betweenness centrality, when compared to others such as degree and closeness centrality, can give us more information about star topology. Degree centrality gauges the number of connections and closeness centrality gauges how quickly a user can
access information. The clustering coefficient fails to find the most influential nodes in a network. Clustering coefficient is a measure of how the nodes in a graph are paired together.

![Full Star and Full Clique](image)

**Figure 12. Full Star and Full Clique**

<table>
<thead>
<tr>
<th>Node</th>
<th>Degree</th>
<th>Betweenness Centrality</th>
<th>Closeness Centrality</th>
<th>Eigenvector Centrality</th>
<th>Clustering Coefficient</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4</td>
<td>6.000</td>
<td>0.250</td>
<td>0.200</td>
<td>0.000</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>0.000</td>
<td>0.143</td>
<td>0.200</td>
<td>0.000</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>0.000</td>
<td>0.143</td>
<td>0.200</td>
<td>0.000</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>0.000</td>
<td>0.143</td>
<td>0.200</td>
<td>0.000</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>0.000</td>
<td>0.143</td>
<td>0.200</td>
<td>0.000</td>
</tr>
</tbody>
</table>

#Nodes = 5, #Edges =4

As shown in Table 1 and Table 2, which are related to Figure 12, betweenness centrality compared with other graph metrics shows more details
about users’ egonets. For instance, betweenness centrality in Table 1 showing node 1 is in the centre of the egonet, with no connection among the other direct neighbours. Moreover, when we look at the number of nodes and edges, we can see the relation that becomes a pattern.

Table 2. Full Clique Analysis

<table>
<thead>
<tr>
<th>Node</th>
<th>Degree</th>
<th>Betweenness Centrality</th>
<th>Closeness Centrality</th>
<th>Eigenvector Centrality</th>
<th>Clustering Coefficient</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4</td>
<td>0.000</td>
<td>0.250</td>
<td>0.200</td>
<td>1.000</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>0.000</td>
<td>0.250</td>
<td>0.200</td>
<td>1.000</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>0.000</td>
<td>0.250</td>
<td>0.200</td>
<td>1.000</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>0.000</td>
<td>0.250</td>
<td>0.200</td>
<td>1.000</td>
</tr>
<tr>
<td>5</td>
<td>4</td>
<td>0.000</td>
<td>0.250</td>
<td>0.200</td>
<td>1.000</td>
</tr>
</tbody>
</table>

#Nodes = 5, #Edges =10

3.1.1.3 COMMUNITY DETECTION

Many science networks, including social, computer, metabolic and regulatory networks, tend to be divided naturally into communities or groups (Amelio & Pizzuti, 2013; Natarajan, Sen, & Chaoji, 2013; Newman, 2006; Ovelgonne, 2013; Takaffectoli, Rabban, Za, #239, & ane, 2013; Yang, Comar, & Xu, 2013). The pattern of communities (or friends of friends) can assist in differentiating users’ behaviours and identifying anomalous users. Very large and very small communities can be considered to be anomalies as the experiments show. A very large community shows a clique or near clique topology in the 2-
level neighbourhood, while a very small community shows a star or near star topology. In a very large community, many users follow the clique pattern and that is why they can be included in the same and large community. However, the clique phenomenon happens rarely by a minority of users.

For community detection we examine users’ super-egonets, which can give us reasonable information to find if there are any similarity and common interests between their friends by examining their connections.

Definition 6 (External Degree or Out Degree). The external degree of egonet $j_m$ to egonet $j_n$ is defined as:

$$d_j(j_m, j_n) = |\mathcal{V}^{egonet_{j_m}} \cap \mathcal{V}^{egonet_{j_n}}| + |\mathcal{E}_1 \mathcal{E}_2 \in \mathcal{E} : \mathcal{E}_1 \in \mathcal{E}^{egonet_{j_m}}, \mathcal{E}_2 \in \mathcal{E}^{egonet_{j_n}}, j_m, j_n \in \mathcal{G}|$$  \hspace{1cm} (3.6)

where $\mathcal{V}^{egonet_{j_m}}$ is the set of nodes of egonet $j_m$, and $\mathcal{V}^{egonet_{j_n}}$ is the set of nodes of egonet $j_n$, $\mathcal{E}^{egonet_{j_m}}$ is the set of edges of egonet $j_m$, and $\mathcal{E}^{egonet_{j_n}}$ is the set of edges of egonet $j_n$. The normalised external degree is defined as follows:

$$d_j(j_m, j_n)_{\text{norm}} = \frac{d_j(j_m, j_n)}{\min(|j_m|, |j_n|)}$$  \hspace{1cm} (3.7)

Definition 7 (Community). The egonets of users $j_m$ and $j_n$ form a community if at least the half of the nodes of the smaller egonet connects to the other egonet. The formal definition of community is as follows.

$$C_j(j_m, j_n) = \begin{cases} 1, & \text{if } d_j(j_m, j_n)_{\text{norm}} \geq \frac{\min(|j_m|, |j_n|)}{2} \\ 0, & \text{otherwise} \end{cases}$$  \hspace{1cm} (3.8)
3.1.1.4 CLIQUENESS AND STARNESSE

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a graph with vertex $\mathcal{V}$. Let $\mathcal{V}_1 \subseteq \mathcal{V}$. If the subgraph induced by $\mathcal{V}_1$ is a complete graph then $\mathcal{V}_1$ is a clique in which every node is connected to every other node. The number of edges in any clique is calculated by $|\mathcal{V}_1|(|\mathcal{V}_1| - 1)/2$, where $|.|$ refers to cardinality. For instance, $\text{egonet}_i$, where $i \in \mathcal{V}$, $\text{egonet}_i \subseteq \mathcal{G}$, called clique if $\text{deg}(\text{egonet}_i) = n(n - 1)/2$, where $n = |\mathcal{V}_{\text{egonet}_i}|$ and $\text{deg}$ refers to the degree of the given graph which simply is the number of edges in the graph. To calculate the cliqueness of given graph the following equation is used: $\zeta_c^{(i)} = \text{deg}(\text{egonet}_i)/n(n - 1)/2$.

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a graph with vertex $\mathcal{V}$. Let $\mathcal{V}_1 \subseteq \mathcal{V}$, the subgraph is called a star if it is a tree consisting of one vertex adjacent to all the others. The number of edges in a star topology is calculated by $(|\mathcal{V}_1| - 1)$, where $|.|$ refers to cardinality. For instance, $\text{egonet}_i$, where $i \in \mathcal{V}$ and $\text{egonet}_i \subseteq \mathcal{G}$, is called a star topology if $\text{deg}(\text{egonet}_i) = n - 1$, where $n = |\mathcal{V}_{\text{egonet}_i}|$. To calculate the starness of a given graph the following equation is used: $\zeta_s^{(i)} = \text{deg}(\text{egonet}_i)/(n - 1)$. The value of starness $\zeta_s^{(i)}$ and cliqueness $\zeta_c^{(i)}$ for each node falls within the range of 0 and 1. The value indicates the similarity between a user’s network and the full cliques, and star topology. The higher value shows more similarity; the lower value, less.

3.3. LAYER-TWO OVERVIEW: ANOMALY DETECTION ALGORITHMS

The success or failure of an anomaly detection technique depends on applying the most suitable approach based on the type of instances in given
datasets. Layer two is about introducing the proposed algorithms; these are different in terms of the inputs and concepts they use, as shown in Table 3.

Table 3. Summary of Methods

<table>
<thead>
<tr>
<th>Type of Method</th>
<th>Methods</th>
<th>Inputs</th>
<th>Model</th>
<th>Output: Outlier Score/Fuzzy Rules</th>
</tr>
</thead>
<tbody>
<tr>
<td>Distance-based</td>
<td>N-E (OddBall)</td>
<td>N(Node), E(Edge), ABC(Average betweenness</td>
<td>Power-law Regression Model</td>
<td>Distance to Regression Model, Fuzzy rules</td>
</tr>
<tr>
<td>methods</td>
<td>E-ABC</td>
<td>centrality), Com (Community), OP (Orthogonal projection)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>E-Com</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>N-Com</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>FCM_OP_EABC</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>FCM_OP_ECOM</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>FCM_OP_VCOM</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>FCM_OP_NABC</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>FCM_OP_EV</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Distribution-based</td>
<td>EM_FUZZY_S</td>
<td>( z_s^{(i)} )</td>
<td>Statistical model, Fuzzy Inference Engine</td>
<td>Fuzzy rules</td>
</tr>
<tr>
<td>methods</td>
<td>EM_FUZZY_C</td>
<td>( z_c^{(i)} )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Clustering-based</td>
<td>EM_FCM_S</td>
<td>( z_s^{(i)} )</td>
<td>Distribution based clustering model, Fuzzy Inference Engine</td>
<td>Fuzzy rules</td>
</tr>
<tr>
<td>methods</td>
<td>EM_FCM_C</td>
<td>( z_c^{(i)} )</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
are based on distance, distribution and clustering concepts. In the first method, we deal with discrete graph metrics and their relationships similar to existing literature such as the well-known method OddBall (Akoglu, et al., 2010). This method uses the distance between the observed value and the predicted value, generated by a power-law regression model as well as orthogonal projection, to spot anomalies. This method has less computational complexity; as well, it is expected to improve the accuracy of the OddBall method (Akoglu, et al., 2010). However the fuzziness of behaviours in online social networks is not considered as a decision factor in this method.

The second and third methods deal with continuous graph metrics such as the starness and cliqueness measures. They therefore need a different approach in order to detect anomalies more accurately. The objective of these methods is to improve the accuracy of the first method by introducing new metrics and by considering the fuzziness of behaviours. In this approach, we use the Gaussian mixture model and fuzzy logic to differentiate between anomalous and normal instances. In the third method we concentrate on using data mining techniques, such as clustering, to locate anomalies. In this approach we utilise fuzzy clustering to locate anomalies and fuzzy logic to define the boundaries of anomalies.

3.3.1. **LAYER-TWO (A): DISTANCE-BASED ANOMALY DETECTION USING GRAPH METRICS**

This layer involves finding anomalies based on distance from the regression model applied on graph metrics including the number of nodes and edges, average betweenness centrality, and community cohesiveness. To do that, new graph
metrics, a power-law regression model, distance (residual) to the model, the orthogonal projection of points on the model are used to model and find anomalies. Residual or distance to the model shows how far a point sits from the model. Based on this distance an anomaly score is computed to find anomalies. The details of this method are explained in Chapter 5.

3.3.2. LAYER-TWO (B): DISTRIBUTION-BASED (STATISTICAL-BASED) ANOMALY DETECTION USING GRAPH METRICS

In the probability distribution-based (statistical clustering) approach we use a combination of the Gaussian mixture model and fuzzy logic as a novel method to differentiate between normal and anomalous instances. The focus of this method is analysing the distribution of instances within natural components generated by a EM-Gaussian mixture model algorithm (Awwad Shiekh Hasan & Gan, 2009). Combining fuzzy logic and the Gaussian mixture model makes this approach suitable for dealing with the fuzzy characteristics of instances such as the friendship relationships found in social networks. EM is an unsupervised learning algorithm that is employed for data clustering based on unobserved latent variables and underlying relationships between instances. It is an iterative algorithm with two main steps, E and M. In the E-step, it tries to “guess” the values of latent variables; in the M-step, it updates the parameters of the model, based on the guesses, to maximise the likelihood of being in a component. We assume there is a joined distribution between each instance and the latent variable (e.g. its component number) that places them into the same component. The
generated components can be represented by the fuzzy membership function. We use the EM algorithm to cluster the users according to their anomaly scores.

One underlying power of fuzzy logic theory is its capability to use linguistic and quantitative variables to symbolise uncertainty, such as the friendship relation in online social networks. For instance, how many friends should a user have or which pattern of a friendship topology is considered anomalous? It is not accurate to use two levels of logic such as binary to describe this fact. In order for being anomalous we should have ten friends with minimum connections among them. The question is, if someone has eleven friends, can we consider it as normal or not? In reality characteristics such as “friendship patterns and number” are a matter of degree and are relative. They can have overlap with the other sets in contrast, to that of binary sets. The detail of this method is explained in Chapter 5.

3.3.3. LAYER-TWO (C): CLUSTERING-BASED APPROACH USING GRAPH METRICS

In the clustering-based approach we developed a Fuzzy-based clustering method in which an object is called an anomaly if it matches the clusters that are dominated by anomalies as members. This proposed fuzzy-based clustering method allows instances to be members of more than one cluster with different levels of certainty. This is important when dealing with fuzzy variables such as the number of users who can make a community. This uncertainty can affect the accuracy of detecting anomalies within the online social network data graph if we use hard-partitioning algorithms such as k-means (Hartigan & Wong, 1979). The details of this method are explained in Chapter 5.
3.4. LAYER-THREE: EVALUATION

A central question in anomaly detection is to how the effectiveness of an outlier detection algorithm can be evaluated. Most of the time it is not an easy task due to rareness of anomalies and lack of the ground-truth that informs which data points are really anomalies. Among algorithms used for anomaly detection, the unsupervised ones suffer more from this unavailability of ground-truth. This leads to not having an effective and rigorous approach that can be used to evaluate developed algorithms. That is why most researchers use case studies or different scenarios to offer an intuitive and qualitative evaluation for unsupervised algorithms (Aggarwal, 2013). In some scenarios, datasets with imbalanced class distributions are used for finding the ground-truth. In these kinds of scenarios, a minority is modelled and used as the ground-truth of anomalies (Kumar & Sheshadri, 2012). In some other scenarios, human experts can make the ground-truth for a subset of the dataset, which can then be used for training a model, with the remaining being used for evaluation.

3.4.1. DATASETS

The three real-life datasets Facebook, Flikr, and Orkut used in the evaluation of the proposed approaches are shown in Table 4; their statistical information is shown in Table 5. These datasets have been previously studied in literature (Cha, et al., 2008; Mislove, et al., 2007; Viswanath, et al., 2009). Each dataset represents relationships that exist in the social network between nodes in the form of a list of adjacencies. These datasets, collected using crawling techniques (Cha, et al., 2008; Mislove, et al., 2007; Viswanath, et al., 2009), show
different characteristics that have been previously studied and generated by other
researchers for measuring and analysing online social networks (Mislove, et al.,
2007). Each dataset includes relationships between users in the form of a list of
adjacencies. We randomly sampled 20,000 users from each network and built
their egonets to apply and evaluate the effectiveness of the proposed methods.

Table 4. Dataset Details

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Vertices</th>
<th>Edges</th>
<th>Structure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Orkut</td>
<td>Online Social networking</td>
<td>3M</td>
<td>23M</td>
<td>Undirected</td>
</tr>
<tr>
<td>Flickr</td>
<td>Online Social networking and</td>
<td>1.8M</td>
<td>22M</td>
<td>Undirected</td>
</tr>
<tr>
<td></td>
<td>online photo sharing</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Facebook</td>
<td>Online Social networking</td>
<td>64K</td>
<td>1.5M</td>
<td>Undirected</td>
</tr>
</tbody>
</table>

As shown in Table 5, these three datasets show slightly different
characteristics, in terms of variation of average, minimum, and maximum number
of nodes and edges. These differences are related to (1) the nature or the purpose
of each social network; (2) how big the collected data is; and (3) the period and
the method of crawling. For instance, Facebook’s nature is to connect people with
friends, colleagues, or the others who have lived around them. People use
Facebook to learn about friends or people they meet, upload a number of photos,
and post links. Flickr, an image and video hosting website, is being used by those
who want to share personal photographs. The main purpose of Flickr is to share
the photos. Orkut is a social networking website designed to find and meet old
and new friends and to keep up with existing connections. The characteristics of
each social network are reflected by the numbers showing in Table 5. The datasets show no self-loops topology in all local networks.

### Table 5. Statistical Information of Generated Egonets

<table>
<thead>
<tr>
<th>Metrics / Datasets</th>
<th>Facebook</th>
<th>Flickr</th>
<th>Orkut</th>
</tr>
</thead>
<tbody>
<tr>
<td>#Nodes (Min)</td>
<td>3</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>#Nodes (Avg)</td>
<td>45</td>
<td>241</td>
<td>130</td>
</tr>
<tr>
<td>#Node (Max)</td>
<td>1016</td>
<td>26186</td>
<td>20252</td>
</tr>
<tr>
<td>#Edges (Min)</td>
<td>3</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>#Edges (Avg)</td>
<td>629</td>
<td>8450</td>
<td>1590</td>
</tr>
<tr>
<td>#Edges (Max)</td>
<td>36626</td>
<td>201094</td>
<td>264283</td>
</tr>
<tr>
<td>Avg-BC (Min)</td>
<td>0.07</td>
<td>2.1</td>
<td>2.0</td>
</tr>
<tr>
<td>Avg-BC (Avg)</td>
<td>18</td>
<td>27</td>
<td>22</td>
</tr>
<tr>
<td>Avg-BC (Max)</td>
<td>498</td>
<td>735</td>
<td>440</td>
</tr>
<tr>
<td>Self-loops</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Local Graph Density (Min)</td>
<td>0.001821494</td>
<td>0.000103858</td>
<td>0.000213858</td>
</tr>
<tr>
<td>Local Graph Density (Max)</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Local Graph Density (Avg)</td>
<td>0.14459187</td>
<td>0.137425255</td>
<td>0.153634191</td>
</tr>
<tr>
<td>Local Clustering Coefficient (Max)</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Local Clustering Coefficient (Min)</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Local Clustering Coefficient (Avg)</td>
<td>0.419051746</td>
<td>0.56475023</td>
<td>0.291374258</td>
</tr>
</tbody>
</table>
3.4.2. LABELLED DATASET

Measuring the effectiveness and efficiency of anomaly detection algorithms is hampered by the lack of labelled datasets (Chandola, et al., 2009). It is much more difficult to find labelled datasets for online social networks due to privacy issues with user’s data. Online social networking sites like Facebook encourage users to provide real personal information in their profile. With these pieces of information individuals can be easily identified; and this information can also be used by domain experts to label users. As a result the online social networking sites are very sensitive to disclosure of any information, such as interacted messages and profile information, which can lead to the identification of users. Therefore, finding labelled datasets which include the required information for labelling and modelling anomalous behaviours is almost impossible. To overcome this problem, we labelled a subset of the datasets using expert inspection. The experts visually examined the egonets according to the common rules stated in Section 3.3.2.1 and labelled the nodes accordingly as either an anomaly or normal. However, star network, which is a type of anomaly, could be related to a celebrity network leading to increase false positive. To overcome this recognised problem hence, the different measures to be presented in Section 3.4.3.

It is common that a human expert may intervene in the detection process to improve the accuracy and effectiveness of anomaly detection algorithms (Aggarwal, 2013). Human expert intervention can be a supplement to the insufficient information given to the algorithms. In the other method, the experts provide and inject labelled instances to an unlabelled dataset in order to evaluate
the proposed algorithms based on the injected instances. In both settings, they provide a subset of labelled datasets for training the model.

The challenges can include a lack of labelled data and the capacity of unsupervised or semi-supervised algorithms to differentiate between noise and anomalies. This human expert intervention can augment inadequate input information given to the anomaly detection algorithms. We can use unsupervised or semi-supervised anomaly detection algorithms to narrow down the number of instances a human expert needs to examine. This narrows down subset of suspicious instances may need human intervention to be labelled appropriately. In the other cases, human experts can provide labelled instances, injecting them into an unlabelled dataset in order to evaluate outputs of the proposed algorithms by looking for injected instances if they are selected. In the third scenario human experts can provide a subset of labelled datasets for use as a training data for the detection model. To mitigate labelling bias we use different views on the same dataset to get an overall score of anomalies for a specific node. Moreover, in the labelling process we use the patterns that have already been established by the previous researcher.

Detecting anomalies in such a large amount of data is considered to be a hard job which needs to be automated as much as possible using machine learning algorithms. However, achieving acceptable detection accuracy needs a significant amount of labelled data for training the algorithms. Most of the time this data labelling needs to be done manually (Ghasemi, Rabiee, Fadaee, Manzuri, & Rohban, 2011).
One of the methods to make labelled data is active learning which is shown in Figure 13; it uses human experts to label data. Since the existing datasets were not labelled, we used visual expert inspection to label anomalies. In particular, the egonet of nodes is visually examined to decide whether the node was anomalous or not based on evidence from previous works. The graph of each egonet is visualised by an expert using the tools such as NodeXL and R and labelled accordingly based on the ground-truth defined by previous research. The information regarding the ground-truth, represented in Section 3.2.2.1, is based on this fact that the majority of users follow the “friends of friends are often friends” pattern and very few users follow either the “cliques or near-cliques” pattern (all the neighbours connected) or the “stars or near-star” pattern (mostly
disconnected). Our random sample of Facebook shows twelve percent stars and cliques topologies; for Flicker, nine percent; and for Orkut, four percent.

3.4.3. EVALUATION MEASURES

Given the ground-truth, most anomaly detection algorithms use an outlier score and threshold on this score to spot anomalies and evaluate the effectiveness of proposed algorithms. Depending on the chosen threshold level, the number of false negatives and positives can increase and decrease. Choosing a restrictive threshold can increase the false negatives; choosing a loose threshold can increase the false positives. To find a good trade-off to measure the effectiveness of an algorithm, the commonly used metrics of precision, recall, and F-Score are employed. F-Score, the accuracy measure where its highest value (1) shows the perfect score and (0) shows the worst F-score, for any given threshold \( t \) on the anomalies score, is defined as follows:

\[
F_{\text{Score}}(t) = \left( 2 \times \frac{\text{Recall}}{\text{Precision} + \text{Recall}} \right) \times 100
\]

where:

\[
\text{Precision}(t) = \frac{TP(t)}{(TP(t) + FP(t))} \times 100
\]

\[
\text{Recall}(t) = \frac{TP(t)}{(TP(t) + FN(t))} \times 100
\]

\[
TP(t) = \text{accurately detecting an anomaly given threshold } t
\]

\[
FP(t) = \text{detecting an anomaly where none exists given threshold } t
\]

\[
TN(t) = \text{no anomaly exists and none is detected given threshold } t
\]

\[
FN(t) = \text{an anomaly exists but is not detected given threshold } t
\]

The precision is expressed as the percentage of detected anomalies that correctly turn out to be anomalies. The recall is defined as the percentage of true anomalies that have been detected as anomalies at threshold \( t \). The value of precision and recall for more effective algorithms show that high values of precision may often correspond to low values of recall and vice versa (Aggarwal,
The trade-off between these values also shows a non-monotonic relationship because finding a new anomaly by the proposed algorithm can cause a spike in the precision.

To evaluate the accuracy of the proposed anomaly detection algorithms we used the case study approaches explained above (Shanbhag & Wolf, 2009) in which we can have a sample of normal and anomalous instances to evaluate the algorithms against the F-Score (True Positive Rate and False Positive Rate). To evaluate the proposed anomaly detection methods, a number of evaluation metrics, such as distance-based and clustering-based, are used in this thesis. In each experiment, all test data contains multiple types of anomalous and normal instances similar to real life. For the sake of simplicity of evaluation, we use the small subset of instances at this time.

3.1.1.5 COEFFICIENT OF DETERMINATION

Coefficient of determination metrics focus on finding the best power-law distribution model which can describe the given datasets. Power-law distributions can describe many scale-free phenomena, such as online social networks connections, well. A scale-free network is a network whose degree distribution follows a power-law. A power-law regression model based on the nature of online social networks is used to characterise the relationship between the graph metrics. This model is used to explain topological behaviour of a social network. By applying a power-law regression model on online social networks data graph, we expect the normal topological behaviour to be close to the regression model and
the anomalous ones to be far from it. A logarithmic transformation prior to computing the correlation and regression model is used to smooth the result.

In order to find the best model, a coefficient of determination ($R^2$) of the linear regression model is used to find the best fit between metrics. The coefficient of determination is computed as a goodness of fit measure for a model.

$$R^2 = 1 - \frac{SS_{\text{residual}}}{SS_{\text{total}}} \quad SS_{\text{residual}} = \sum_{i=1}^{k} (\mathcal{F}_i - \mathcal{F}_i^p)^2, \quad \text{where} \quad \mathcal{F}_i^p \text{ is the predicted value of } \mathcal{F}_i \quad \text{and} \quad SS_{\text{total}} = \sum_{j=1}^{k} \left( \mathcal{F}_j - \mathbb{E}(\mathcal{F}_j) \right)^2, \quad \text{where} \quad \mathbb{E}(\cdot) \text{ gives the expected value.}$$

$R^2$ confirms which regression model is the best choice for modelling the relationships between graph metrics. The relationships between graph metrics, such as average betweenness centrality and number of nodes, are modelled using different techniques including power-law regression, distribution and clustering.

### 3.4.4. BENCHMARK BASED ON STRUCTURAL BEHAVIOURS IN OSNS

This section details the benchmarks used for evaluating the proposed methods. The purpose of evaluating against these benchmarks is to understand the strengths and drawbacks of the proposed and benchmark methods applied to the same datasets. We used different state-of-the-art methods as benchmarks to prove the effectiveness of the proposed approaches. These methods are chosen based on their similarity to our core concepts. The experiments were evaluated against the methods that use techniques similar to the proposed methods. The benchmark methods include OddBall (Akoglu, et al., 2010), k-means (Hartigan & Wong, 1979), fuzzy c-means (Gath & Geva, 1989), and expectation-maximization
(Moon, 1996). All these methods apply to the structural data graph in the same way as the proposed methods. The OddBall method discovers and uses the power-law rules which are followed by the majority of egonets. The generated rules are based on the extracted features of all egonets. These rules are used to detect anomalies. The fuzzy c-means clustering algorithm is an unsupervised algorithm for grouping data points into a chosen number of clusters with fuzzy boundaries defined by a fuzziness index (Ling, Zhi-Chun, & Dong-Mei, 2008). The expectation-maximization algorithm, which is used for finding the maximum likelihood or maximum a posteriori estimates of parameters in statistical models, is an iterative algorithm with two main steps. In the E-step, it “guesses” the values of an unobserved latent variable. In the M-step, it updates the parameters of the model (Fortunato, Latora, & Marchiori, 2004). These methods are used for benchmarking as they align with the concepts that are used by the proposed approaches.

3.4.5. FIND THRESHOLD

Using the anomaly score developed in each method and a set of labelled datasets as inputs, we start from the minimum value of the anomaly score and iteratively compute the number of false negatives and the false positives rate. The threshold algorithm starts from an initial value, which can be the minimum anomaly score among the sample data, and iteratively updates the threshold. It stops once we have no improvement in maximising the objective function, which is equation 3.10.

\[
\text{argmax } L(\tau, \phi), \text{ where } \tau \text{ is threshold and } \phi \text{ is F-Score} \quad (3.10)
\]
This approach is applied for all methods (proposed and benchmarks) separately. The steps for computing the threshold are shown in Figure 14.

<table>
<thead>
<tr>
<th>Algorithm Threshold Finding (ATF)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Input:</strong> aScore (Anomaly score)</td>
</tr>
<tr>
<td><strong>Output:</strong> The best F-Score</td>
</tr>
</tbody>
</table>

```
Begin
1. TrShld [ ] = [Min(aScore) Max(aScore)]
2. FSTmp = 0
3. F-Score = 0
4. iterations = 0, Max_Iter = sizeOf (TrShld [ ])
5. While (iterations < Max_Iter)
6. Use TrShld [iterations] to compute:
7. TP, FP, FN // TP=True Positive, FP = False Positive FN = False Negative
8. Prs = TP / (TP + FP)
9. Rec = TP / (TP + FN)
10. F-Score = 2 * Prs * Rec / (Prs + Rec)
11. FSTmp = Max(FSTmp, F-Score)
12. iterations++
13. End
14. Return FSTmp
End
```

Figure 14. The Threshold Finding Algorithm

3.5. SUMMARY

This chapter introduced the multi-layered framework in which our methods developed. This framework provides the full requirements needed for graph modelling, features selection and extraction, developing, and evaluation of the proposed approaches. The first layer of the framework was concerned with pre-processing data including normalisation, cleaning, transformation, integration, reduction, feature extraction and selection, modelling OSNs using graph theory, and computing graph metrics. The extracted features, one of our contributions in this research, will be used as inputs for the proposed methods. The details of each
method are explained in Chapter 4. The second layer of the framework briefly introduced the proposed methods, which differ based on their input data and concepts, including distance, distribution, and clustering. In the last layer of the framework, our emphasis was on describing the process of labelling the datasets, applying the presented methods as well as the benchmark methods on the Facebook, Orkut, and Flickr datasets, generating the anomaly score, and finding the best threshold to maximise the F-score. Datasets details from real-life online social networks were also detailed.
Chapter 4: Anomaly Detection Methods

This chapter concentrates on introducing the proposed approaches in detail. These approaches are based on distance, distribution and clustering concepts. For the distance-based approach we compute the regression model, the distance to model, and the orthogonal projection of points on the model to find anomalies. In the distribution-based approach we use the Gaussian mixture model, and fuzzy logic to differentiate between anomalous and normal instances. In the clustering-based approach we employ a fuzzy clustering to locate anomalies. The proposed method centred on the application of fuzzy logic as a post-processing of the results obtained from the use of clustering algorithms such as EM, and FCM. The key contributions are extracting features such as the orthogonal projection, average betweenness centrality, starness and cliqueness, and the inference system that computes the aScore.

4.1 INTRODUCTION

The proposed methods for finding anomalies can be divided into three groups based on distance, distribution, and clustering models according to their application and techniques. Different types of data input need different techniques in order to handle them. The proposed distance-based method, working on discrete instances, calls an object an anomaly if it is seated a long way away from the generated model. The proposed distribution-based method, working on
continuous instances, uses statistical models to call an object abnormal if it falls outside of the model. The proposed clustering-based method, working on discrete instances, calls an object an anomaly if it fits the object into a cluster dominated by anomalies. A distribution-based approach depends on the local density of the neighbourhood and uses a local outlier factor to detect outliers. This type of algorithm uses a number of objects and density. The central advantage of using these three well-known concepts in the anomaly detection area is to analyse and find the most suitable approach for characterising online social networks behaviours.

For the distance-based approach, a power-law regression model, the distance (residual) to the model, and the orthogonal projection of points on the model are computed to model and find anomalies. Distance to the model shows how far a point sits from the model. Based on this distance an anomaly score is computed to find anomalies. In the distribution-based approach, we use the Gaussian mixture model and fuzzy logic to differentiate between anomalous and normal instances. In the clustering-based approach, we employ fuzzy clustering to locate anomalies. We use fuzzy logic to define the boundaries of anomalies as they can be treated as a multiple-valued logic problem.

Of the outliers defined by Akoglu, et al. (2010)- star or near-star and clique or near-clique, heavy vicinity, and dominant edge- in this research we concentrate on types of anomaly that are more applicable to the existing online social networks topology. In online social networks, especially the ones modelled by undirected graph (e.g. Facebook); users can only have a connection with each user that shows their friendship. Therefore heavy vicinity and dominant edge outliers
cannot be applicable to this kind of online social networks data. For instance, heavy vicinity is used in a who-calls-whom network in which the number of phone calls used as a weight for each edge. The extreme total weight for a given number of contacts would consider as anomaly. For dominant heavy links, in the who-calls-whom scenario a very heavy single edge in the egonet of a user would be considered as an anomaly, possibly a stalker who keeps on calling only one of the user’s friends an extreme count of times. The “stars or near-star” pattern shown in Figure 15 (a) is a type of topology in which nodes are mostly disconnected. The “cliques or near-cliques” pattern shown in Figure 15(b) is a type of topology in which nodes are mostly connected.

![Figure 15. Stars and Cliques](image)

a) Star  
b) Clique
4.2 DISTANCE-BASED APPROACH USING GRAPH METRICS AND ORTHOGONAL PROJECTION

As shown in Figure 16 the steps of the distance-based method is detailed. This method identifies anomalies by differentiating people’s online behaviour using various graph properties (metrics) and a power-law regression model as well as orthogonal projection of objects. The method models user relationships with combinations of different graph metrics extracted from the user egonet. Anomalies can be identified based on the distance from the regression model and the cluster of orthogonal projection of objects on the regression model. The objects are considered anomalous if they fit in a small neighbourhood dominated by anomalous objects or sit far from the regression model.
4.2.1 METHOD OVERVIEW

This method aims to find the common behaviour followed by the majority of nodes. It computes graph metrics of a user’s egonet and then finds a regression model which can be the best predictor of anomaly behaviours. The model uses the relationships between the graph properties to find common patterns to distinguish users that may be anomalous. The method consists of the following steps, explained in the previous sections and shown in Figure 17.

**Algorithm Distance-Based**

**Input:** $G = (\mathcal{V}, \mathcal{E}), \varepsilon(Threshold), Max\_Iter$ (Maximum iteration), $c(\#clusters)$

**Output:** anomalous nodes, $\mathcal{U} = \{u_1, u_2, \ldots, u_t\}$, where $\mathcal{U} \subset \mathcal{V}, t << |\mathcal{V}|$

**Begin**

1. **Initialize**
2. $m = |\mathcal{V}|$
3. **For each** node $j \in \mathcal{G}$
   4. Identify egonet $\mathcal{J}^e_j = \{j, j_1, j_2, \ldots, j_n\} \in \mathcal{V}^e_{\text{egonet}_j}$
   5. Compute the graph metrics: $(A_j = \sigma^{\text{Avg}_{BC}}, |V_j| = |\mathcal{V}^e_{\text{egonet}_j}|, E_j = |\mathcal{E}^e_{\text{egonet}_j}|, C_j)$
   6. Find the coordinate of points $(x^{(j)}, y^{(j)})$ on the X-Y plane where X and Y become the graph metrics under consideration
7. **End**
8. Compute power-law regression $f(x) = \alpha x^\theta$ between graph metrics:
   9. $f(\mathcal{V}) = \alpha_1 A^\theta$, where $\mathcal{V}$ is a set of nodes and $A$ is a set of Avg-BC
   10. $f(\mathcal{E}) = \alpha_2 E^\theta$, where $E$ is a set of edges and $\mathcal{V}$ is a set of nodes
   11. $f(\mathcal{E}) = \alpha_3 A^\theta$, where $E$ is a set of edges and $A$ is a set of Avg-BC
   12. $f(\mathcal{V}) = \alpha_4 C^\theta$, where $\mathcal{V}$ is a set of nodes and $C$ is a set of Com
   13. $f(E) = \alpha_5 C^\theta$, where $E$ is a set of edges and a set of $C$ is Com
14. **For each** node $j \in \mathcal{G}$ wrt. $f(\Omega)$, where $\Omega = \{\mathcal{V}, \mathcal{E}\}$
   15. $x_j(\rho_x^{(j)}, \rho_y^{(j)}) = \arg\min_{\mathcal{X}} D^{(j)} = \sqrt{(x - x^{(j)})^2 + (\alpha x^\theta - y^{(j)})^2}$
   16. $\mathcal{X} = \{x_1(\rho_x^{(1)}, \rho_y^{(1)}), x_2(\rho_x^{(2)}, \rho_y^{(2)}), \ldots, x_m(\rho_x^{(m)}, \rho_y^{(m)})\}$
   17. MEM = FCM ($\varepsilon, Max\_Iter, c, m, \mathcal{X}$) \cite{(Figure 28)}
18. **End**
19. **End**
20. **If** $(x_j \in \text{MEM}_0)$ or $(x_j \in \text{MEM}_c)$
21. $\mathcal{U} \leftarrow u_i$
22. **End**
**End**

Figure 17. Steps to Detect Anomalies Using Distance-Based Approach
The method starts to compute the graph metrics using egonet, power-law regression model \((Y = CX^\theta)\), and orthogonal projection of instances on the model. The graph metrics include the number of nodes \((N)\), the number of Edges \((E)\), the average betweenness centrality \((ABC)\), and community cohesiveness \((Com)\). The relationship between the metrics is formulated using \(f(x) = \alpha_n x^{\theta_n}\) function which represents, the power-law regression model.

The next step of the approach is to apply the FCM algorithm to the instances obtained from the last step in order to group them into the proper clusters. The methods associated with each relationship after using orthogonal projection and FCM respectively are called FCM_OP_EABC, FCM_OP_ECOM, FCM_OP_VCOM, FCM_OP_NABC, and FCM_OP_EV. Using the result of FCM in the fuzzy IF-THEN rules-based system (fuzzy inference engine) such as Mamdani (Mamdani & Assilian, 1975), we generate rules to model anomalous behaviours. Mamdani is a fuzzy inference system that uses fuzzy set theory to map the input membership function (IMF) to output membership function (OMF). Once the rules are generated they can be used to find anomalies in online social networks with similar structure in future. The new rules might need to be generated if a given online social network graph has different structures.

### 4.2.2 INPUT-COMPUTING GRAPH METRICS

We use local graph properties to extract common rules. Local metrics refer to a single node (an ego), its 1-level neighbourhood (an egonet) and 2-level neighbourhood (a super-egonet). In particular, we propose the use of our new
extracted features, such as betweenness centrality and average betweenness centrality of a user’s egonet, and the community cohesiveness of the user’s super-egonet, as potential measures for identifying anomalies based on the structure of users’ links.

Ego, a focal node in online social networks, can be an individual, a group, organisation, or community. The N-level neighbourhood of an egonet includes a group of nodes which have a connection at a path length of N to that egonet. The N-level neighbourhood also includes all connections between neighbours. For instance, a 1-level neighbourhood is a collection of nodes that are directly adjacent to an ego and is called an egonet. An egonet has a path length of one and includes all of the connections between direct neighbours of its ego. A 2-level neighbourhood in our context is called a super-egonet which includes all the connections with a path length of two. Our analysis is based on an undirected graph and therefore we do not need to consider in or out neighbours for each ego (Hanneman, 2013). Instead we take the degree of each node into consideration.

After identifying the egonet and super-egonet of a user, the metrics computed include:

- N: number of nodes in a user’s egonet;
- E: number of edges in a user's egonet;
- ABC: the average betweenness centrality of all nodes in a user’s egonet;
- Com: the community cohesiveness of the user’s super-egonet.
The regression model used to find common patterns between the extracted features are as follows. The regression model is formed based on a relationship between the metrics introduced in this research.

- **N vs. E (N-E) (Akoglu, et al., 2010):** Compute a regression model \( E_i \propto N_i^a \), where \( 1 \leq a \leq 2 \), \( E_i \) is the number of edges, \( N_i \) is the number of nodes, and \( a \) is the power-law exponent for user \( i \)’s egonet.

- **E vs. ABC (E-ABC) and V vs. ABC:** Compute a regression model \( Y = C \chi^\theta \), where \( Y \) is \( E/V \), and \( \chi \) is ABC, and \( \theta \) is the power-law exponent for user \( i \)’s egonet.

- **N vs. Com (N-Com) and V vs. Com:** Compute a regression model \( Y = C \chi^\theta \), where \( Y \) is Com, \( \chi \) is \( N/E \), and \( \theta \) is the power-law exponent for user \( i \)’s super-egonet.

### 4.2.3 COMPUTE REGRESSION MODEL

A statistical model is computed to describe the relationship between the graph metrics calculated in the previous step as independent (e.g. E) and dependent variables (e.g. ABC). The model parameters are estimated so as to minimise the difference in the observed values and the predicted ones. The regression model uses these two values, from the dataset consisting of measurements of the graph metrics, to develop a model predicting the value of the dependent variable. The independent and dependent variables come from the relationships between N vs. E, ABC vs. E, and N vs. Com, which can be in the
form of power-law. For instance we use N, ABC, or Com as dependent variables and find their relationship with the others.

To find suitability of the power-law model we apply it on the labelled datasets. Estimated values are used in finding the best model which can make better predictions than others. The goal is to have a best mathematical model that can predict the values of a dependent variable (e.g. $y_{p,i}$) based upon an independent variable (e.g. $x_i$). A regression analysis including an independent and a dependent variable is normally plotted in two dimensions (scatter plot). Visually reviewing the data plot before applying a regression analysis allows us to get a rough idea of the relationship between variables to find the best model. To have the best fit the error sum of squares is usually used, given the fact that we have only a subset of all possible data. The model parameters have to be estimated from the subset of the data. The estimated parameters are more accurate if we have a more complete data set. To find the best fit we employ $R^2$.

In our sample datasets based on the coefficient of determination ($R^2$), we found that a power-law regression model, for instance compared to a linear model, is the best model to describe the relationships among the different graph metrics, and to explain how the graph metrics change with relation to each other. The model is formulated as follows:

\[
\text{Power-law regression model: } y_p = C \cdot x^\theta + \epsilon \tag{4.1}
\]

\[
\text{Linear regression model: } y_e = \beta_0 + \beta_1 \cdot x + \epsilon \tag{4.2}
\]

where $\theta$ is a power-law exponent, $C$ is a constant factor, $\epsilon$ is error, $\beta_1$ is a slope, $\beta_0$ is a intercept. In this formula we use different metrics as inputs such that
\( Y_X \) can be the number of node or edges and \( X \) can be the number of node or edges, the average betweenness centrality, and the community.

### 4.2.4 COMPUTING THE DISTANCE FROM REGRESSION MODEL

Scoring techniques are utilised to assign an anomaly score to each instance in the test data. This aims to show the degree of anomalousness for each instance. We expect this score to give us a ranked list of anomalies to choose. It is up to the analyser to either pick the top few anomalies or use a cut-off threshold. For each relationship, an anomaly score function based on distance from the regression model is determined.

For each regression model, we used Equations 4.5, 4.6, and 4.7 as anomaly scores (Akoglu, et al., 2010) to determine the distance from the regression model for each ego \( i \) or the user \( i \).

\[
\text{aScore}_{pl}(i) = \frac{\max(y, C X^X_i)}{\min(y, C X^X_i)} \log \left( |y_i - C X^X_i| + 1 \right)
\]

\[
\text{aScore}(i) = \text{aScore}_{pl}(i)
\]

where \( y_i \) is the y-value (e.g. number of edges), \( X_i \) is x-value (e.g. average betweenness centrality) of computed graph metrics related to egonet \( i \). This measures “distance to regression model” by penalising the number of times that \( y_i \) deviates from the line. The \( \max \) and \( \min \) functions return the largest and smallest values between the expected value for given point \( i \) and the actual value respectively. It uses log to penalise each node with the number of times that actual value deviates from the expected value.
4.2.5 COMPUTE ORTHOGONAL PROJECTION

In this step the orthogonal projection of points, representing relations between the graph metrics on the regression model, are computed. We aim to overcome the weakness of the distance-based approach in which the detecting anomalies are based on the distance of points from the regression model. They assume anomalous instances to be way off the regression model, meaning that they do not fit the pattern. However, all outliers are not anomalies; sometimes the normal points also sit far from the regression model, because of the metrics’ characteristics used to make this model as well as the average of the instances. Thus the instances that fall away from the curve with negative correlation cannot definitely be considered as anomalies. For example, if the relationship between the extracted features scatters along the regression line and can segment the line into different area, therefore, using only the distance to the regression line can increase the false alarms. For these reasons, by applying orthogonal projection we are able to overcome the problem of normal instances with negative correlation. Now to spot anomalies we can apply any density-based approaches such as clustering on projected instances.

As shown in Figure 18 and Figure 19, we use orthogonal projection of points on a power-law regression line , \( f(x) = \alpha x^\theta \), in the \( xy \)-coordinate system. \( x \) and \( f(x) \) represent the relationship between the graph metrics, and \( x^{(j)} \) and \( y^{(j)} \) are defined as the point values. Figure 18 shows the original points and the regression line that models the points. Figure 19 represents the orthogonal projection of the points on the regression line.
To find the projected points on the power-law regression line as shown in the above figures, let $D^{(j)} = \sqrt{(x - x^{(j)})^2 + (\hat{y}(x) - y^{(j)}))^2}$ denote the distance between the points $(x^{(j)}, y^{(j)})$ and the points $(x, \hat{y}(x))$ on the regression line. Intuitively, it makes sense that the shortest distance between two points turns
out to be a perpendicular line. To do that, we compute the distance between the
given point \( j \) and regression line \( \alpha x^\theta \) with the objective of minimising the
distance of the point to the line \( D^{(j)} \). Solving \( x_j(p_{x}^{(j)}, p_{y}^{(j)}) \) with respect to \( x \)
gives us the orthogonal projection of the points \( x^{(j)}, y^{(j)} \) on the regression power-

law line such that \( X = \{x_1(p_{x}^{(1)}, p_{y}^{(1)}), x_2(p_{x}^{(2)}, p_{y}^{(2)}), ..., x_m(p_{x}^{(m)}, p_{y}^{(m)})\} \).

and \( x_j(p_{x}^{(j)}, p_{y}^{(j)}) = \text{argmin}_x D^{(j)} = \text{argmin}_x \sqrt{(x - x^{(j)})^2 + (\alpha x^\theta - y^{(j)})^2} \)

Therefore to find the orthogonal projection of the points \( (x^{(j)}, y^{(j)}) \) to the
line we compute the absolute minimum of \( D^{(j)} \) by taking its derivative and
setting it to zero as follows:

\[
x_j(p_{x}^{(j)}, p_{y}^{(j)}) = 0 = D^{(j)'} = \frac{2(x - x^{(j)}) + 2(\alpha x^\theta - y^{(j)})(\alpha x^\theta - y^{(j)})'}{\sqrt{(x - x^{(j)})^2 + (y - y^{(j)})^2}} \quad (4.5)
\]

where \( \hat{f}(x) = \alpha x^\theta; \hat{f}(x)' = \theta \alpha x^{\theta - 1} \)

\[
2(x - x^{(j)}) + 2(\alpha x^\theta - y^{(j)})\theta \alpha x^{\theta - 1} = 0 \quad \text{Where} \ x^{(j)}, y^{(j)} \text{are constant.}
\]

The set \( X \) includes the orthogonal projection of given instances on the
regression model. To find the orthogonal projection of the points \( (x^{(j)}, y^{(j)}) \) to
the fitting line we compute the absolute minimum of \( D^{(j)} \) by taking its derivative
and setting it to zero. It starts by calculating the distance between two points
\((a, b)\) and \((c, d)\) using the Pythagorean Theorem which presumes that \((a, b)\) and
\((c, d)\) are not horizontally or vertically aligned in the \( xy - plane \). Therefore a
right triangle can form where the hypotenuse would be the distance between the
two points.
4.2.6 FUZZY C-MEANS (FCM) CLUSTERING

In this section we use the fuzzy c-mean (FCM) clustering algorithm as an unsupervised algorithm for grouping orthogonal points into a chosen number of clusters with fuzzy boundaries defined by the fuzziness index (Ling, et al., 2008). FCM is a data clustering technique in which a dataset is grouped into c clusters, with every data point in the dataset belonging to every cluster to a certain degree (Gath & Geva, 1989). For example, a certain data point that lies close to the centre of a cluster will have a high degree of belonging or membership to that cluster while another data point that lies far away from the centre of a cluster will have a low degree of belonging or membership to that cluster. FCM starts with an initial guess for the cluster centres, which are intended to mark the mean location of each cluster. Next, FCM assigns every data point a membership grade for each cluster, then iteratively updates the cluster centres and the membership grades for each data point. This iteration is based on minimising an objective function that represents the distance from any given data point to a cluster centre weighted by that data point's membership grade.

As shown in Eq. 4.5, FCM aims to minimise the objective function $J(U,\mu)$, where $U$ is a matrix of the degree of membership of all instances such as $i$ in the cluster $j$ and $\mu$ a set of all the cluster means. This is based on the maximum likelihood estimation, using a fuzzy membership function.

$$\text{argmin } J(U,\mu) = \sum_{j=1}^{n} \sum_{i=1}^{c} (m_{ij})^\beta d^2(x_j,\mu_i)$$

$(4.6)$

$U$ represents a matrix of the fuzzy membership $(m_{ij})_{n\times c}$;
\[
m_{ij} = \frac{1}{\sum_{k=1}^{C} \left( \frac{d^2(x_j, \mu_k)}{d^2(x_j, \mu_k)} \right)^{1/(\beta-1)}}
\]  

(4.7)

where \( m_{ij} \in [0,1] \); \( \sum_{i=1}^{C} (m_{ij}) = 1 \); \( n \) is the number of data points; \( \beta \) is the fuzziness index \( \beta \in [1,\infty] \), usually \( \beta = 2 \) is considered a good setting for most of the problems (Gath & Geva, 1989);

\[
\mu_j = \frac{\sum_{j=1}^{n} (m_{ij})^{\beta} x_j}{\sum_{j=1}^{n} (m_{ij})^{\beta}}, \quad i = 1,2,\ldots,c
\]

(4.8)

\[
\mathcal{L}(i \mid x_j) = 1/d^2(x_j, \mu_i) / \sum_{k=1}^{k} 1/d^2(x_j, \mu_k);
\]

(4.9)

where the \( \mathcal{L}(i \mid x_j) \) is the posterior probability (the probability of selecting the \( x_j \)th cluster given the \( x_j \));

\[
F_i = \frac{\sum_{j=1}^{N} \mathcal{L}(i \mid x_j)(x_j-\mu_i)^T(x_j-\mu_i)}{\sum_{j=1}^{N} \mathcal{L}(i \mid x_j)}
\]

(4.10)

\( d^2(x_j, \mu_i) \) is the distance between the \( j \)th data and \( i \)th cluster centre;

\[
d^2(x_j, \mu_i) = (x_j - \mu_i)^T F_i (x_j - \mu_i) = \| x_j - \mu_i \|^2
\]

(4.11)

where \( F_i \) is the fuzzy covariance; for Euclidian distance \( F_i \) is the identity matrix ( \( F_i = I \) ); for an exponential distance which is based on maximum likelihood estimation (Gath & Geva, 1989), \( F_i \) and distance are calculated as follows:

\[
d^2(x_j, \mu_i) = \left[ \frac{\det(F_i)}{\vartheta_i} \right]^{1/2} \exp \frac{1}{2} \left[ (x_j - \mu_i)^T F_i^{-1} (x_j - \mu_i) \right]
\]

(4.12)

\( \vartheta_i \) is probability of instance \( i \) to be clustered in cluster \( j \) computed as follows:

\[
\vartheta_i = \frac{1}{N} \sum_{j=1}^{N} \mathcal{L}(i \mid x_j);
\]

(4.13)
4.3 DISTRIBUTION-BASED (STATISTICAL-BASED) APPROACH USING GRAPH METRICS

![Diagram of the distribution-based method]

Figure 20. Layer 2 of Framework: Distribution Based method

Figure 20 shows the steps of the distribution-based method. The distribution-based approach fits a model to instances according to their distribution patterns. This method calls instances an anomaly if they deviate from the model. The model is based on probability distribution and its parameters are learned during a training process. For instance, a generative model such as the Gaussian mixture (Mahadevan, Weixin, Bhalodia, & Vasconcelos, 2010) describes the data in the form of a mixture of Gaussian components. To learn the parameters of the model, the Expectation-Maximization algorithm (Awwad Shiekh Hasan & Gan, 2009) is used. Using the probability distribution and the density-based fit approaches together with this method makes it suitable for modelling anomalies (Aggarwal, 2013) in which data instances with very low fit
can be considered as anomalies. The main advantage of distribution-based methods is that they can be applied to any type of data, including categorical.

4.3.1 METHOD OVERVIEW

In the distribution-based approach, the distribution of instances is computed by a model; outliers are then defined as observations with a low distribution. This method presents the graph theory to model the user network topology, the Gaussian mixture model (Liu, Lung, Lambadaris, & Seddigh, 2013), to model the graph data and fuzzy logic to deal with the variation and sharp boundary. The Gaussian mixture model is a parametric probability density model represented by components fitted onto the probability distribution of input instances. The best fitted parameters of the GMM matching the given data should be computed. These parameters can be obtained by maximizing the likelihood estimation using the expectation-maximization (EM) (Dempster, Laird, & Rubin, 1977) algorithm. The purity measure of the clusters is then used to find which cluster is predominantly anomalous and which ones are not. Finally, we employ fuzzy logic (Zadeh, 1965) to label the clusters and define the degree of abnormality by using membership functions and a linguistic variable.

We use fuzzy logic to deal with anomalies as they can be treated as a multiple-valued logic problem. The degree of abnormality is determined by the lower and upper bound of the membership functions which now identifies the label for each cluster. The use of fuzzy logic (Zadeh, 1965) allows us to handle the quantitative features of online social networks, such as relationship topology, which can be considered as a fuzzy variable. For instance, how many friends
should a user have to be considered a social or influential person (Gupta, Sycara, et al., 2013; Subbian, et al., 2013; Zhang, Zhu, et al., 2013)? Or how much topology of a user network should be similar to a star or clique topology before being considered to be an anomaly? It is not accurate to use two-level logics such as binary to describe these kinds of characteristics. In reality the characteristics such as friendship, starness, cliqueness, and community can be best represented by degree as they are relative concepts. The methods for computing starness and cliqueness are called EM_FUZZY_S, and EM_FUZZY_C respectively.

Algorithm Distribution-based
Input: $\mathcal{G} = (\mathcal{V}, \mathcal{E})$
Output: anomalous nodes, $\mathcal{U} = \{u_1, u_2, \ldots, u_t\}$ where $\mathcal{U} \subset \mathcal{V}$, $t << |\mathcal{V}|$

Begin
1. Initialize $\mathcal{U} = \{\}$, $B=0$
2. For each node $i \in \mathcal{G}$
3. 
4. End
5. For each egonet$_i$
6. $D \leftarrow \text{deg}(\text{egonet}_i)$
7. $n \leftarrow |\text{egonet}_i|$
8. $B \leftarrow \text{max}(B,D/(n-1))$
9. End
10. For each egonet$_i$
11. $D \leftarrow \text{deg}(\text{egonet}_i)$
12. $n \leftarrow |\text{egonet}_i|$
13. $\zeta_c^{(i)} \leftarrow D/((n(n-1)/2))$
14. $Z_s^{(i)} \leftarrow D/((n-1) \times B)$
15. End
16. $\mu_x, \sigma_x, \alpha_x, \beta_x, \gamma_x, K_{c/s} \leftarrow \text{Expectation–Maximization (}\zeta_c \text{ or } \zeta_s)$
17. $M_{x-K_{c/s}} \leftarrow \text{Fuzzy-Membership-Function (} \mu_x, \sigma_x, \alpha_x, \beta_x, \gamma_x )$
18. If $(\tilde{r}_s^{(i)} == \ell \sigma w) \in M_0$ or $(\tilde{r}_c^{(i)} == \ell \mu g H) \in M_{K_{c/s}}$
19. $\mathcal{U} \leftarrow u_i$
End

Figure 21. Steps Required in Distribution-Base Anomalies Detection
Figure 21 provides an overview of the proposed algorithm. The Time complexity of the steps is proportional to the degree of nodes $O(N)$ as each node is visited once for analysis, independent of the other nodes. The complexity of EM is $O(N \log N)$ where $N = |\mathcal{V}|$ is the number nodes in the graph.

### 4.3.2 INPUT–LOCAL GRAPH PROPERTIES

The graph theory (Newman, et al., 2002) is used to model metadata such as links that users have established with other users in the network (friendship relationship). Nodes represent people and edges represent the links that connect nodes/people using a range of relationships such as friendship, affiliation, family and many others. The input to our approach is an online social network where users and their relationships are modelled with a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. The graph $\mathcal{G}$ consists of a set $\mathcal{V}$ of vertices (nodes) and set $\mathcal{E}$ of edges where each $e \in \mathcal{E}$ is an unordered pair of distinct vertices. As described in Chapter 3, we compute a preliminary anomaly score (starness, and cliqueness) using the user’s egonet properties such as the numbers of nodes and edges. These preliminary scores represent each user in the network for the purpose of this analysis.

### 4.3.3 CLUSTERING PRELIMINARY ANOMALY SCORE WITH UNSUPERVISED LEARNING

We have an unlabelled dataset of size $m$, $\zeta = \{ \zeta^{(1)}, ..., \zeta^{(m)} \}$; where $\zeta^{(i)} = \zeta_{c}^{(i)} \mid \zeta_{s}^{(i)}$, $\zeta_{c} = \{ \zeta_{c}^{(1)}, ..., \zeta_{c}^{(m)} \}$, and $\zeta_{s} = \{ \zeta_{s}^{(1)}, ..., \zeta_{s}^{(m)} \}$. These parameters, called starness and cliqueness of each ego, are computed based on the
formula represented in Section 3.2.2. The members of $\zeta$ are distributed with $\mathcal{P}(\zeta \mid \phi)$. The $\zeta^{(i)}$ distribution is controlled by $\phi$, which is a vector of unknown parameters. The density of the sample dataset set can be defined as:

$$
\mathcal{P}(\zeta \mid \phi) = \prod_{i=1}^{m} \mathcal{P}(\zeta^{(i)} \mid \phi) = \mathcal{L}(\phi \mid \zeta) \tag{4.14}
$$

This represents a maximum likelihood estimation problem in which we need to maximise the $\mathcal{L}$, $\hat{\phi}$. Usually maximising $\log \mathcal{L}(\phi \mid \zeta)$ instead of $\mathcal{L}(\phi \mid \zeta)$ is analytically easier; therefore, it is determined such as:

$$
\ell(\phi) = \log(\mathcal{L}(\phi \mid \zeta^{(i)})) \tag{4.15}
$$

To solve this problem we assume there is a latent variable, $\mathcal{Y}^{(i)}$, which is a distributed multinomial with parameter $\phi$ and has a joint distribution with $\zeta^{(i)}$. Formally, $\mathcal{Y}^{(i)} \sim \text{Multinomial}(\phi)$, where $\phi = \{\phi_0, \ldots, \phi_c\}$, $\phi_c \geq 0$ and $\sum_c \phi_c = 1$, $c$ is an index to the Gaussians modelling $\zeta^{(i)}$.

$$
\mathcal{P}(\zeta^{(i)}, \mathcal{Y}^{(i)}) = \mathcal{P}(\zeta^{(i)} \mid \mathcal{Y}^{(i)}) \mathcal{P}(\mathcal{Y}^{(i)}) \tag{4.16}
$$

This problem can be considered as a maximum likelihood estimation problem such as that shown in Figure 22 and Figure 23. The parameters $(\phi, \mu, \Sigma)$ shown in Equation 4.18 to 4.21 can be solved by utilising a maximum likelihood estimation algorithm, such as expectation-maximization (EM) (Dempster, et al., 1977). EM is an iterative algorithm with two main steps E and M. In the E-step, it tries to “guess” the values of $\mathcal{Y}^{(i)}$. In the M-step, it updates the parameters of the model based on the guesses (Ng, 2012).

E-step:

$$
\hat{\phi} = \arg \max_{\phi} \ell(\phi) \tag{4.17}
$$
M-step:

\[
\ell (\phi, \mu, \Sigma) = \sum_{i=1}^{m} \log \mathcal{P} (\xi^{(i)}; \phi, \mu, \Sigma) \\
= \sum_{i=1}^{m} \log \mathcal{P} (\xi^{(i)}|y^{(i)}; \mu, \Sigma) + \log \mathcal{P} 
\] (4.18)

\[
\phi_d = \frac{1}{m} \sum_{i=1}^{m} \mathbf{1}\{y^{(i)} = d\}; \\
\mu_d = \mathbb{E}[\xi^{(i)}] \mathbf{1}\{y^{(i)} = d\} \\
\Sigma_d = \mathbb{E}[(\xi^{(i)} - \mu_d)(\xi^{(i)} - \mu_d)^T] \mathbf{1}\{y^{(i)} = d\} 
\] (4.19), (4.20), (4.21)

where \(1\{True\} = 1, \ 1\{False\} = 0\)

Figure 22. Observed Data Points
Application of the EM algorithm on each anomaly score set \((\zeta^{(i)}, \zeta^{(j)})\) will result in the set of different components Gaussian mixture model. We use log likelihood to determine the number of components that defines the best fit model. As shown in Figure 24, with the three datasets used in this research, there is an increase in the log likelihood of generated components with the increase in the number of components until it reaches six. There is a very minor increase in log likelihood when the number goes from six to seven. However, the change is not big enough to be considered for adding more components. This step produced the components which accommodate instances based on their underlying relationships. However the components holding the anomalies need to be represented using the fuzzy inference engine. The advantages of using this probabilistic model along with fuzzy logic can include accuracy improvement, a soft boundary between normal and anomalous instances, a one-time rule.
generation process, the separation of the model and representation of its components, and unsupervised learning. It is very important for real-life applications that probabilistic modelling does not require classified data for training and that it can work in an unsupervised way. Moreover representing the generated component using the fuzzy membership function can make the results more interpretable compared to using only the model itself.

Figure 24. Number of Component vs Log Likelihood for Three Datasets

4.3.4 CLASSIFICATION USING FUZZY INFERENCE ENGINE

In the previous step, the natural regions and boundaries of the anomaly score are determined by the components. We propose to use fuzzy logic to label each region and discover the degree of abnormality of a user, based on the lower and upper bound of anomalousness. An observation would be called an anomaly
or normal if it falls into a certain interval. The intervals are represented by membership functions (MFs). There are different shapes of membership functions such as triangular, trapezoidal, piecewise linear, Gaussian, or singleton.

A fuzzy inference system (FIS) transforms the crisp inputs (distinct inputs) into linguistic variable inputs and outputs using membership functions (MFs). Fuzzy logic refers to domain interval as a range in which an input will most probably lie, and refers to the interval as the boundary of each MFs. The Inference Engine is based on rules expressed in the form of IF-THEN statements. All the scores inside or outside of the interval have the same degree of normality or abnormality respectively. This means by using the fuzziness concept we relax the problem of exact division of normality and abnormality, which otherwise ends up producing a false alarm.

- **Input membership functions**

  We divide the domain interval into a number of regions (the number of components generated by the EM algorithm). The preliminary score used as input data in this method lies between 0 and 1. We assign each component/interval to an input fuzzy membership function representing the degree of truth. The membership functions map each component into a linguistic variable and describe the degree of membership of an anomaly score in a fuzzy set. Three different input membership functions- S-shape (e.g. first membership function), Z-shape (e.g. last membership function), and trapezoidal (e.g. middle membership functions)- are used for transforming the crisp inputs (components generated by EM) into linguistic variable inputs. The type of a MF is context-dependent and based on data distribution. The membership functions are expressed as below. The
input parameters of each membership function can be derived from the components generated using the EM algorithm.

\[
M_0(\zeta^{(i)}, \alpha_0, \beta_0) = \begin{cases} 
1 & , \zeta^{(i)} \leq \alpha_0 \\
1 - 2\left(\frac{\zeta^{(i)} - \alpha_0}{\beta_0 - \alpha_0}\right)^2, & \alpha_0 \leq \zeta^{(i)} \leq \frac{\beta_0 + \alpha_0}{2} \\
2\left(\frac{\zeta^{(i)} - \alpha_0}{\beta_0 - \alpha_0}\right)^2, & \frac{\beta_0 + \alpha_0}{2} \leq \zeta^{(i)} \leq \beta_0 \\
0 & , \zeta^{(i)} \geq \beta_0
\end{cases}
\] (4.22)

where \( \alpha_0 = \mu_0, \beta_0 = \mu_1 \)

\[
M_1 \left( \zeta^{(i)}, \alpha_1, \beta_1, \gamma_1 \right) = 1/(1 + \left| \frac{\zeta^{(i)} - \gamma_1}{\alpha_1} \right|^{2\beta_1})
\] (4.23)

where \( \alpha_1 = \sigma_1, \beta_1 = \frac{\gamma_1}{\sigma_1}, \gamma_1 = \mu_1 \)

\[
M_2 \left( \zeta^{(i)}, \alpha_2, \beta_2, \gamma_2 \right) = 1/(1 + \left| \frac{\zeta^{(i)} - \gamma_2}{\alpha_2} \right|^{2\beta_2})
\] (4.24)

where \( \alpha_2 = \sigma_2, \beta_2 = \frac{\gamma_2}{\sigma_2}, \gamma_2 = \mu_2 \)

\[
M_3 \left( \zeta^{(i)}, \alpha_3, \beta_3, \gamma_3 \right) = 1/(1 + \left| \frac{\zeta^{(i)} - \gamma_3}{\alpha_3} \right|^{2\beta_3})
\] (4.25)

where \( \alpha_3 = \sigma_3, \beta_3 = \frac{\gamma_3}{\sigma_3}, \gamma_3 = \mu_3 \)

\[
M_4 \left( \zeta^{(i)}, \alpha_4, \beta_4, \gamma_4 \right) = 1/(1 + \left| \frac{\zeta^{(i)} - \gamma_4}{\alpha_4} \right|^{2\beta_4})
\] (4.26)

where \( \alpha_4 = \sigma_4, \beta_4 = \frac{\gamma_4}{\sigma_4}, \gamma_4 = \mu_4 \)

\[
M_5 \left( \zeta^{(i)}, \alpha_5, \beta_5 \right) = \begin{cases} 
0 & , \zeta^{(i)} \leq \alpha_5 \\
2\left(\frac{\zeta^{(i)} - \alpha_5}{\beta_5 - \alpha_5}\right)^2, & \alpha_5 \leq \zeta^{(i)} \leq \frac{\beta_5 + \alpha_5}{2} \\
1 - 2\left(\frac{\zeta^{(i)} - \alpha_5}{\beta_5 - \alpha_5}\right)^2, & \frac{\beta_5 + \alpha_5}{2} \leq \zeta^{(i)} \leq \beta_5 \\
1 & , \zeta^{(i)} \geq \beta_5
\end{cases}
\] (4.27)

where \( \alpha_5 = \mu_4, \beta_5 = \mu_5 \)

- Inference engine
A semi-supervised learning approach is adopted to generate the inference engine rules. To generate the rules we use a pre-labelled patterns approach. In this approach, with expert assistance, a subset of instances is labelled as anomalous and normal. These labelled instances are randomly injected into the dataset before applying EM to generate clusters. Expert visualisation of the generated clusters is then used to find which clusters accommodate the anomalous and which ones do not. The characteristics of the clusters are used for producing membership functions. These characteristics include mean and standard deviation (as shown as parameters in Equations 4.22 to 4.27). Finally the membership functions are utilised as inputs for the inference engine to generate rules and define aScore (Equation 4.23). The process of finding rules needs to be done only once.

After applying this approach, we discovered two clusters associated with pre-labelled anomalous instances in all three datasets. These two clusters with the low value of $\xi_x^{(i)}$ and the high value of $\xi_e^{(i)}$ are predominantly anomalous. The characteristics of these two clusters are used to make the input membership functions $M_0$ and $M_5$. To generate rules we use these input and output membership functions as shown in Figure 25. According to the fuzzy theory, rules are generated by the fuzzy reasoning component which performs several fuzzy logic operations such as resolver, computation, and derivation to infer the output from the inputs. Resolver is to control the matching degree between the input and the pre-defined fuzzy sets for each input variable. Computation is to calculate relevance degree for each rule based on the matching degree and the operations (e.g. AND, OR) used with input variables in the antecedent rule. Derivation is to control outputs based on the computed fire strength and the pre-defined fuzzy sets.
for each output in the consequent rule. The antecedent and consequent rule is defined as follows:

\[
aScore = \begin{cases} 
\text{ano}, & \zeta^{(i)}_s = \text{low} \; \text{or} \; \zeta^{(i)}_c = \text{high} \\
\text{nor}, & \text{otherwise}
\end{cases}
\]  

(4.28)

Users receive an anomaly score based on Equation 4.28 and the components they are fitted in. If their aScore value matched in the components represented by lower and higher membership functions, they are called anomalies; if not, they are called normal.
4.4 CLUSTERING-BASED ANOMALY DETECTION IN ONLINE-SOCIAL-NETWORK GRAPHS

As shown in Figure 26, this method is categorised under a clustering-based method. Clustering is used to categorise instances into the same group if they are similar or into a different group if they are dissimilar. There are two main techniques in clustering-based approaches (Patcha & Park, 2007). The first approach is based on the assumptions that anomalies form a small portion of the dataset and that they can be detected by the size of clusters. The normal data belongs to the bigger clusters and anomalies belong to small ones. In the second approach, anomalies will by assumption deviate from normal instances and therefore form different clusters. The main advantages that clustering-based approaches provide are the ability to learn from unlabelled data and working with multidimensional datasets better.
Different measures of similarity (e.g. density, distance, and connections) may be used depending upon the nature of the data and the purpose of clustering group instances. The similarity measure also plays a key role in how to form the clusters. This can decrease the accuracy in solving the anomaly detection problem. Clustering algorithms can be divided into hard and soft (fuzzy). In the hard clustering, any instance can belong to only one cluster, while in fuzzy clustering instances can belong to more than one cluster, with different membership degrees ranging from 0 (not belongs) to 1 (fully belongs). As with many algorithms for the application of anomaly detection, fuzzy clustering-based algorithms seem to work better when we deal with uncertainty.

Fuzzy clustering-based algorithms are flexible and can detect anomalies more objectively (Wang, Hao, Ma, & Huang, 2010). In fuzzy clustering, a cluster is denoted by a fuzzy subset of a set of objects. Each object has a “degree of membership” or a “degree of belongingness” for each cluster. The membership degree or cluster-fitting likelihood is computed as a key part of a fuzzy clustering algorithm to fit instances to the proper clusters. Having a membership degree enables us to deal with the fuzzy nature of instances (e.g. online social networks characteristics) in the given datasets.

For instance, the fuzzy c-means (FCM) algorithm is one of the most general fuzzy clustering methods used to deal with uncertainty of data. FCM is a data clustering technique in which a dataset is grouped into c clusters, with every data point in the dataset belonging to a certain degree to every cluster. In FCM the number of clusters is not automatically selected; therefore this research uses the expectation-maximization technique to find the best number of clusters for given
datasets. Moreover, to produce rules from the generated membership degree related to each cluster, we suggest using of the fuzzy logic inference engine.

### 4.4.1 METHOD OVERVIEW

The proposed fuzzy-based unsupervised anomaly detection method follows the standard definition of anomaly; that is, behaviour not followed by the majority can be called anomalies. The aim is to distinguish between the behaviour followed by the majority of users in the network, and the behaviour followed by the minority. To deal with the fuzzy characteristic, we propose to use the FCM algorithm to cluster the users according to the scScore that they hold. A requirement of FCM is the setting of cluster numbers apriori. We use the expectation-maximization algorithm and log likelihood metrics to determine the natural number of clusters. The clusters (or users within) require labelling to represent the anomalous users; therefore, each generated cluster is represented by a fuzzy membership function. The degree of anomaly is determined by the lower and upper bound of the membership functions, which can define a cluster as normal or anomalous. Figure 27 provides an overview of the proposed algorithm. We proposed FCM to cluster the data points and EM to find the number of clusters given to FCM. The time complexity of steps one to eight is proportional to the degree of nodes $O(N)$, as each node is visited once for analysis independent of the others. The complexity of EM is $O(N \log N)$ where $N = |V|$ is the number nodes in the graph, and that of FCM is $O(DimNC)$ where $C$ is the number of clusters ($C << N$) and $Dim$ is the dimension of dataset ($Dim << N$).
A semi-supervised learning approach is adopted to generate the inference engine rules. To generate the rules we use a pre-labelled patterns approach. In this approach, with experts’ assistance, subsets of instances are labelled as anomalous (e.g. star or clique topology) and normal (friends of friends are often friends). These labelled instances are injected into the dataset before applying FCM to generate clusters.
4.4.2 INPUT TO ALGORITHM

The input to the method is an online social network in which user relationships are modelled as the graph \( \mathcal{G} = (\mathcal{V}, \mathcal{E}) \). The graph \( \mathcal{G} \) consists of a set \( \mathcal{V} \) of vertices (nodes) and a set \( \mathcal{E} \) of edges where each \( e \in \mathcal{E} \) is an unordered pair of distinct vertices. As described in Chapter 3, we compute a preliminary anomaly score (starness, and cliqueness) using the user’s egonet properties, such as the numbers of nodes and edges. Input to this step is scScore, \( \zeta = \{ \zeta^{(1)}, ..., \zeta^{(m)} \} \) where \( \zeta^{(i)} = \zeta_c^{(i)} \) or \( \zeta_s^{(i)} \) and \( m \) is the number of nodes in an egonet \( i \).

4.4.3 FINDING CLUSTER NUMBER USING GMM-EM

In this step we aim to find a natural number of clusters using a Gaussian mixture model with expectation-maximization which is applied independently on both sets of \( \zeta_c^{(i)} \) and \( \zeta_s^{(i)} \) to cluster the instances. A GMM is a parametric probability density function that assumes the data can be modelled as a weighted sum of Gaussian components (Dempster, et al., 1977). Gaussian components parameter can be estimated from iterative expectation-maximization (EM), or Maximum A Posteriori (MAP) using training data. More formally \( P(\zeta | \Theta) = \sum_{g=1}^{M} \omega_g P(\zeta | \theta_g) \), where \( M \) is number of Gaussian components in GMM, \( \omega_g \) is the weight of each Gaussian, and \( \theta_g \) is the estimated parameters. The parameters \( \Theta = (\phi, \mu, \Sigma) \) are solved by utilizing the EM algorithm section 4.3.
4.4.4 CLUSTERING USING FUZZY C-MEANS (FCM)

The fuzzy clustering-based algorithm (FCM) is used to find patterns and to detect the meaningful characteristics in an unsupervised way. FCM is a data clustering technique in which a dataset is grouped into \( c \) clusters, with every data point in the dataset belonging to every cluster to a certain degree (Gath & Geva, 1989). For example, a certain data point that lies close to the centre of a cluster will have a high degree of belonging or membership to that cluster, while another data point that lies far away from the centre of a cluster will have a low degree of belonging or membership to that cluster. The number of clusters \( c \) has to be manually given. However, we use the expectation-maximization technique to find \( c \). FCM assigns every data point a membership degree between 0 and 1 for each cluster and iteratively updates the cluster centres and the membership grades. This iteration is based on minimizing an objective function (equation 4.24) that represents the distance from any given data point to a cluster centre weighted by that data point's membership grade. Due to the local nature of the centroid based clustering approach, results can be dependent on the starting cluster seeds.

\[
\arg\min J(U, \mu) = \sum_{i=1}^{n} \sum_{j=1}^{c} (m_{ij})^\beta d^2(\xi^{(i)}, \mu_j) \tag{4.30}
\]

\[
m_{ij} = \frac{1}{\sum_{k=1}^{c} \left( \frac{d^2(\xi^{(i)}, \mu_k)}{\sum_{l=1}^{c} d^2(\xi^{(i)}, \mu_l)} \right)^{\frac{1}{\beta}}} \tag{4.31}
\]

where \( m_{ij} \in [0,1] \); \( \sum_{j=1}^{c} (m_{ij}) = 1 \); \( U \) represents a matrix of the fuzzy membership \( (m_{ij})_{n \times c} \); \( n \) is the number of data points; \( \beta \) is the fuzziness index \( \beta \in [1,\infty] \), usually \( \beta = 2 \) is considered a good setting for most of the problems. \( \beta \) must satisfy the condition of \( \beta > 1 \) based on the Equation 4.26. In order to achieve...
better performance, an appropriate $\beta$ value should be selected. Many approaches have been recommended in the literature. For example, Pal and Bezdek (1995) have given heuristic guidelines to choose the best $\beta$, suggesting that it should be in the interval $[1.5, 2.5]$.

$$\mu_j = \frac{\sum_{i=1}^{n}(m_{ij})^\beta \bar{X}^{(i)}}{\sum_{i=1}^{n}(m_{ij})^\beta}, \quad j = 1, 2, ..., c$$  \hspace{1cm} (4.32)

Where $d^2(\bar{X}^{(i)}, \mu_j)$ is the distance between $i$th data and $j$th cluster center ($\mu$).

$$d^2(\bar{X}^{(i)}, \mu_j) = (\bar{X}^{(i)} - \mu_j)^TF_j(\bar{X}^{(i)} - \mu_j) = \| \bar{X}^{(i)} - \mu_j \|^2$$  \hspace{1cm} (4.33)

where $F_j$ is the fuzzy covariance; for Euclidian distance $F_j$ is the identity matrix ($F_j = I$); for an exponential distance which is based on maximum likelihood estimation $F_i$ (Gath & Geva, 1989). Distance is calculated as follows using equation 4.28:

$$d^2(\bar{X}^{(i)}, \mu_j) = \left[ \frac{\text{det}(F_j)^{1/2}}{\varrho_j} \exp \frac{1}{2} \left[ (\bar{X}^{(i)} - \mu_j)^TF_j^{-1}(\bar{X}^{(i)} - \mu_j) \right] \right]$$  \hspace{1cm} (4.34)

where: $\varrho_j = \frac{1}{N} \sum_{i=1}^{N} L (j \mid \bar{X}^{(i)})$;

where $\varrho_j$ is probability of instance $i$ to be clustered in $j$.

$$L (j \mid \bar{X}^{(i)}) = 1/d^2(\bar{X}^{(i)}, \mu_j)/\sum_{k=1}^{c} 1/d^2(\bar{X}^{(i)}, \mu_k);$$  \hspace{1cm} (4.35)

where the $L (j \mid \bar{X}^{(i)})$ is the posterior probability (the probability of selecting the $j$th cluster given the $\bar{X}^{(i)}$):

$$F_j = \frac{\sum_{i=1}^{N} L (j \mid \bar{X}^{(i)})(\bar{X}^{(i)} - \mu_j)^T(\bar{X}^{(i)} - \mu_j)}{\sum_{i=1}^{N} L (j \mid \bar{X}^{(i)})}$$  \hspace{1cm} (4.36)
Figure 28 shows the steps needed to implement and apply FCM (Gath & Geva, 1989) on a dataset using the number of clusters c obtained from the expectation-maximization technique. This can improve the quality of clusters as the expectation-maximization technique uses the natural underlying relationship between instances for finding the number of clusters. However, EM cannot deal with the fuzzy nature of the dataset including the friendship relations. Therefore we suggest using FCM to handle this problem. The use of the FCM algorithm improves the quality of clusters of the fuzzy nature dataset (e.g. degree of friendship). By using membership degree we can better cluster the overlapping instances between the clusters. The membership functions generated by FCM are used as inputs to a fuzzy inference engine in order to produce rules needed for finding anomalies in given clusters. The methods for computing starness and cliqueness are called EM_FCM_S, and EM_FCM_C respectively.

4.4.5 REPRESENTING CLUSTERS WITH FUZZY INFERENCE ENGINE

Normal and suspicious behaviours, especially in online social networks, are hard to differentiate, as the boundaries between them cannot be accurately defined (Kriegel & Pfeifle, 2005). In this step we propose to use a fuzzy inference engine and fuzzy membership function to generate rules and to label the normal and anomalous regions. An observation would be called an anomaly or normal if it falls into a certain interval. The intervals are represented by membership functions (MFs).
Algorithm Fuzzy C-Means (FCM)

\textbf{Input:} $\zeta$, Threshold: $\mathcal{E}$, Max\_Iter, $c$, centroids: CNT, distance: DIS, membership: MEM, pervious_membership: PMEM, fuzzy_covariance: FCOV, $\mu_x$: means for cluster $x$

\textbf{Output:} MEM membership degree matrix

\begin{enumerate}
\item Initialise MEM, CNT, DIS, PMEM, FCON, Sall to zero
\item $m=\text{Sizeof}(\zeta)$
\item For $j = 1$ to $c$ do
\item For $i = 1$ to $m$ do
\item $\text{DIS}[j, i] = (\zeta^{(i)} - \mu_j)^T (\zeta^{(i)} - \mu_j)$
\item $\text{Sall} = \text{Sall} + 1/ \text{DIS}[j, i]$ // where Sall : sum of all
\item $\text{MEM}[i, j] = (1/\text{DIS}[j, i]) / \text{Sall}$
\item End
\item End
\item While ((objective $> \mathcal{E}$ & (iterations $< \text{Max\_Iter}$))
\item For $j = 1$ to $c$ do
\item For $i = 1$ to $m$ do
\item $\rho_i = \rho_i + \text{MEM}[i, j] / m$
\item End
\item End
\item For $i = 1$ to $m$ do
\item $\text{DIS}[j, i] = (\zeta^{(i)} - \mu_j)^T (\zeta^{(i)} - \mu_j)$
\item $\text{FCOV}_i = \text{FCOV}_i + \text{MEM}[i, j]\text{DIS}[j, i]$
\item $\text{FCOV}_D_i = \text{FCOV}_D_i + \text{MEM}[i, j]$
\item End
\item $\text{FCOV}_i = \frac{\text{FCOV}_i}{\text{FCOV}_D_i}$
\item $\text{SFD}_i = (\det(\text{FCOV}_i))^{1/2}$ // where det : determinant
\item For $i = 1$ to $m$ do
\item $\text{DIS}[j, i] = \frac{\text{SFD}_i}{\rho_i} \exp((\zeta^{(i)} - \mu_j)^T \text{FCOV}_i^{-1} (\zeta^{(i)} - \mu_j)/2)$
\item End
\item $\text{PMEM}[i, j] = \text{MEM}[i, j]$
\item For $i = 1$ to $m$ do
\item $\text{Sall} = \text{Sall} + 1/ \text{DIS}[j, i]$ // where Sall : sum of all
\item $\text{MEM}[i, j] = (1/\text{DIS}[j, i]) / \text{Sall}$
\item End
\item Objective $= ||\text{PMEM}[i, j] - \text{MEM}[i, j]||$
\item Iterations = Iterations + 1
\item End
\item Return MEM
\item End
\end{enumerate}

Figure 28. Implementation of Fuzzy C-Means (FCM) Algorithm
There are different shapes of membership functions such as triangular, trapezoidal, piecewise linear, Gaussian, or singleton. The inference engine, based on rules, is expressed in the form of IF-THEN statements. All the scores inside or outside of the interval have the same degree of normality or anomaly respectively. This means by using the fuzziness concept we relax the problem of exact division of normality and anomaly, which otherwise ends up in producing a false alarm.

4.5 SUMMARY

In this chapter we introduced three different methods proposed in this research: distance-based, distribution-based and clustering-based. The central advantage of using these three well-known concepts in the anomaly detection area is to analyse and find a suitable approach which can characterise online social networks behaviours. For the distance-based approach, a power-law regression model, residual (distance) to the model, and the orthogonal projection of points on the model are computed to model and find anomalies. Residual or distance to the model shows how far a point sits from the model. Based on this distance an anomaly score is computed to find anomalies. In the distribution-based approach, we use the Gaussian mixture model and fuzzy logic to differentiate between anomalous and normal instances. In the clustering-based approach, we employ a fuzzy clustering to locate anomalies.

The proposed methods for finding anomalies can be divided into distance-based, distribution-based, and clustering-based according to their ground-truth
theory. Different types of data input need different techniques in order to handle
them. The proposed distance-based methods working on discrete instances call an
object an anomaly if it sits a long way away from the generated model. This
simple-to-implement method takes into account the relationships between the
features of extracted graphs, in order to find any common patterns. Considering
the local network features for finding anomalies makes this method fast, with
reasonable accuracy. By computing orthogonal projection and performing
clustering techniques, this method is more accurate than those conducted without
clustering. This method uses a power-law regression model which matches with
the scale-free properties of online social networks. However, this method cannot
deal with fuzziness of behaviours in online social networks.

The proposed distribution-based methods working on continuous instances
use a statistical model and a fuzzy inference engine to call an object abnormal if it
does not follow the generated rules. The combination of these concepts improves
the accuracy of anomaly detection. Different fuzziness coefficients can be used to
adapt the proposed method to new conditions. The rules generated by the fuzzy
inference engine are adjustable to new circumstances in order to improve the
accuracy. However, the rules need expert interference and help in order to be
generated.

The proposed clustering-based method calls an object an anomaly if it fits
the object into a cluster dominated by anomalies. This method shows better
accuracy compared to the distance-based approach, as it uses a combination of
fuzzy logic and clustering. A learning algorithm for finding the number of clusters
is added, thus enhancing the performance. A fuzziness index can be fitted to new
requirements imposed by the given datasets. However, the fuzzy rules should be
generated with help from domain experts.
Chapter 5: Experiments and Discussions

This Chapter presents a discussion on the empirical evaluation of the methods described in the previous Chapter as the third layer of the framework shown in Figure 29. F-Score is used as an evaluation measure for all proposed algorithms, where its highest value of (1) indicates a perfect classification of labelled data and its lowest value of (0) indicates a completely incorrect classification of labelled data. This chapter is organised as follows. First we go through each method briefly, followed by their experiment design and results. The discussion for each method, in terms of strengths and shortcomings, performance comparisons, and summary, are the next sections of this Chapter.

![Figure 29. Layer 3 of Framework-Evaluation](image-url)
5.1 FRAMEWORK

The experiments are designed based on the third layer of the proposed multi-layer framework explained in Chapter 3. The results of the proposed algorithms which are based on distance, distribution, and clustering, which apply to three different real-life instances, are presented in this framework. All experiments were conducted using the labelled datasets. The biggest advantage of using the framework is in comparing the effectiveness of different algorithms in the same environment settings used for the processing steps.

The first step of our experiments, according to the framework, is pre-processing, in which we build the egonet and super-egonet and compute the graph metrics for users’ networks. The network of users is modelled by undirected graph. We find all immediate neighbours of a user and convert this into a graph structure. Non-connected users are removed during the pre-processing steps. Also all the anomaly scores are normalized in order to put them in the same range. Orthogonal projection of instances is computed in order to improve the quality of the distance based method.

Local networks generated for every node, called egonet or super-egonet, have different levels of neighbourhoods as they accommodate only the direct neighbours and direct neighbours of the neighbours. Having a full user network enables us to do more investigation on each node and its network. This can be done through the use of different tools, such as R, Matlab, and NodeXL, in order to compute graph metrics and visualise the networks. Once each node’s local network is built, they will be easily analysed with minimum computational
complexity. At the time that one egonet is processed, only that egonet needs to be in memory. The memory size is considerably less than when the entire network is loaded in the memory for processing.

The third layer also accommodates the process of labelling, applying the proposed and benchmarks methods on the datasets, computing the F-Score and finding the best thresholds that give the highest F-score. The test-bed of instances used in our experiments consists of two broad groups: normal and anomalous instances of various types and sizes of egonet. The size of egonets within our datasets varies from ten to five hundred users. We use the degree of full star and full clique networks as a bottom line for measuring each node’s network deviation. The degree of starness and cliqueness are calculated as $deg(\text{egonet}_i) = n(n-1)/2$ and $deg(\text{egonet}_i) = n - 1$ respectively, where $n$ is the number of nodes, and $deg$ is a function returning the number of edges of an egonet. The degree ($deg$) of an egonet is the number of edges within the egonet, connecting the nodes together as shown in Figure 30.

![Figure 30. A Network with Degree of Eight](image)
5.1.1 DISTANCE-BASED APPROACH

For the distance-based method, distance to the model is computed to detect anomalies. Distance to the model shows how far a point sits from the model. Based on this distance, an anomaly score is computed to spot the anomalies. This method models user relationships with combinations of different graph metrics extracted from user egonets.

The experiment shows better performance, compared to the existing approach, by using average betweenness centrality as a metrics to describe user network topology. To improve accuracy of the pervious approach, we introduced using the orthogonal projection of instances on the regression model and a clustering algorithm. The objects are considered anomalous if they fit in a cluster dominated by anomalies. By using orthogonal projection we are able to overcome the problem of normal instances which sit far away from the fitting line in the distance based approach.

5.1.1.1 EXPERIMENT DESIGN

There are various experiment settings with the distance-based approach. Using a graph of OSN (Facebook, Flickr, Orkut) with $v$ vertices (nodes or users) and $E$ edges, a local network of each user $i$ ($egonet_i$) includes the user’s direct neighbours. For each built egonet, a user’s super-egonet includes the user’s egonet and the egonets of all its neighbours. $R$ is used to compute the graph metrics of each egonet, such as number of nodes and edges, average betweenness centrality, and community. In the distance-based method we compute three components: a
power-law regression model, the orthogonal projection of instances, and the residual to the model. Residual or distance to the model shows how far a point sits from the model. This method consists of the relationship between different graph metrics which are modelled by a power-law regression model as shown in Figure 31. The metrics of each egonet include the number of edges (E), the number of vertices or nodes (N), the average betweenness centrality (ABC), and the Community (Com). This method focuses on finding the best regression model able to accurately characterise the relationship between metrics in given datasets. The power-law distributions model is used to characterise the relationship between the graph metrics. This model, which is aligned with the scale-free nature of online social networks, is used to explain the topological behaviour of a user network.

The orthogonal projections of instances are calculated using minimum distance to the curve. The arguments of the function of computing distance to curve are points of the curve; these instances need to be mapped. The outputs are the closest points identified along the curve to each of the points. The points of the curve saved in a real numeric array which for 2-dimensional curves it will be a list of points (each row of the array is a new point) that define the curve. The points are mapped to the curve, in term of their closest distance, are saved in the real numeric array. These points were mapped to the given curve in terms of the minimum (Euclidean, 2-norm) distance to the curve. For clustering, we use K-means and FCM algorithms in which the number of clusters can be defined manually or using expectation maximization and log likelihood.
a). Facebook - Edges vs Average BC  

b). Facebook - Edges vs Nodes  

c). Flickr - Edges vs Average BC  

d). Flickr - Edges vs Nodes  

e). Orkut - Edges vs Average BC  

f). Orkut - Edges vs Nodes  

Figure 31. Power-Law Regression Model
5.1.1.2 POWER-LAW REGRESSION METHOD RESULTS

As shown in Figure 32, the results for this method find that E_ABC method has the best overall performance across the sampled data, with the highest F-score. The reason behind this performance is the way betweenness centrality is computed. Betweenness centrality can be used for finding the central users in a given graph (Pfeiffer III & Neville, 2011). Users with higher centrality can be called the most prominent role players in the network. The betweenness centrality method, when compared to the others, concentrates more on the shortest path and the dense subnetworks, a base for finding star and clique topology within networks. The relationship between the average betweenness centrality and the number of edges can be a good indicator of a user’s network density. The average betweenness centrality shows the domination of the user, while the number of edges shows the similarity between the user’s network and the starness or cliqueness topology.

![Figure 32. F-Score for Power-Law Regression Method](image-url)
5.1.1.3 ORTHOGONAL PROJECTION METHOD RESULTS

The second method involves computing the orthogonal projection of these graph metrics on a power-law regression model and applying a fuzzy clustering algorithm to them to detect anomalies. As shown in Figure 33, the results for this group show that the best performance belongs to the FCM_OP_EABC method. This method has shown improvement compared to the first group.

![Figure 33. F-Score for Orthogonal Projection Method](image)

To justify this improvement we can point to the fact that the first group is based only on the distance from the regression model, which assumes that the normal points sit near to the regression model and the anomalous ones sit far away. This assumption does not align with the characteristics of our datasets, in which normal points also sit far away from the regression fitting line. To overcome this problem, we use density of points on the regression model instead of distance from the model. To find the higher density areas, a fuzzy clustering algorithm such as fuzzy c-means is applied.
5.1.2 DISTRIBUTION-BASED APPROACH

In the distribution-based approach, we use the Gaussian mixture model and fuzzy logic to differentiate between anomalies and normal instances. The results of applying the proposed methods to three different datasets are used to evaluate their performance in terms of accuracy. Inputs to this method include cliqueness and starness. The cliqueness or local connectivity of a node is a measure that the neighbours of the node are also connected to each other. The cliqueness of a node ranges between 0 and 1 where higher value shows more similarity to the clique. The starness or local disconnection of a node is a measure that shows the connectivity among the neighbours of the node. The starness of a node ranges between 0 and 1 where lower value shows more similarity to star.

5.1.2.1 EXPERIMENT DESIGN

These are a number of experiment settings with the distribution-based approach. Using a graph of OSN (Facebook, Flickr, Orkut) with a set of vertices (nodes or users) and a set of edges, the egonet of each user is built. This approach uses starness and cliqueness as metrics to measure similarity between the topology of a given egonet and the full star and clique topology. After modelling online social networks using a graph we use a statistical distribution to model the instances. This method calls instances anomalies if they deviate from the model, which is based on probability distribution. Required parameters of the model such as a Gaussian mixture model are learned during training process. To estimate the parameters and generate components an expectation-maximization algorithm is used. Fuzzy logic is applied on the generated components to characterise the
degree of abnormality using membership functions. The use of fuzzy logic allows us to handle the quantitative features, which can be considered as a fuzzy variable.

As shown in Figure 34 EM divides the three different datasets into six components based on natural relationships between instances, and confirmed by the log-likelihood measure. Log likelihood is used to determine the number of components that defines the best-fit model. For many algorithms using the likelihood function, it is easier to work with the logarithm of the likelihood, called the log-likelihood, instead of the likelihood function itself. The reasons behind it are that (1) both likelihood and log-likelihood functions can achieve the maximum value at the same points, and (2) taking the derivative of a function and finding the parameters that maximise the likelihood are easier using log-likelihood than using likelihood.

![Figure 34: Number Components vs Log Likelihood](image)

**Figure 34. Number Components vs Log Likelihood**

When applying EM on the three datasets used in this thesis, there is an increase in the log likelihood with an increase in the number of components until this reaches six. There is a very minor increase in log likelihood when the number goes from six to seven. However, the change is not big enough to be considered
for adding more components. Therefore, six components can be justified as the best choice for the number of clusters (c) within all these given datasets.

EM and fuzzy membership function are used to associate the discovered components for both the starness and the cliqueness scores with the membership functions. We discovered two predominantly anomalous components which are associated with the first and last input membership functions $M_0$ and $M_5$ (Figure 35) for the starness and cliqueness score respectively. $M_0$ is associated with $\zeta_s^{(c)}$, the score of starness, and $M_5$ is associated with $\zeta_c^{(c)}$, the score of cliqueness. Therefore, the output membership after combination of these two score is divided into three functions: (1) $ano_s$ which links to $M_0$; (2) $nor_{sc}$ which links to $M_1$ to $M_4$; and (3) $ano_c$ which links to $M_5$. These three output membership functions $ano_s$, $nor_{sc}$, and $ano_c$ are defined as a Z-shape, a trapezoidal, and a S-shape (Figure 36). As shown in Figure 35, the preliminary score used as input data in this method lies between 0 and 1. We assign each component/interval to an input fuzzy membership function which represents the degree of truth. The membership functions map each component into a linguistic variable and describe the degree of membership of an anomaly score in a fuzzy set. Three different input membership functions, Z-shape (e.g. $M_0$), S-shape (e.g. $M_5$), and trapezoidal (e.g. $M_1$ to $M_4$), are used for transforming the crisp inputs (components generated by EM) into linguistic variable inputs. The Type of a MF is context-dependent and based on data distribution.
5.1.2.2 DISTRIBUTION METHOD RESULTS

We statically analyse each component as shown in Table 6. For each component we show mean ($\mu$), standard deviation ($\sigma$) and percentage of instances (Po).
Table 6. Statistical Information

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Starness / Cliqueness</th>
<th>Components/Statistical information</th>
<th>$C_0$</th>
<th>$C_1$</th>
<th>$C_2$</th>
<th>$C_3$</th>
<th>$C_4$</th>
<th>$C_5$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Facebook</td>
<td>$\zeta_e^{(i)}$</td>
<td>$\mu$</td>
<td>0.14</td>
<td>0.24</td>
<td>0.33</td>
<td>0.46</td>
<td>0.66</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>$\zeta_s^{(i)}$</td>
<td>$\sigma$</td>
<td>0.04</td>
<td>0.03</td>
<td>0.04</td>
<td>0.05</td>
<td>0.10</td>
<td>0.18</td>
</tr>
<tr>
<td></td>
<td></td>
<td>P o</td>
<td>38%</td>
<td>32%</td>
<td>13%</td>
<td>9%</td>
<td>6%</td>
<td>2%</td>
</tr>
<tr>
<td></td>
<td>$\zeta_e^{(d)}$</td>
<td>$\mu$</td>
<td>0.05</td>
<td>0.11</td>
<td>0.17</td>
<td>0.27</td>
<td>0.39</td>
<td>0.53</td>
</tr>
<tr>
<td></td>
<td>$\zeta_s^{(d)}$</td>
<td>$\sigma$</td>
<td>0.01</td>
<td>0.02</td>
<td>0.04</td>
<td>0.06</td>
<td>0.07</td>
<td>0.13</td>
</tr>
<tr>
<td></td>
<td></td>
<td>P o</td>
<td>29%</td>
<td>32%</td>
<td>21%</td>
<td>10%</td>
<td>5%</td>
<td>3%</td>
</tr>
<tr>
<td>Flickr</td>
<td>$\zeta_e^{(i)}$</td>
<td>$\mu$</td>
<td>0.08</td>
<td>0.15</td>
<td>0.25</td>
<td>0.41</td>
<td>0.67</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>$\zeta_s^{(i)}$</td>
<td>$\sigma$</td>
<td>0.02</td>
<td>0.03</td>
<td>0.04</td>
<td>0.07</td>
<td>0.10</td>
<td>0.22</td>
</tr>
<tr>
<td></td>
<td></td>
<td>P o</td>
<td>44%</td>
<td>25%</td>
<td>16%</td>
<td>7%</td>
<td>3%</td>
<td>5%</td>
</tr>
<tr>
<td></td>
<td>$\zeta_e^{(d)}$</td>
<td>$\mu$</td>
<td>0.0051</td>
<td>0.0136</td>
<td>0.03</td>
<td>0.06</td>
<td>0.15</td>
<td>0.44</td>
</tr>
<tr>
<td></td>
<td>$\zeta_s^{(d)}$</td>
<td>$\sigma$</td>
<td>0.0022</td>
<td>0.0053</td>
<td>0.01</td>
<td>0.02</td>
<td>0.06</td>
<td>0.21</td>
</tr>
<tr>
<td></td>
<td></td>
<td>P o</td>
<td>24%</td>
<td>27%</td>
<td>29%</td>
<td>16%</td>
<td>4%</td>
<td>1%</td>
</tr>
<tr>
<td>Orkut</td>
<td>$\zeta_e^{(i)}$</td>
<td>$\mu$</td>
<td>0.0119</td>
<td>0.1123</td>
<td>0.18</td>
<td>0.26</td>
<td>0.39</td>
<td>0.65</td>
</tr>
<tr>
<td></td>
<td>$\zeta_s^{(i)}$</td>
<td>$\sigma$</td>
<td>0.0048</td>
<td>0.041</td>
<td>0.04</td>
<td>0.06</td>
<td>0.09</td>
<td>0.14</td>
</tr>
<tr>
<td></td>
<td></td>
<td>P o</td>
<td>2%</td>
<td>51%</td>
<td>28%</td>
<td>12%</td>
<td>5%</td>
<td>2%</td>
</tr>
<tr>
<td></td>
<td>$\zeta_e^{(d)}$</td>
<td>$\mu$</td>
<td>0.0518</td>
<td>0.0961</td>
<td>0.15</td>
<td>0.25</td>
<td>0.38</td>
<td>0.56</td>
</tr>
<tr>
<td></td>
<td>$\zeta_s^{(d)}$</td>
<td>$\sigma$</td>
<td>0.0139</td>
<td>0.0267</td>
<td>0.04</td>
<td>0.06</td>
<td>0.10</td>
<td>0.15</td>
</tr>
<tr>
<td></td>
<td></td>
<td>P o</td>
<td>13%</td>
<td>29%</td>
<td>30%</td>
<td>17%</td>
<td>9%</td>
<td>3%</td>
</tr>
</tbody>
</table>

Figure 37 shows the results of applying the distribution-based method, which is based on the combination of expectation–maximization algorithm and logic fuzzy, to the starness and cliqueness scores. In this method, the EM_FUZZY outperforms the other methods’ performance in terms of accuracy of detecting anomalies. Using a combination of EM and fuzzy logic is the reason for this improvement, as together they can deal very well with the fuzzy characteristic of...
the starness and cliqueness scores and their natural relationships. EM is used to
find the natural relationship between the scores in an unsupervised way. EM is
employed to cluster the score based on unobserved latent variables which can be a
cluster number. EM assumes there is a joint distribution between each score and
the latent variable that places them together.

Fuzzy logic is used to characterise the fuzziness of the scores, and
representing the components, which are generated by EM algorithm, with fuzzy
membership functions. Representing the components enables this method to deal
with the fuzzy nature of the score in an efficient way using linguistic and
quantitative variables. One underlying accuracy improvement of this method is its
capability to use linguistic and quantitative variables to symbolise uncertainty
such as how much a given topology is similar to the star or clique. The details of
these methods are explained in the previous section.

Figure 37. F-Score for Distribution-Based Method
5.1.3 CLUSTERING-BASED APPROACH

In the clustering-based method, we employ fuzzy clustering and k-means to locate anomalies and verify the effectiveness of using clustering algorithms in general. In this experiment we apply our clustering-based approach on the datasets and illustrate its performance. Clustering-based algorithms are used to detect meaningful patterns in an unsupervised way. They are also able to learn and work with unlabelled data and multidimensional datasets. In our approach, the clustering algorithm is applied on both the normal and anomalous instances. The results of applying the proposed methods on three datasets are used to evaluate their performance in terms of accuracy.

K-means clustering, which generates a set of k disjoint clusters is numerical, unsupervised, non-deterministic, iterative algorithm, and categorised as a hard clustering. K-means is an unsupervised learning algorithm that classifies a given dataset over a pre-defined number of clusters (e.g. k clusters with k centroids). The main idea is that the centroids should be placed as far away from each other as possible, with each point associated to the nearest centroid. Through the iterations, k new centroids have been changed until convergence, meaning that there is no need any further moving of centroids.

A fuzzy clustering-based algorithm such as FCM is used to find patterns and detect the meaningful characteristics in an unsupervised way. FCM is a data clustering technique in which a dataset is grouped into c clusters, with every data point in the dataset belonging to every cluster to a certain degree. For example, an instance that sits close to the centre of any cluster will have a high degree of membership to that cluster and a lower degree to the other clusters. FCM assigns
every data point a membership degree and iteratively updates the cluster centres and the membership grades based on minimizing the distance between the instances and the centroids (objective function). Due to the local nature of the centroid based clustering approach, results can be dependent on the starting cluster seeds. The number of clusters in FCM needs to be defined manually; however, using EM and a log likelihood measure can be used to find the number of clusters.

In the hard clustering, any instances can only belong to one cluster while in fuzzy clustering, instances can belong to more than one cluster with different membership degrees ranging from 0 (not belongs) to 1 (fully belongs). The membership degree or cluster fitting likelihood is computed as a key part of a fuzzy clustering algorithm to fit instances to the proper clusters.

5.1.3.1 EXPERIMENT DESIGN

The weakness of FCM is that the number of clusters $c$ has to be manually specified. However, we use EM algorithms to overcome this problem and find the $c$. The maximum iteration is set to be 100 as several experiences show that the objective function is converged less than 100 iterations. The fuzziness index $\beta$ is set to 2 as it is considered a good setting for most of the problems (Gath & Geva, 1989). We convert the membership degree generated by FCM to a fuzzy membership function in order to label each cluster and intervals linguistically. The generated membership degrees assigned to standard fuzzy membership functions represents the degree of truth. The membership functions map each component
into a linguistic variable and describe the degree of membership of an anomaly score in a fuzzy set.

5.1.3.2 CLUSTERING METHOD RESULTS

In the third method a customised FCM, which is a soft clustering technique, is developed in order to improve the accuracy of the clustering-based anomaly detection. K-means, which is a hard clustering technique, is also used in order to verify the effectiveness of using clustering algorithms in general for detecting anomalies. Moreover, we compare the results of FCM and k-means to assess the efficiency of the soft clustering against the hard clustering algorithms. This helps us to analyse whether clustering algorithms are beneficial for improving the accuracy of detecting anomalies.

These algorithms are applied on the computed orthogonal projection of a user’s local graph metrics. The metrics of each egonet include the number of edges (E), the number of vertices or nodes (N), the average betweenness centrality (ABC), and the community (Com). As shown in Figure 38 to Figure 40, the experiments for these methods confirm that clustering can improve the accuracy about thirty percent on average compare to the OddBall method. In this experiment we repeat our experiments for both FCM and k-means with the same datasets. The results show better performance when clustering algorithms are applied to the orthogonal projections than the non-orthogonal projections. This is because the application of the clustering algorithm to non-orthogonal projection instances leads to missing normal instances that sit further away from the fitting line. The experiment results confirm that using a clustering algorithm is beneficial
for improving the accuracy of detecting anomalies in the distance-based methods. However, using the fuzzy clustering (soft clustering) algorithm can improve the accuracy more than hard clustering does.

Figure 38. F-Score for K-means Clustering Method Using Projected Instances

---

Figure 39. F-Score for FCM Clustering Method
5.2 EXPERIMENT RESULTS DISCUSSION

This discussion in this section is based on Table 8 to Table 10. The proposed hybrid methods are used to detect anomalous users in online social networks. In these methods we compute and use the graph metrics, the relationship between the metrics, the power-law regression model, orthogonal projection, the distribution model, fuzzy logic and clustering algorithms. Different combinations of techniques are used based on the type of input data used by each method. A subset of the expert-labelled dataset, including normal and anomalous users, is used in the threshold finding to identify for each method the best threshold outlier score that maximises F-score. The resulting F-Score is used to compare the different approaches. We compare the proposed methods with different approaches including OddBall (Akoglu, et al., 2010). The results are categorized into different groups according to their underlying approaches, which are based on distance, distribution and clustering. The groups include the power-
law regression model, orthogonal projection, clustering, non-orthogonal projection, and the distribution-based group. Each group includes different methods which are based on the same underlying concepts. Each group is presented in different sections in which the computed F-Score of each method applied to three datasets is shown. More details of each method, including precision and recall, are displayed in different tables in the discussion section. For every dataset we discuss and show the results of the top performing methods.

5.2.1 POWER-LAW DEGREE AND NORMAL INSTANCES DISTRIBUTION

As well as all sampled graphs being constructed from real-life data, another commonality among them is that they obey a power-law degree distribution. This means nodes with small degrees are most frequent and new nodes tend to connect to nodes with a large degree, as shown in Figure 41, Figure 42, and Figure 43. Moreover, Table 7 shows the distribution across the whole datasets of normal and anomalous instances which are randomly selected and visualised in order to label them.

Figure 41. Facebook Degree Distribution
Chapter 5: Experiments and Discussions

Figure 42. Flickr Degree Distribution

Figure 43. Orkut Degree Distribution

Table 7. Normal and Anomaly Distribution

<table>
<thead>
<tr>
<th>Dataset</th>
<th>anomaly(%)</th>
<th>normal(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Facebook</td>
<td>12</td>
<td>88</td>
</tr>
<tr>
<td>Flickr</td>
<td>9</td>
<td>91</td>
</tr>
<tr>
<td>Orkut</td>
<td>4</td>
<td>96</td>
</tr>
</tbody>
</table>
Table 8. Facebook Result

<table>
<thead>
<tr>
<th>Approaches</th>
<th>P (%)</th>
<th>R (%)</th>
<th>F-Score (%)</th>
<th>Metrics</th>
<th>P (%)</th>
<th>R (%)</th>
<th>F-Score (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group 1</td>
<td></td>
<td></td>
<td></td>
<td>Group 4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>N vs. E (OddBall)</td>
<td>59.32</td>
<td>81.4</td>
<td>68.63</td>
<td>FCM_NOP_EABC</td>
<td>84.00</td>
<td>97.67</td>
<td>90.32</td>
</tr>
<tr>
<td>E vs. ABC</td>
<td>84.78</td>
<td>90.70</td>
<td>87.64</td>
<td>FCM_NOP_EV</td>
<td>66.04</td>
<td>81.40</td>
<td>72.92</td>
</tr>
<tr>
<td>N vs. Com</td>
<td>53.33</td>
<td>74.42</td>
<td>62.14</td>
<td>FCM_NOP_NABC</td>
<td>60.00</td>
<td>76.74</td>
<td>67.35</td>
</tr>
<tr>
<td>N vs. ABC</td>
<td>47.89</td>
<td>79.07</td>
<td>59.65</td>
<td>FCM_NOP_VCOM</td>
<td>48.84</td>
<td>97.67</td>
<td>65.12</td>
</tr>
<tr>
<td>E vs. Com</td>
<td>42.22</td>
<td>88.37</td>
<td>57.14</td>
<td>FCM_NOP_ECOM</td>
<td>56.14</td>
<td>68.09</td>
<td>61.54</td>
</tr>
<tr>
<td>Group 2</td>
<td></td>
<td></td>
<td></td>
<td>Group 5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FCM_OP_EABC</td>
<td>86.00</td>
<td>100.00</td>
<td>92.47</td>
<td>kmeans_NOP_EABC</td>
<td>83.33</td>
<td>93.02</td>
<td>87.91</td>
</tr>
<tr>
<td>FCM_OP_NABC</td>
<td>62.07</td>
<td>83.72</td>
<td>71.29</td>
<td>kmeans_NOP_EV</td>
<td>60.02</td>
<td>83.72</td>
<td>69.23</td>
</tr>
<tr>
<td>FCM_OP_EV</td>
<td>69.81</td>
<td>86.05</td>
<td>77.08</td>
<td>kmeans_NOP_VCOM</td>
<td>52.31</td>
<td>79.07</td>
<td>62.96</td>
</tr>
<tr>
<td>FCM_OP_ECOM</td>
<td>65.22</td>
<td>69.77</td>
<td>67.42</td>
<td>kmeans_NOP_NABC</td>
<td>54.72</td>
<td>67.44</td>
<td>60.42</td>
</tr>
<tr>
<td>FCM_OP_VCOM</td>
<td>62.50</td>
<td>81.40</td>
<td>70.71</td>
<td>kmeans_NOP_ECOM</td>
<td>58.06</td>
<td>59.02</td>
<td>58.54</td>
</tr>
<tr>
<td>Group 3</td>
<td></td>
<td></td>
<td></td>
<td>Group 6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>kmeans_OP_EABC</td>
<td>82.00</td>
<td>95.35</td>
<td>88.17</td>
<td>EM_FUZZY_S</td>
<td>98.77</td>
<td>97.56</td>
<td>98.16</td>
</tr>
<tr>
<td>kmeans_OP_EV</td>
<td>63.64</td>
<td>81.40</td>
<td>71.43</td>
<td>EM_FUZZY_C</td>
<td>96.49</td>
<td>94.83</td>
<td>95.65</td>
</tr>
<tr>
<td>kmeans_OP_VCOM</td>
<td>55.93</td>
<td>76.74</td>
<td>64.71</td>
<td>EM_FCM_C</td>
<td>91.23</td>
<td>89.66</td>
<td>90.43</td>
</tr>
<tr>
<td>kmeans_OP_NABC</td>
<td>54.24</td>
<td>74.42</td>
<td>62.75</td>
<td>EM_FCM_S</td>
<td>87.50</td>
<td>92.45</td>
<td>89.91</td>
</tr>
<tr>
<td>kmeans_OP_ECOM</td>
<td>59.09</td>
<td>60.47</td>
<td>59.77</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Table 9. Flickr Result

<table>
<thead>
<tr>
<th>Approaches</th>
<th>P (%)</th>
<th>R (%)</th>
<th>F-Score (%)</th>
<th>Metrics</th>
<th>P (%)</th>
<th>R (%)</th>
<th>F-Score (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>N vs. E (OddBall)</td>
<td>44.44</td>
<td>97.78</td>
<td>61.11</td>
<td>FCM_NOP_EABC</td>
<td>96.00</td>
<td>87.27</td>
<td>91.43</td>
</tr>
<tr>
<td>E vs. ABC</td>
<td>90.91</td>
<td>88.89</td>
<td>89.89</td>
<td>FCM_NOP_EV</td>
<td>70.91</td>
<td>81.25</td>
<td>75.73</td>
</tr>
<tr>
<td>N vs. Com</td>
<td>54.67</td>
<td>91.11</td>
<td>68.33</td>
<td>FCM_NOP_NABC</td>
<td>71.70</td>
<td>79.17</td>
<td>75.25</td>
</tr>
<tr>
<td>N vs. ABC</td>
<td>57.81</td>
<td>82.22</td>
<td>67.89</td>
<td>FCM_NOP_VCOM</td>
<td>82.22</td>
<td>57.81</td>
<td>67.89</td>
</tr>
<tr>
<td>E vs. Com</td>
<td>44.79</td>
<td>95.56</td>
<td>60.99</td>
<td>FCM_NOP_ECOM</td>
<td>52.00</td>
<td>86.67</td>
<td>65.00</td>
</tr>
<tr>
<td>Group 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FCM_OP_EABC</td>
<td>98.00</td>
<td>89.09</td>
<td>93.33</td>
<td>kmeans_NOP_EABC</td>
<td>92.45</td>
<td>89.09</td>
<td>90.74</td>
</tr>
<tr>
<td>FCM_OP_NABC</td>
<td>80.36</td>
<td>81.82</td>
<td>81.08</td>
<td>kmeans_NOP_EV</td>
<td>67.39</td>
<td>73.81</td>
<td>70.45</td>
</tr>
<tr>
<td>FCM_OP_EV</td>
<td>70.00</td>
<td>77.78</td>
<td>73.68</td>
<td>kmeans_NOP_VCOM</td>
<td>49.12</td>
<td>87.50</td>
<td>62.92</td>
</tr>
<tr>
<td>FCM_OP_ECOM</td>
<td>84.09</td>
<td>67.27</td>
<td>74.75</td>
<td>kmeans_NOP_NABC</td>
<td>62.07</td>
<td>61.02</td>
<td>61.54</td>
</tr>
<tr>
<td>FCM_OP_VCOM</td>
<td>82.00</td>
<td>78.85</td>
<td>80.39</td>
<td>kmeans_NOP_ECOM</td>
<td>69.09</td>
<td>74.51</td>
<td>71.70</td>
</tr>
<tr>
<td>Group 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>kmeans_OP_EABC</td>
<td>97.92</td>
<td>85.45</td>
<td>91.26</td>
<td>EM_FUZZY_S</td>
<td>96.36</td>
<td>94.64</td>
<td>95.50</td>
</tr>
<tr>
<td>kmeans_OP_EV</td>
<td>72.73</td>
<td>76.92</td>
<td>74.77</td>
<td>EM_FUZZY_C</td>
<td>89.09</td>
<td>100.00</td>
<td>94.23</td>
</tr>
<tr>
<td>kmeans_OP_VCOM</td>
<td>61.29</td>
<td>88.37</td>
<td>72.38</td>
<td>EM_FCM_C</td>
<td>96.36</td>
<td>88.33</td>
<td>92.17</td>
</tr>
<tr>
<td>kmeans_OP_NABC</td>
<td>56.60</td>
<td>75.00</td>
<td>64.52</td>
<td>EM_FCM_S</td>
<td>89.09</td>
<td>96.08</td>
<td>92.45</td>
</tr>
<tr>
<td>kmeans_OP_ECOM</td>
<td>72.00</td>
<td>58.06</td>
<td>64.29</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table 10. Orkut result

<table>
<thead>
<tr>
<th>Approaches</th>
<th>P (%)</th>
<th>R (%)</th>
<th>F-Score (%)</th>
<th>Metrics</th>
<th>P (%)</th>
<th>R (%)</th>
<th>F-Score (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Group 1</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>N vs. E (OddBall)</td>
<td>32.61</td>
<td>85.71</td>
<td>47.24</td>
<td>FCM_NOP_EABC</td>
<td>98.21</td>
<td>88.71</td>
<td>93.22</td>
</tr>
<tr>
<td>E vs. ABC</td>
<td>89.19</td>
<td>94.19</td>
<td>91.67</td>
<td>FCM_NOP_EV</td>
<td>68.42</td>
<td>60.00</td>
<td>63.93</td>
</tr>
<tr>
<td>N vs. Com</td>
<td>44.83</td>
<td>74.29</td>
<td>55.91</td>
<td>FCM_NOP_NABC</td>
<td>86.11</td>
<td>50.82</td>
<td>63.92</td>
</tr>
<tr>
<td>N vs. ABC</td>
<td>35.79</td>
<td>97.14</td>
<td>52.31</td>
<td>FCM_NOP_VCOM</td>
<td>58.33</td>
<td>70.00</td>
<td>63.64</td>
</tr>
<tr>
<td>E vs. Com</td>
<td>34.02</td>
<td>94.29</td>
<td>50.00</td>
<td>FCM_NOP_ECOM</td>
<td>67.65</td>
<td>54.76</td>
<td>60.53</td>
</tr>
<tr>
<td><strong>Group 2</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FCM_OP_EABC</td>
<td>96.36</td>
<td>92.98</td>
<td>94.64</td>
<td>kmeans_NOP_EABC</td>
<td>96.49</td>
<td>88.71</td>
<td>92.44</td>
</tr>
<tr>
<td>FCM_OP_NABC</td>
<td>83.02</td>
<td>57.89</td>
<td>68.22</td>
<td>kmeans_NOP_EV</td>
<td>49.15</td>
<td>72.50</td>
<td>58.59</td>
</tr>
<tr>
<td>FCM_OP_EV</td>
<td>72.22</td>
<td>60.00</td>
<td>65.55</td>
<td>kmeans_NOP_VCOM</td>
<td>60.78</td>
<td>50.82</td>
<td>55.36</td>
</tr>
<tr>
<td>FCM_OP_ECOM</td>
<td>84.21</td>
<td>57.14</td>
<td>68.09</td>
<td>kmeans_NOP_NABC</td>
<td>70.73</td>
<td>44.62</td>
<td>54.72</td>
</tr>
<tr>
<td>FCM_OP_VCOM</td>
<td>81.82</td>
<td>55.38</td>
<td>66.06</td>
<td>kmeans_NOP_ECOM</td>
<td>60.78</td>
<td>46.97</td>
<td>52.99</td>
</tr>
<tr>
<td><strong>Group 3</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>kmeans_OP_EABC</td>
<td>98.08</td>
<td>87.93</td>
<td>92.73</td>
<td>EM_FUZZY_S</td>
<td>95.38</td>
<td>98.41</td>
<td>96.88</td>
</tr>
<tr>
<td>kmeans_OP_EV</td>
<td>70.37</td>
<td>58.46</td>
<td>63.87</td>
<td>EM_FUZZY_C</td>
<td>89.23</td>
<td>100.00</td>
<td>94.31</td>
</tr>
<tr>
<td>kmeans_OP_VCOM</td>
<td>72.34</td>
<td>52.31</td>
<td>60.71</td>
<td>EM_FCM_C</td>
<td>96.36</td>
<td>89.83</td>
<td>92.98</td>
</tr>
<tr>
<td>kmeans_OP_NABC</td>
<td>78.05</td>
<td>49.23</td>
<td>60.38</td>
<td>EM_FCM_S</td>
<td>93.62</td>
<td>91.67</td>
<td>92.63</td>
</tr>
<tr>
<td>kmeans_OP_ECOM</td>
<td>74.29</td>
<td>50.00</td>
<td>59.77</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
5.2.2 STRENGTHS AND SHORTCOMINGS OF EACH METHOD

This section considers the advantages and disadvantages of each method in detail.

5.2.2.1 POWER-LAW REGRESSION METHOD

The advantages of this method include simplicity of implementation, easy of interpretation, easy generalisation to any other graph metrics, and low complexity of computing an anomaly score. However, its accuracy depends highly on the nature of the dataset its extracted features. For example, if the relationship between the extracted features scatters along the regression line and can segment the line into different areas, using only distance to the regression line can increase the false alarms.

5.2.2.2 ORTHOGONAL PROJECTION METHOD

The advantages of this method include increasing accuracy, independency of domain knowledge, simplicity, and overcoming the limitation of the distance-based method. This method can apply to all types of extracted metrics using any regression model including linear or power-law. However, its accuracy is influenced by an employed clustering algorithm and its limitations. For instance, if k-means is used it will face some drawbacks such as dependency on initialisation, and defining the number of clusters. Although a clustering algorithm can increase the accuracy, it increases the complexity of detecting anomalies because the extra step is introduced.
5.2.2.3 DISTRIBUTION METHOD

The advantages of bringing this probabilistic model along with fuzzy logic to the problem of anomaly detection can include accuracy improvement, a soft boundary between normal and anomalous instances, a one-time rule generation process, separating the model and representation of components, and unsupervised learning. It is very important for real-life applications that probabilistic modelling does not require classified data for training and that it can work in an unsupervised way. Application of the probabilistic model can achieve high performance especially in low dimension space because the distribution of instances can be better modelled with mixture Gaussian. Moreover, representing the generated component using the fuzzy membership function can make the results more interpretable than using only the model itself. Disadvantages of this method include complexity and slow convergence of EM, and inability to provide a suitable number of components that have minimum complexity. Therefore we proposed to use log-likelihood measure to overcome the number of components and complexity by reducing the iterations steps. Without any stopping threshold, EM would use several steps to optimise the cluster assignments. Moreover, distribution-based approaches are not suitable for high-dimensional datasets (Barnett & Lewis, 1984). It is hard to verify which model fits a given dataset without prior knowledge of the data distribution in different dimensions (Zhu, Kitagawa, Papadimitriou, & Faloutsos, 2011).
5.2.2.4 CLUSTERING METHOD

The advantages of the clustering method include analysing the effectiveness of using a clustering algorithm, better accuracy compared to distance based method, comparing Euclidian distance and fuzzy distance, domain knowledge and input independent, and simplicity. This method can apply to all type of inputs extracted from graph metrics. However, its accuracy is influenced by an employed clustering algorithm and its limitations. For instance, k-means, which is considered to be a hard clustering algorithm, will face some drawbacks, such as dependency on initialisation, being an exact member of a cluster, and defining the number of clusters. These characteristics mean that we cannot clearly know the probability of the observation instances being a part of different clusters. It reduces the effectiveness of hard clustering methods in many real situations. To overcome this issue a fuzzy clustering method such as FCM, which incorporates fuzzy set theory, is used. FCM is an extension of the classical and the crisp k-means clustering method in fuzzy set domain. The advantages of this algorithm include robustness for uncertainty and maintenance of much more information than any hard clustering methods.

5.2.3 PERFORMANCE COMPARISONS

This section is concerned with discussing the performances of each method. The detailed results from all methods, including the benchmark methods are shown and compared. The main focus of this section is on relating the challenges to the results, analysing the top performance of each group, and discussing the
reasons behind each method’s outperformance. This discussion is based on Table 8 to Table 10. The proposed and benchmark methods are compared in terms of the precision, recall, and F-Score measures. Either precision or recall concentrates on only one part of performance. The F-Score is used to compare all the methods using both precision and recall.

5.2.3.1 DEALING WITH ANOMALY DETECTION CHALLENGES

To detect anomaly this research faced different challenges. One of these challenges was to define reasonable boundaries between normal and anomalous instances, especially for those which are sitting close to the boundary. This problem reflects on the comparison between the experiment of distance-based approaches with lower F-Score value and the others. Using fuzzy logic and different machine learning techniques improves up to thirty percent accuracy of detecting anomalies with regard to the OddBall method.

The other challenge is defining an anomaly which is application dependent and which varies based on domains. As a result, anomaly in OSNs needs to be defined by either finding patterns followed by minority in the dataset or using other researcher’s definition. Both these methods are used in this research. These challenges limited this research to solve a particular formulation of the anomaly detection problems: starness and cliqueness. To solve these problems we needed to find a suitable approach. As a result we developed and customised the existing techniques from statistics, machine learning, and data mining. Different techniques are used depending on the nature of the input data (e.g. discrete,
continuous), the type of anomaly, and the availability of labelled data. This thesis mitigates these challenges by using hybrid approaches.

5.2.3.2 DISTANCE-BASED METHOD VS CLUSTERING-BASED METHOD

In terms of recall and precision, group 1, the E vs. ABC method, outperforms all of the other methods. This method performs better than others as it used both the number of edges and betweenness centrality to find the anomaly. Using these metrics together imposes the strict conditions of using the shortest path and dense subnetworks. Moreover, the average betweenness centrality focuses on the behaviour of a central node, in terms of centrality and the flow of information. Although the F-Score of E vs. ABC is not high compared to all the other methods, it is higher than the other methods in the first groups, with an average of twenty-nine percent among all three datasets.

The recall performance of E vs. Com has the second higher rank in this group but its precision performance is one of the lowest ranks. By using community information most of the anomalies can be detected with a high rate of false positive which effects precision performance. The reason behind the low precision is through the lack of required information of communities, such as the availability of all connections in 2-level neighbourhoods of a node under investigation in the datasets. Using average betweenness centrality and the number of nodes together does not show a good performance in terms of recall and precision. The reason is that the number of nodes has the minimum effect on a user network topology. However, the number of nodes can affect the formation of the community in OSNs. This is confirmed by the result of the N vs. Com method,
which shows a better F-Score compared to E vs. Com method. Using only the number of nodes and edges (N vs. E) is not sufficient to achieve a high success rate; however, better accuracy can be achieved when combining other metrics such as average betweenness centrality with them. Using orthogonal projection and a clustering algorithm together can improve the accuracy, as shown in groups 2 and 3 (Table 8 to Table 10).

The accuracy of each method is varied, and influenced by the metrics (e.g. E, N, ABC) they use as inputs. In general using clustering algorithms regardless of their techniques can enhance the accuracy of detecting anomaly. This is confirmed by the experiment results shown in groups 4 and 5 (Table 8 to Table 10). However, a fuzzy-based clustering algorithm such as FCM shows better result compares to hard partitioning ones, for example k-means. The reason for using k-means as a benchmark is that FCM can be considered an extension of the classical and the crisp k-means clustering method in the fuzzy set domain. The results of applying the clustering algorithms are consistence among the three different real-life datasets. The proposed fuzzy-based clustering method allows instances to be members of more than one cluster, with different levels of certainty, in contrast to hard partitioning algorithms such as k-means, in which any instances can only belong to a single cluster. This means that the fuzzy nature of friendship relations is lost during clustering. It affects the quality of detecting anomalies within the online social network data.
5.2.3.3 DISTRIBUTION-BASE METHOD VS CLUSTERING-BASED METHOD

Among the six groups of methods, EM_FUZZY_S method achieves the best F-score, followed by EM_FUZZY_C. In general, distance-based methods generate the lowest F-scores. The reasons behind the lower accuracy of the distance-based methods include: applying no machine learning algorithms or fuzzy logic on the extracted graph’s features. It uses only the distance to regression model, which brings some limitations such as inability to detect normal instances sitting far from the regression line.

The results of the combination of EM and fuzzy logic presented in group 6 are also consistent with the assumption related to the hypothesis that using fuzzy approaches can increase the accuracy of detecting anomalies in OSNs. These distribution-based methods show a higher F-Score than the other methods in all three datasets. This consistency is highly significant in terms of proving that the proposed methods can overcome the accuracy problem and introducing new formalisation of anomaly detection in OSNs area. The main reasons behind this improvement include relaxation of the sharp boundary between normal and anomalous instances, unsupervised learning, and disjointing of the model to represent the components using fuzzy logic. The use of fuzzy logic and the mixture model to learn distribution parameters and to model the overlapping between two components plays a key role. Fuzzy logic when used to define the boundary of anomalies contributes more than EM in terms of improving the accuracy.
5.2.3.4 PROPOSED METHODS VS BENCHMARKING

All of the proposed methods are better performed, in terms of F-score, than the existing benchmarking methods, including OddBall and k-means methods. The OddBall method uses only distance between the observed value and the predicted value, generated by a power-law regression model. However, it does not consider the density and fuzziness of instances. Although k-means considers the density of instance, it fails to overcome the overlapping between two clusters. As the results indicate, using these two factors is the main cause of the higher F-Score for the proposed methods. However, the recall of the proposed methods is worse than that of the benchmarking methods in some instances. There are two reasons why proposed methods achieve lower recall than OddBall. First, the proposed methods use 2-level-neighbourhoods to detect community information in order to detect anomalies, while the OddBall method uses 1-level-neighbourhoods information. Some users just join OSNs, so 2-level information is not available. Second, the OddBall method computes anomaly scores based on the distance from the regression line; and the density of instances sits close to the regression line.

5.2.3.5 EFFECTIVENESS OF CLUSTERING AND ORTHOGONAL PROJECTION

In this section, comparisons are conducted to observe the effect of using the orthogonal projection and clustering algorithms. The observations show that using clustering algorithms applied onto the orthogonal projection of instances improves the accuracy in all three datasets. The reason behind this improvement is related to this use of orthogonal projection. Using only distance to fitting line or
only a clustering algorithm can be prone to missing some anomalies or normal instances. The combination of these two methods of orthogonal projection and clustering can overcome the existing limitation by reducing the dimensions and grouping them accordingly. New dimensions introduce new transformed distances on which the clustering algorithm is carried out. Using this hybrid method we can find hidden similarities in an accurate way, in order to detect any abstract patterns.

5.2.3.6  TOP PERFORMANCE COMPARISONS

In Figure 44, Figure 45, and Figure 46 we compare the top performance of each group, revealing facts about the characteristics of our proposed method that we will discuss in this section. The first observation shows that higher performance in all three datasets belongs to the EM_FUZZY methods. This is because of the ability to find the natural relationship between instances with EM (Expected-Maximization) and to describe the fuzziness of the instances with fuzzy logic. EM, an unsupervised learning algorithm, is employed for data clustering based on unobserved latent variables which can be the cluster number. One underlying power of EM is to assume there is a joint distribution between each instance and the latent variable that place them together.

The other observation shows that using fuzzy clustering algorithms applied onto the orthogonal projection of instances improves the accuracy in all three datasets. The reason behind this improvement is related to the use of orthogonal projection, fuzzy clustering algorithms, and fuzzy linguistic representation of each generated cluster.
Because all outliers are not anomalies; sometimes the normal instances also sit far from the regression model, based on metrics’ characteristics used to make this model. Thus the instances that fall away from the curve with negative correlation cannot definitely be considered as anomalies especially if the instances scatter along the regression line. By using the orthogonal projection we will deal with density problem instead of distance from regression model. Therefore to spot anomalies we can apply any density-based approaches such as clustering on projected instances.

However, fuzzy clustering show better results as we deal with the fuzzy nature variables such as friendship. The fuzzy clustering algorithms generate a matrix in which the membership degree of each instances to the all generated clusters is denoted. This matrix can be represented with fuzzy membership functions in order to map each cluster into a linguistic variable which help to describe the degree of membership of an anomaly score in a fuzzy set. This can improve the accuracy of anomaly detection by relaxing the sharp definition of anomalies using linguistic variable.

In general, fuzzy-based methods show better accuracy than other methods. For instance, the EM_FUZZY method shows the highest accuracy, with an average of thirty percent improvement compared to the OddBall method across all three datasets. The most important factors for this improvement are (1) having a soft boundary between normal and anomalous instances using fuzzy logic, (2) separating the model and representation of components by using membership functions, and (3) generating rules using the fuzzy inference engine. The benefit of employing the fuzzy inference engine includes generating rules using input and
output membership functions, reducing the complexity of the methods and increasing the scalability. The reason behind the low complexity and scalability of fuzzy-based methods is that the rules are generated offline, based on training datasets, and this process needs to be conducted only once. Any new rules, if needed to be generated based on new training datasets, will be added to the current rules using the inference engine.

Figure 44. Facebook Top Performance

Figure 45. Flickr Top Performance
Chapter 6: Experiments and Discussions

5.3 SUMMARY

This Chapter has discussed the experimental results of the proposed methods. Performances of all the proposed methods are gathered and compared in order to analyse the impact of various metrics with the aim of providing a better understanding of the quality of the proposed approach by applying it to three different real-life datasets. The reasons behind these improvements can also be discussed, along with analyses of how the proposed methods are suitable to deal with online social networks data. Experimental investigation of the proposed algorithms for anomaly detection problem is carried out on different test-beds of instances. The strengths and limitations of the proposed methods, as well as their performances, were discussed. The higher F-Score of fuzzy based methods lie in the powerful fuzzy logic concept, which can overcome the limitation of the multi-level logic problem in which a degree of truth rather than only two possible values (normal or abnormal) is used. Comparisons are conducted to analyse the
performance of the proposed and benchmarking methods. The proposed methods outperform the OddBall method because they use different techniques in a hybrid way instead of using only the distance to a model as a measure of detecting anomalies.

The experiments applied to three different datasets reveal the facts about the characteristics of our proposed method. In general, using a clustering algorithm, regardless of which clustering techniques have been used, can increase the accuracy of detecting the anomaly. However, fuzzy based clustering show better result compares to hard portioning ones. The results are consistence among the three different real-life datasets.

Different combinations of techniques are used based on the type of the input data used by each method. By using the hybrid method we can find hidden similarities in an accurate way in order to detect any abstract patterns. In view of the limitations of the existing methods in anomalies detection, the best strategy is to combine detection techniques in a hybrid approach with multi-level logic such as fuzzy-logic to achieve better performance. For instance, the combination of EM and fuzzy logic can improve the F-Score by reducing the false positive and negative rate. This method outperforms the accuracy among all groups of methods. This is consistent with the hypothesis that using a fuzzy hybrid method can increase accuracy. The EM_FUZZY method shows higher performance in all three datasets, because of its ability to find the natural relationship between instances with expectation-maximization algorithm and to describe the fuzziness of the instances with fuzzy logic. The high performance of the proposed methods is related to the use of fuzzy logic in which we can define the degree of fuzziness.
and can use a domain expert’s intervention. Intuitively, the more specific the rules are, the higher precision we can obtain.
Chapter 6: Conclusions

In recent years, the number of users of Online Social Networks (OSNs) such as Facebook and Twitter has been growing rapidly. These online social networks allow users to create a profile including their personal information, to add other users as friends, and to exchange messages. Such networks also allow users to join various communities according to their common interests or groups organised by workplace, school, or college. This emerging technology, which is being used for various purposes such as business, education, telemarketing, medical, and entertainment, also opens the door for unlawful activities. Analysing user behaviour to identify anomalies in this new perspective of social life which articulates and reflects the off-line relationships is also becoming necessary. This rising need is based on two assumptions: (1) the user behaviour patterns can carry useful information for the social network analysers; and (2) these patterns can be linked to unlawful activities such as cyber-attacks.

Existing anomaly detection algorithms which are applied to online social networks are limited because of issues such as complexity, low accuracy, privacy, lack of labelled datasets and lack of sufficient information. They mostly focus on the dynamic usage behaviour of users and the structure of the user network graph using statistical approaches. However, these algorithms have no consideration of the fuzzy nature of behaviours in online social networks when they model the behaviours. In order to tackle these new problems, this thesis has proposed hybrid fuzzy structure-based anomaly detection methods, as the structural data cannot be
denied by users. Moreover, we have analysed the effectiveness of using different metrics and modelling to find the best parameters for our models in order to improve accuracy. We used graph theory for modelling a user’s local network and extracting features that can help find anomalies. In terms of accuracy, a combination of modelling using graph theory, distribution-based clustering, and fuzzy logic shows the best result among the approaches.

6.1 RESEARCH CONTRIBUTIONS

Based on the literature review, methods for finding outliers in a large data graph can be divided into density-based, distance-based, distribution-based, depth-based and clustering based techniques. However, they are not specifically customised for online social networks purposes in which we are dealing with human behaviours as key inputs. In general the research gaps from the literature review are identified as:

- Lack of accurate model to understand and model the topological behaviours in online social networks environment. The model should be able to distinguish between different degrees of anomalies. Most of the times, the line between anomalies and legitimate user is blurred which make it difficult to detect.

- Dependence on technology, in terms of how users’ usage patterns are mined;

- Lack of support for the multi-level logic problem in which a degree of truth rather than only two possible values (normal or abnormal) is used.
These shortcomings are overcome by:

- Employing hybrid methods using graph metrics, a power-law regression model, orthogonal projection, and clustering algorithms;
- Employing structural meta-data which are not deniable by users as inputs to the proposed algorithms;
- Employing fuzzy logic to deal with anomalies by treating them as a multiple-valued logic problem.

The main contributions are summarised below:

- Conducted an analysis of online social networks data graph in terms of degree distribution, and features selection. As a consequence, the properties and features of the online social network are discovered and applied to the proposed methods.
- Developed distance-based anomaly detection methods.
  - Used orthogonal projection to get the benefit of the density of instances in order to detect anomalies more accurately, unlike the existing distance-based methods which focus on distance from the power-law regression model;
  - Adapted distance-based methods using orthogonal projection and a clustering algorithm to solve the problem of normal instances which sometimes sit a long away from power-law regression model;
  - Introduced new graph metrics and definitions such as average betweenness centrality, and community cohesiveness;
• Designed a framework to evaluate the proposed approaches.

• Developed Distribution-based anomalies detection methods.
  
  o Focused the methods on the distribution of anomalous instances within natural clusters generated by a EM-Gaussian Mixture Model algorithm;
  
  o Used graph theory to model users’ relationships inherent in online-social-network;
  
  o Introduced two anomaly scores for starness and cliqueness;
  
  o Used a combination of Gaussian Mixture Model and fuzzy logic as a novel method to differentiate between normal and anomalous instances;
  
  o Used fuzzy linguistic and quantitative variables to symbolise uncertainty such as friendship relations in online social networks.

• Developed Clustering-based anomalies detection methods.
  
  o Defined the number of clusters automatically using the natural underlying relationship between instances.
  
  o Used a fuzzy membership function to define the boundary of anomalies.
6.2 MAIN FINDINGS

The main findings in this research are summarised and explained as follows:

- Analysing the structure and features of an Online Social Network helped to understand how the anomalies should be detected.
  - The analysis of degree distribution of users in given real-life datasets shows the existence of power-law distribution, meaning that the majority of users have few friends and the minority have a large number of friends;
  - On average, 91% of users’ local network topology follows the “friend of friend is often friend” patterns.
  - The quantitative features of online social networks, such as friendship relation topology, community cohesiveness, kinship, loose and tied friendship, and activities, should be seen as a fuzzy variable (multiple-valued logic) to improve the accuracy of anomaly detection. This is shown in the experiment results in which using fuzzy logic can have up to a thirty percent average improvements.

- Distance-based methods findings are as follows:
  - In general, using orthogonal projection and a clustering algorithm shows better performance in terms of accuracy in detecting anomalies compared to using only the distance from regression model. By using orthogonal projection we are able to overcome the
problem of normal instances sitting far away from the fitting line in the distance based approach.

- Using orthogonal projection of the relationship between average betweenness centrality (ABC) and the number of edges (E) on a power-law regression model shows better performance as the average betweenness centrality focuses on behaviour of a central node, in terms of influence and the flow of information.

- The FCM_OP_EABC method, compared to the baseline method, shows higher recall and precision. The reason behind the better performance is the converting of distance to density by using orthogonal projection. This helps to overcome the problem of a distance-based approach where some normal instances sit far from the regression model.

- The power-law regression model shows better modelling performance to explain topological behaviour of a user network, according to the $R^2$ measure. This is because of the scale-free nature of online social networks which are a class of power-law networks. In the power-law networks the high-degree nodes tend to be linked to the other high degree nodes.

- Clustering-based methods findings are as follows:

  - Using adapted fuzzy c-means in which we use the log likelihood to find the natural number of clusters shows better performance compared to using a hard clustering such as k-means. Having a
membership degree enables us to deal with the fuzzy nature of instances (e.g. online social networks characteristics) in the given datasets. For instance, friendship relationship, community cohesiveness, or influence in online social networks can be considered as fuzzy characteristics.

- The higher performance is because fuzzy clustering instances can belong to more than one cluster, with different membership degrees ranging from 0 (not belongs) to 1 (fully belongs). The membership degree or cluster fitting likelihood is computed as a key part of a fuzzy clustering algorithm that fits instances to the proper clusters.

- Distribution-based methods findings are as follows:
  - The proposed method, EM_FUZZY, which is based on a statistical distribution and fuzzy logic, shows higher accuracy compared to the other methods including FCM_OP/NOP, kmeans_OP/NOP and EvsABC. This is because of using fuzzy logic and the mixture model to learn distribution parameters and to model the data.
  - The inclusion of a learning method (EM) into EM_FUZZY enhances the performance of the results.
  - Fuzzy logic used to define the boundary of anomalies contributes more than EM in terms of improving the accuracy.

- Results comparisons are conducted on all the proposed methods and benchmarking methods. All of the proposed methods perform better than the benchmarking methods, including OddBall, EvsABC, NvsCom,
and NvsABC, in terms of accuracy. The main reason behind this improvement is the use of fuzzy logic in a hybrid manner with the distribution model.

- Graph theory modelling is the basis for all of the proposed methods. The methods in which fuzzy logic is employed outperform the other methods. Moreover utilising the orthogonal projection and clustering algorithms together can increase the accuracy beyond that achieved when used individually. For instance, FCM_OP_EABC outperforms EvsABC in detecting anomalies more accurately.

- Fuzzy logic enables us to use and convert linguistic language to a crisp value. For instance, we are able to measure the degree of similarity of a user topology to a star or a clique.

- Reliability of user information is solved by using meta-data information such as connections established with other users by users.
6.3 ANSWERS TO RESEARCH QUESTIONS

This section relates the findings of this research to the research questions.

- **What are the new features to select from the graph modelling of the online social network data in order to represent anomalies and to get better insight to discover anomalies?**

Sometime the attributes used for building a model may not construct meaningful input and can result in an inaccurate model. For instance, to detect community between different users considering only the number of nodes in the users’ networks without the links (external edges) between them leads to have an inaccurate model. This is because of the definition of community detection which is more effected by the edges instead of nodes. The other example is to use profile information of a user, which can be fabricated, to build a model to detect structural connectivity. Social network components include users, friendship ties, communities, and the information passing through. Some important attributes that can be modelled using graph theory in order to detect anomalies include user’s network information (e.g. number of nodes and edges, betweenness centrality), structural connectivity such as presence of star and clique sub-networks (topology), community cohesiveness, popularity, and isolated users. In this thesis, we analyse and model online social network characteristics using a combination of graph and statistical theory. Our statistical proposition is based on the fact that the majority of users in online social networks follow the “friend of friend often friend” topology pattern and minority of users follow star and clique topology. We extract new and meaningful structural features using graph theory for detecting starness, cliqueness, and “friend of friend often friend” topologies.
These new features from graph modelling, which are the best to represent anomalies and to get better insight to discover anomalies, include: betweenness centrality, average betweenness centrality, number of nodes and edges, community cohesiveness, starness degree, and cliqueness degree.

- How can fuzzy-based machine learning techniques are developed to detect anomalies and increase accuracy using the selected features of online social networks?

An object is, naturally, an anomaly if it differs significantly from the other instances around it. Based on definitions of “differs from the other instances”, and whether or not the variation is considerable, there will be different sets of anomaly definitions. Different anomaly definitions need to be considered differently. Some well-known techniques used to deal with these differences are based on distance, distribution, and clustering concept. The main advantages of using these concepts are to analyse the suitability of approaches in characterising online social network. These methods aim to tackle the anomaly detection problem from different angles in order to reach a better accuracy and simplicity by using the extracted features. In the first method our focus is on differences between the observed value and the predicted value generated by a power-law regression model as well as by orthogonal projection. This method uses the power-law relationships between the extracted features. The second method, which uses the starness and cliqueness features, focuses on distribution of the topological behaviours of instances. This distribution is utilised to discover common patterns that can lead to the detection of anomalies. In this approach, we use the Gaussian mixture model and fuzzy
logic. In the third method, fuzzy clustering and fuzzy membership functions are employed to cluster the extracted features, and to deal with the sharp boundary between the generated clusters.

- How a multi-layered framework should be used for analysing and evaluating the proposed methods using unlabelled datasets with semi-supervised learning approaches?

This research uses a multi-layered framework to evaluate the proposed methods. To evaluate, we use the datasets which are generated from the selected features of given online social networks. The multi-layered framework provides the full requirements needed for detecting anomalies in the online social networks graph data which include pre-processing, data modelling, features selection and extraction, developing methods, data labelling, and evaluation. Data collection processes are usually faced with out of control issues which result in having noisy data such as missing values, wrong data combinations, and duplicate data. Therefore, the first layer of the framework concerns pre-processing which includes normalisation, cleaning, transformation, integration, reduction, feature extraction and selection. The appropriate features are then extracted for each node in the graph. These features are used as input data for all the methods developed in this research. The second layer of the framework introduces various methods that handle the problem of anomaly detection in different ways. These methods tackle the anomaly detection problem from different angles in order to achieve better accuracy and simplicity. The third layer of our framework is allocated to labelling a subset of datasets and evaluation. Due to a lack of labelled data, a semi-supervised approach using human experts who manually prepare the labelled training data set is employed. For the labelling, visualisation software such as R
and NodeXL are used to view random user networks, to decide whether they are an anomaly or not. The ground-truth for this labelling is based on the “friends of friends are often friends” pattern which is followed by the majority. The variations of the minor patterns, such as star and clique, can lead to anomalies. Using the set of labelled data we can evaluate the proposed methods as well as the benchmark methods.

6.4 FUTURE WORK

A number of open problems must be solved to allow the development of an automatic fuzzy inference system to generate rules for detecting anomaly more accurately. These problems propose different research directions that need to be pursued to make such a system feasible. One of the directions would be to study the effect of allowing automatic learning of the rules from the probabilistic model. The proposed framework requires that the model be built sequentially with generating rules. It would be better to allow the rules to be adapted and extended from the built model during the training model to best fit the data. The Bayesian model selection is suggested to use to find the best model for a given dataset. Another direction would be to create hybrid fuzzy inference systems from different models generated from different types of data which allows rules to be made more general. Finally, in terms of applications, the framework can be applied beyond online social networks to any area that needs to find anomalies. My specific interest is to use the framework to learn more complex models of human behaviours, as a means of performing more general anomaly detection on the Internet.
Bibliography


the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, Niagara, Ontario, Canada.


Eberle, W., & Holder, L. (2007). Discovering Structural Anomalies in Graph-Based Data. In Proceedings of the 7th IEEE International Conference on Data Mining Workshops (pp. 393-398).


Joshi, M. V., Agarwal, R. C., & Kumar, V. (2002). Predicting Rare Classes: Can Boosting Make Any Weak Learner Strong? In Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 297-306). ACM.


