|  |  |  |  |
| --- | --- | --- | --- |
| LOW LEVEL DESIGN AND IMPLEMENTATION DOCUMENT  COMMUNITY DETECTION IN DYNAMIC NETWORKS  UE18CS390B – Capstone Project Phase – 2  ***Submitted by:***   |  |  | | --- | --- | | **Mahammad Thufail**  **Manna Vasanth**  **Purushotham S**  **Pulle Manikya Sri Manasa** | **PES2201800646**  **PES2201800425**  **PES2201800480**  **PES2201800468** |   Under the guidance of   |  | | --- | | **Prof. Sreenath MV**  Designation  PES University |   **August - December 2021**  **DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**  FACULTY OF ENGINEERING  **PES UNIVERSITY**  (Established under Karnataka Act No. 16 of 2013)  Electronic City, Bengaluru – 560 100, Karnataka, India |

TABLE OF CONTENTS

|  |  |
| --- | --- |
| 1. Introduction |  |
| 1.1 Overview | 3 |
| 1.2 Purpose | 3 |
| 1.3 Scope | 4 |
| 1. Proposed Methodology / Approach |  |
| 2.1 Designing the terrorist groups network | 4 |
| 2.2 Finding influential nodes | 4 |
| 2.3 Community Detection: | 5 |
| 2.3.1 Louvain algorithm using locality modularity optimization | 5 |
| 2.3.2 Adding the edges/nodes | 6 |
| 2.3.3 Removing the edges/nodes | 8 |
| 2.4 Algorithm and Pseudocode | 10 |
| 2.5 Implementation and Results | 10 |
| Appendix A: Definitions, Acronyms and Abbreviations | 11 |
| Appendix B: References | 1 |
| Appendix C: Record of Change History | 3 |
| Appendix D:Traceability Matrix | 13 |

# Note:

|  |  |
| --- | --- |
| **Section 1** | **Common for Prototype/Product Based and Research Projects** |
| **Section 2 & 3** | **Applicable for Prototype / Product Based Projects.** |
| **Section 4** | **Applicable for Research Projects.** |
| **Appendix** | **Provide details appropriately** |

# Introduction

# 1.1 Overview

Terrorism is an organised type of violence that has a direct impact on stability, a country's or community's daily routine, and security, as well as a means of instilling fear in civilians. Terrorism is a fluid phenomenon, so equipping counter-terrorism operators with the resources they need to combat it is important.

We used a tool that helps us to find terror organisations with common operational features in clusters. Specifically, we used open access data from terrorist activities worldwide since 1970 to create a network of terrorist organisations and related information on tactics, weapons, goals, and active areas.

Each partition is linked to the terrorist groups in our model. Later, we'll try to avoid attacks by identifying the most powerful party with the greatest number of connections to other networks. Community identification is a technique for identifying groups of nodes in which the connections between nodes within a group are greater than the connections between nodes in other networks.

**1.2 Purpose**

The key aim of this study is to find a mechanism for eliciting information about perpetrators in terrorist incidents by looking at terror attacks over time. The aim is to build a sociogram, or criminal network, with nodes representing terrorist organisations and edges representing generic connections between two groups.

To understand the composition and evolution of terrorist networks, we will find influential nodes based on centrality measures. Finally for Community Detection we use Louvain algorithm using locality modularity optimization.

**1.3 Scope**

We are using an approach that will allow us to find clusters of similar terror groups using information on their operational characteristics. Specifically, using open access data of terrorist attacks occurred worldwide since 1970, we are trying to build network that includes terrorist groups and related information on tactics, weapons, targets, active regions.

We model this data with each partition joined to the terrorist groups. Later on we will find the most influential group with maximum number of relations between other networks and try to prevent the attacks .Community detection to identify sets of nodes in such a way that the connections of nodes within a set are more than their connection to other network nodes.

# Proposed Methodology / Approach

**2.1 Designing the terrorist groups network**

The final move is to use the similarity matrix to construct the terrorist group's network while still considered a thresholding operation. The network-building algorithm is fairly straightforward. Assume W = (V, E), with W representing the terrorist group's network, V representing the network's nodes, and E = V X V representing the network's borders.

**2.2 Finding influential nodes**

Degree centrality is defined as the number of links incident upon a node (i.e., the number of ties that a node has). If the network is directed (meaning that ties have direction), then two separate measures of degree centrality are defined, namely, indegree and outdegree.

An edge with a high edge betweenness centrality score represents a bridge-like connector between two parts of a network, and the removal of which may affect the communication between many pairs of nodes through the shortest paths between them.

Eigenvector centrality is used to measure the level of influence of a node within a network. Each node within the network will be given a score or value: the higher the score the greater the level of influence within the network. This score is relative to the number of connections a node will have to other nodes. Connections to high-scoring eigenvector centrality nodes contribute more to the score of the node than equal connections to low-scoring nodes.

**2.3 Community Detection:**

**2.3.1 Louvain algorithm using locality modularity optimization:**

To detect groups, we use the Louvain algorithm with local modularity optimization. This algorithm uses a greedy optimization approach to increase the modularity of a network partition iteratively.

Modularity, a metric of network structure, is used to determine the strength of division of modules. In networks with high modularity, connections between nodes are dense, but connections between nodes in various modules are sparse. It's mostly used in network community structure detection optimization methods.

The goal function is maximised in each iteration to calculate the populations. In step 1, small groups have been formed by maximising the modularity on a area level. Only local infrastructure upgrades are allowed at this time. In the following step, nodes that belonging to the same entity are combined into a unique node that is going to point out a community in a original aggregating network of gangs. Through the construction of a hierarchy of units, steps will get repeated iteratively until no further changes in modularity are legally possible.

The groups must be re-computation from the beginning if the original algorithm avoids the insertion or elimination of newly originated edges and nodes right after getting the correct structure of community.

**2.3.2 Adding the edges/nodes:**

Adding edges/nodes results in four types of effects at the community structure level in this method.

**Cross-community edge:**

When we try to join two nodes in a Cross-community edge that are already linked to other nodes, two things can happen. The community structure remains unchanged if the linking nodes belong to the same community. If the linking nodes belong to separate communities, the two communities are merged into one.

**Inner community edge:**

If the two nodes on the edge already exist and belong to the same party, inserting a new edge between them increases the community's internal links thus leaving the intercommunity connections unchanged. As a consequence, the organisational structure of the company remains intact.

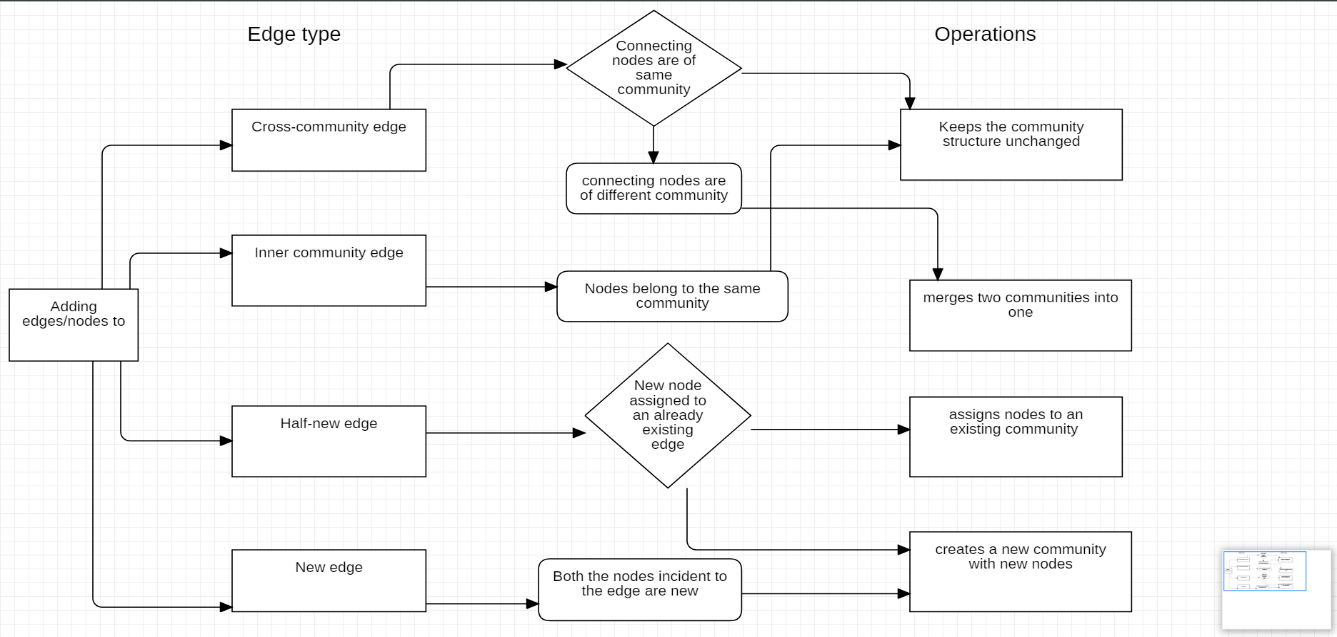


Fig.2 Adding edges/nodes

**Half-new edge:**

In this type of edge, the elaborating edge is a half new edge, which means one nodes in the structure is present in the network and another edge is newly originated. When a node is assigned to an existing community, the community configuration remains unchanged. Otherwise, a new group is generated with new nodes.

**New edge:**

In the new edge, all nodes adjacent to the edge are new. Here, there may happen two kind of things, that is nodes are assigned to the new community which is just generated or it may get assigned to two different communities where each one is for the each newly generated node.

**2.3.3 Removing the edges/nodes:**

At the group structure stage, removing edges/nodes results in four types of effects:

**Cross-community edge:**

In a cross-community edge, two different nodes adjacent to a excluded edge assigned to varying communities. The inner links of the group are maintained while intercommunity relations are minimised by eliminating these types of edges. This operation would not result in any existing communities being merged, nor does it disband any communities in which the excluded edge is a part.

**Inner community edge:**

The two nodes in the inner-community edge that are nearby to a edge is assigned to the original same group. Through removing these types, the community's inner links are reduced while intercommunity bonds are maintained. The disbanding has very little issues to the whole networks if the nodes belonging to the excluded edge are connected to other edges; otherwise, the community is split into smaller categories and branches that join other established communities.

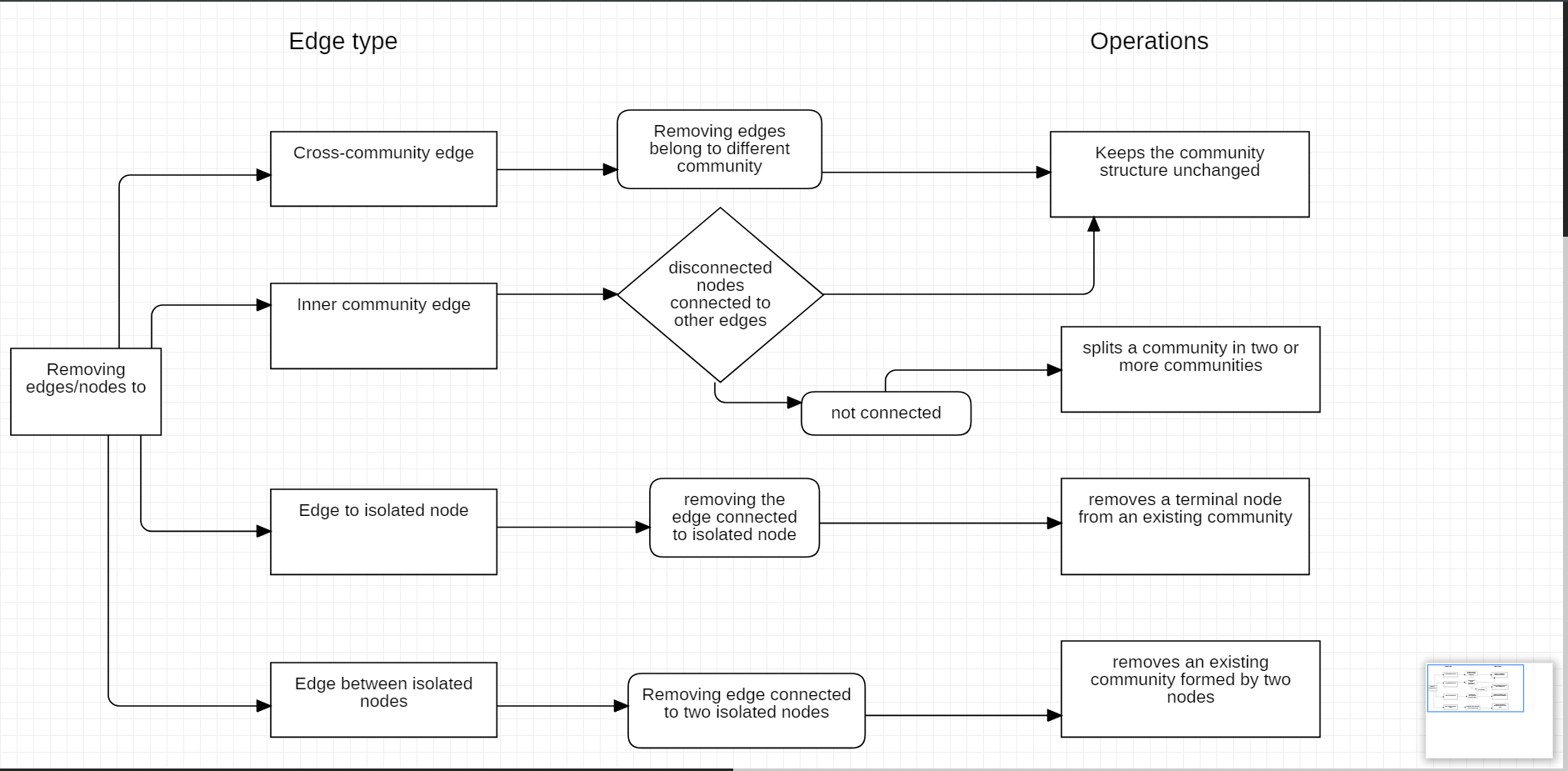


Fig.3 Removing edges/nodes

**Edge to isolated node:**

An isolated node is one of the nodes incident to the boundary, so disorienting these type of isolated edges leads to disorientation of the complete node itself. Since the deleted node is a node having single value, it has no impact on the group system and therefore on the community's inner relations.

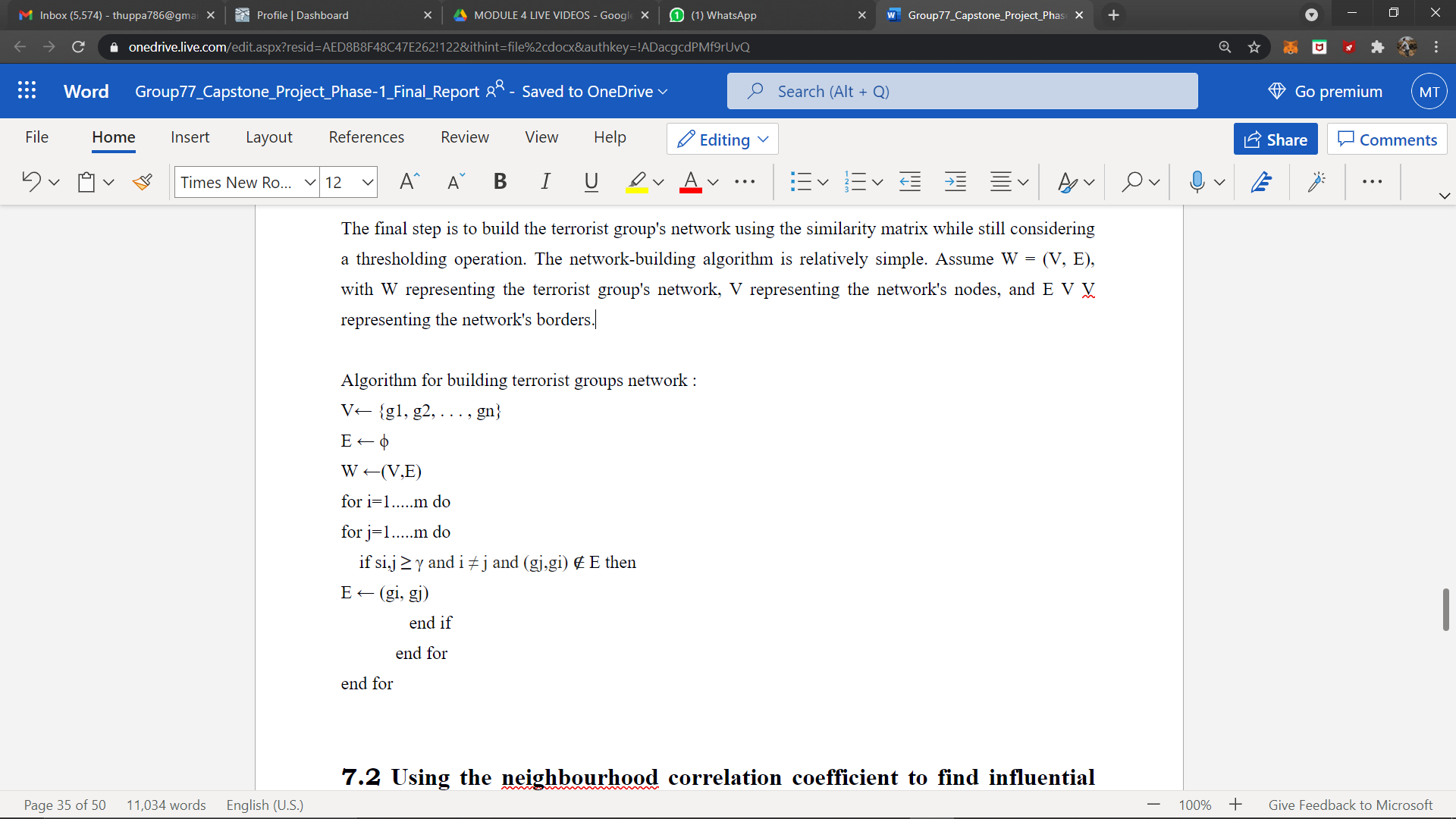
**Edge between isolated nodes:**

The edge in Edge of isolated nodes that needs to be disoriented from the community is assigned to two different terminal nodes. By eliminating the edge, all nodes are removed as well, essentially ending the group or groups of which they belong. The remainder of the neighbourhood will be unaffected.

Two of the eight operations arising from the disorientation/assignment of nodes/edges reduce the number of communities, two increase the count of communities, and four keep the population structure unchanged.

**2.4 Algorithm and Pseudocode**

Algorithm for building terrorist groups network:



**2.5 Implementation and Results**

To understand the composition and evolution of terrorist networks we built the network using network , then we will find influential nodes based on centrality measures.

Degree centrality is defined as the number of links incident upon a node  which finds the important node in the network. An edge with a high edge betweenness centrality score represents a bridge-like connector between two parts of a network which calculates the best connector in the network. Eigenvector centrality is used to measure the level of influence of a node within a network which gives the most influential node I the network. Finally for Community Detection we use Louvain algorithm using locality modularity optimization and detect communities in the network.

# Appendix A: Definitions, Acronyms and Abbreviations

GTD - Global Terrorism Database

# Appendix B: References

1. Vinceno Loia, Francesco, “Understanding the composition and evolution of terrorist group networks: A rough set approach.” Paper 2019, <https://www.sciencedirect.com/science/article/pii/>[S0167739X19307757](https://www.sciencedirect.com/science/article/pii/S0167739X19307757)
2. Ahmad Zareie, Amir Sheikhahmadi, Mahdi Jalili, Mohammad Sajjad Khaksar Fasaei, “Finding influential nodes in social networks based on neighborhood correlation coefficient.” Paper 2020, <https://www.sciencedirect.com/science/article/pii/>[S0950705120300630](https://www.sciencedirect.com/science/article/pii/S0950705120300630)
3. Mehdi Azaouzi, Delel Rhouma, Lotf Ben Romdhane, “Community detection in largescale social networks: stateoftheart and future directions.” Paper 2019,

<https://www.sciencedirect.com/science/article/pii/>[S0020025517310101](https://www.sciencedirect.com/science/article/pii/S0020025517310101)

1. Kun Hea, Yingru Li a, Sucheta Soundarajanc, John E. Hopcroft , “Hidden community detection in social networks.” Paper 2019,

<https://link.springer.com/article/10.1007/>[s12652-020-01760-2](https://link.springer.com/article/10.1007/s12652-020-01760-2)

1. Hamid Ahmadi Beni, Asgarali Bouyer, “TI-SC: top-k infuential nodes selection based on community detection and scoring criteria in social networks.” Paper 2020,

<https://www.researchgate.net/publication/333197917>

1. Aftab Farooq,Muhammad Uzair,Gulraiz Javaid Joyia,Usman Akram , “Detection of Influential Nodes Using Social Networks Analysis Based On Network Metrics” Paper 2018,

<https://ieeexplore.ieee.org/abstract/document/8346372>

1. Hong-Jian Yin, Hai Yu, Yu-Li Zhao, Zhi-Liang Zhu, Wei Zhang, “Analysis of the Dynamic Influence of Social Network Nodes.” Paper 2019

<https://www.hindawi.com/journals/sp/2017/5046905/>

1. Nesrine Hafiene,Wafa Karoui1, and Lotfi Ben Romdhane, “Influential Nodes Detection in Dynamic Social Networks”. Paper  2019

<https://link.springer.com/chapter/10.1007%2F978-3-030-20482-2_6>

1. Mohamed EL-Moussaoui, Tarik Agouti, Abdessadek Tikniouine, Mohamed Eladnani , “Community detection: Approaches and applications.” Paper 2019,

<https://www.sciencedirect.com/science/article/pii/S1877050919305046>

1. Punam Bedi and Chhavi Sharma, “ Community detection in social networks ". Paper 2016,

<https://www.researchgate.net/publication/295395520_Community_detection_in_social_networks>

1. Vinicius da Fonseca Vieira, Carolina Ribeiro Xavier, Nelson Francisco Favilla Ebecken and Alexandre Goncalves Evsukoff , “Performance Evaluation of Modularity Based Community Detection Algorithms in Large Scale Networks.” Paper 2014,

<https://www.hindawi.com/journals/mpe/2014/502809/>Appendix C: Record of Change History

[This section describes the details of changes that have resulted in the current Low-Level Design document.]

# Appendix C: Record of Change History

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **#** | **Date** | **Document Version No.** | **Change Description** | **Reason for Change** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# Appendix D: Traceability Matrix

[Demonstrate the forward and backward traceability of the system to the functional and non-functional requirements documented in the Requirements Document.]

|  |  |  |
| --- | --- | --- |
| **Project Requirement Specification Reference Section No. and Name.** | **DESIGN / HLD Reference Section No. and Name.** | **LLD Reference Section No. Name** |
|  |  |  |
|  |  |  |
|  |  |  |