

The Role of Opinion in Community Detection in Social Media Network

submitted by

M Mahabub A Khoda

Class Roll No: 002210503049

Exam Roll: MCA244048

Master of Computer Application

under the supervision of

Prof. (Dr.) Nandini Mukherjee

in the

Department of Computer Science and Engineering

Faculty of Engineering and Technology

Jadavpur University

2024

Faculty of Engineering and Technology

Jadavpur University

CERTIFICATE OF RECOMMENDATION

To Whom It May Concern

I hereby recommend that the project report entitled “The Role of Opinion in Community Detection in Social Media Network” has been carried out by M Mahabub A Khoda (Reg. No.: 163641 of 2022-23, Roll No.: 002210503049, Exam Roll: MCA244048), under my guidance and supervision and be accepted in partial fulfillment of the requirement for the degree of Master of COMPUTER APPLICATION in DEPARTMENT of COMPUTER SCIENCE and ENGINEERING, JADAVPUR UNIVERSITY during the academic year 2023-2024.

Prof. (Dr.) Nandini Mukhopadhyay
Project Supervisor
Dept. of Computer Science and
Engineering
Jadavpur University

Prof. (Dr.) Nandini Mukhopadhyay
Head of the Department
Dept. of Computer Science and Engineering
Jadavpur University

Prof. Dipak Laha
Dean, Faculty Council
of Engineering & Technology
Jadavpur University, Kolkata-032

Faculty of Engineering and Technology

Jadavpur University

CERTIFICATE OF APPROVAL

This is to certify that the project report entitled “**The Role of Opinion in Community Detection in Social Media Network**” is a bonafide record of work carried out by M MAHABUB A KHODA, in fulfillment of the requirements for the award of the degree of Master of Computer Application in the Department of Computer Science and Engineering, Jadavpur University. It is understood that by this approval the undersigned do not necessarily endorse or approve any statement made, opinion expressed or conclusion drawn therein but approve the project report only for the purpose for which it has been submitted.

Signature of Examiner 1

Date :

Signature of Examiner 2

Date :

Faculty of Engineering and Technology
Jadavpur University

**DECLARATION OF COMPLIANCE WITH
ACADEMIC ETHICS**

I hereby declare that this project report entitled “The Role of Opinion in Community Detection in Social Media Network” is an outcome of my genuine and independent work, as part of my Master of Computer Application program.

All information in this document has been obtained and presented in accordance with academic rules and ethical conduct.

I also declare that, as required by these rules and conduct, all materials and results that are not original to this work have been fully cited and referenced.

M Mahabub A Khoda

Registration No.: 163641 of 2022-23

Examination Roll No.: MCA244048

Thesis Title:- The Role of Opinion in Community Detection in Social Media Network.

Signature with Date

ACKNOWLEDGEMENT

I want to express my gratitude and appreciation to all those who have contributed to the completion of this project. Their support, guidance, and encouragement have been invaluable throughout this research journey.

First and foremost, I am deeply indebted to my supervisor, Prof. Dr. Nandini Mukherjee, Department of Computer Science & Engineering, Jadavpur University, for his exceptional guidance, unwavering support, and insightful feedback. Her expertise, enthusiasm, and dedication have played a pivotal role in shaping this project. I am truly grateful for his constant availability, patience, and encouragement, which motivated me to strive for excellence.

I am grateful to the faculty members of the Computer Science & Engineering, at Jadavpur University for providing a stimulating academic environment. Their teachings, mentorship, and constructive criticism have been instrumental in broadening my horizons and honing my research skills.

Moreover, Prof. Nandini Mukhopadhyay, Head of the Computer Science & Engineering Department, at Jadavpur University has been there to listen and provide valuable advice.

Although it is impossible to mention everyone individually, please accept my sincere gratitude if you have played a part, big or small, in the completion of this project. Your contributions have been invaluable, and I am deeply thankful for your support.

M Mahabub A Khoda
Class Roll No.: 002210503049
Examination Roll No.: MCA244048
Department of Computer Science & Engineering
Jadavpur University

ABSTRACT

Community detection in online social networks is a complex research area, especially when context information is involved. This study addresses this challenge by developing a semi-supervised node labeling approach that enhances the Louvain algorithm to be context-aware. This improved method is tested against the standard Louvain algorithm using a dataset from Twitter, focusing on user interactions related to the “West Bengal State Assembly Election 2021”. The dataset models a network based on replies and retweets, incorporating rich context information.

The results show that the approach significantly increases modularity, more than doubling the value achieved by the unmodified Louvain algorithm. This improvement validates the effectiveness of incorporating context information into community detection, providing a more accurate representation of community structures in real-life network data. This study demonstrates that the context-aware Louvain algorithm is a powerful tool for detecting communities in online social networks, leading to better insights into user interactions and network dynamics.

CONTENTS

❖ CHAPTER-I:INTRODUCTION.....	1-3
➤ Overview.....	1
➤ Background and Motivation.....	1
➤ Objective.....	3
➤ Outline of the Thesis.....	3
❖ CHAPTER-II: LITERATURE SURVEY.....	4-7
❖ CHAPTER-III: DATA DESCRIPTION.....	8-9
❖ CHAPTER-IV: METHODOLOGY.....	10-17
➤ Using traditional Louvain Algorithm.....	10-12
■ Modularity Maximization Phase.....	11
■ Community Aggregation Phase.....	11
■ Algorithm for Tradition Louvain.....	12
➤ Weakness of the Louvain Algorithm.....	12
➤ Using Context-aware Louvain Algorithm.....	13-17
■ Node Labeling.....	13
■ Label Propagation.....	14
■ Incorporating Context in Louvain Algorithm.....	15
■ Algorithm for Context-aware Louvain.....	16
❖ CHAPTER-V: RESULTS AND DISCUSSION.....	18-26
➤ Result for February 1st Week.....	18
➤ Result for February 2nd Week.....	19
➤ Result for March 1st Week.....	20
➤ Result for March 2nd Week.....	22
➤ Result for April 1st Week.....	23
➤ Result for April 2nd Week.....	24
➤ Overall Observation.....	26
❖ CHAPTER-VI: DRAWBACKS OF CONTEXT-AWARE LOUVAIN.....	27
❖ CHAPTER-VII: CONCLUSION.....	28-30
➤ Future Scope.....	28
➤ Pros and Cons of Proposed Approach.....	29
References.....	31-33

LIST OF TABLES

Table Caption

- Table 3.1:
Representation of data in file1.csv.....8
- Table 3.2:
Representation of data in file2.csv.....9
- Table 5.7:
Comparison of Modularity Value for each collection of dataset for
Context-aware Louvain and traditional Louvain algorithm.....26

LIST OF FIGURES

Figure Caption

- Fig 5.1.1:
Output Graph for February 1st week by Traditional Louvain.....18
- Fig 5.1.2:
Output Graph for February 1st week by Context-aware Louvain.....19
- Fig 5.2.1:
Output Graph for February 2nd week by Traditional Louvain.....19
- Fig 5.2.2:
Output Graph for February 2nd week by Context-aware Louvain.....20
- Fig 5.3.1:
Output Graph for March 1st week by Traditional Louvain.....21
- Fig 5.3.2:
Output Graph for March 1st week by Context-aware Louvain.....21
- Fig 5.4.1:
Output Graph for March 2nd week by Traditional Louvain.....22
- Fig 5.4.2:
Output Graph for March 2nd week by Context-aware Louvain.....22
- Fig 5.5.1:
Output Graph for April 1st week by Traditional Louvain.....23
- Fig 5.5.2:
Output Graph for April 1st week by Context-aware Louvain.....24
- Fig 5.6.1:
Output Graph for April 2nd week by Traditional Louvain.....24
- Fig 5.6.2:
Output Graph for April 2nd week by Context-aware Louvain.....25

CHAPTER I - INTRODUCTION

1.1. Overview

Online social media platforms are web-based digital services that enable individuals to create, share content, and engage in virtual social interactions. These platforms have greatly simplified communication, allowing users to connect not just with acquaintances but also with people from many parts of the world. Presently, they are widely popular, functioning as central points where users from across the globe exchange their viewpoints on a multitude of topics. The topics discussed encompass personal experiences, as well as commentary on current events such as festivals, climate change, the environment, culture, politics, and many other related subjects. The ability to interact on such a broad scale has transformed the way information is disseminated and has fostered a global community where ideas and opinions can be shared instantly.

Social network analysis, an essential tool for understanding the interactions among various users on any social media platform, is the study of the structure and dynamics of social networks. It involves various techniques to analyze relationships and interactions within a network of individuals, organizations, or other entities. A critical aspect of social network analysis is community detection, which involves identifying clusters of nodes with dense internal connections. Community detection helps reveal the underlying structure within complex networks, enabling the discovery of groups with shared interests, behaviors, or attributes. This insight is invaluable for targeted marketing strategies, efficient resource allocation, improved collaboration within organizations, and more effective policy-making and intervention strategies tailored to specific sub-groups within larger populations. Integrating contextual information into community detection algorithms enhances the accuracy and relevance of identified communities, providing a deeper understanding of the diverse social landscapes present in online networks.

1.2. Background and Motivation

Currently, these social media platforms reflect all the activities going on around the globe in real-time. Subsequently, individuals have the opportunity to actively participate in current updates and discussions, making social media platforms indispensable as sources of immediate information and interaction. Whether it's breaking news, live events, or hot topics, users can stay informed and engage in global conversations as they develop. These interactions remain available on the platform along with their timestamps,

providing a historical record of discourse. These recorded interactions are reliable sources of data for social media network analysis, as they offer a dynamic overview of social networks by constantly updating with the most recent events and interactions. This captures users' changing relationships and groupings. Researchers and analysts utilize this vast amount of real-time data to find and study communities within online networks, revealing patterns of connection, similar interests, and influence dynamics. One such social media platform is Twitter. Twitter is a social media platform and microblogging service where users can post and interact with messages known as "tweets." Tweets were originally restricted to 140 characters but have been expanded to 280 characters for most languages. Users can also share photos, videos, links, and polls within their tweets. The real-time nature and brevity of tweets make Twitter a rich source of data for analyzing social interactions and trends.

Social network analysis involves the use of various techniques and tools to analyze the relationships and interactions within a network of individuals, organizations, or other entities. The aim is to help researchers investigate the patterns and implications of these connections and how they influence behaviors, information flow, and community cohesion. Applications of social network analysis include various fields such as sociology and anthropology, where it helps in understanding social structures, group dynamics, and cultural patterns; business and marketing, for analyzing organizational structures, consumer behavior, and the spread of information; and communication studies, to examine the flow of information and its impact.

Community detection in social network analysis offers significant advantages by revealing the underlying structure within complex networks. Identifying communities, or clusters of nodes with dense internal connections, helps in understanding the organization and functioning of social, biological, or information networks. This insight enables the discovery of groups with shared interests, behaviors, or attributes, facilitating targeted marketing strategies, efficient resource allocation, and improved collaboration within organizations. Additionally, community detection aids in identifying influential nodes and key connectors, enhancing strategies for information dissemination, epidemic control, and the resilience of networks against disruptions. By highlighting the modular nature of networks, community detection also supports more effective policy-making and intervention strategies tailored to specific sub-groups within larger populations. Various approaches exist for community identification; some use contextual information while others rely solely on network topology and interaction patterns. Integrating contextual information into community detection algorithms enhances the accuracy and relevance of identified communities, enabling a deeper understanding of the diverse social landscapes present in online networks.

1.3. Objective

There is a fixed number of communities for a particular twitter dataset used for community detection in the traditional Louvain algorithm, but the traditional Louvain algorithm does not look into the context of the dataset and thus gives multiple number of possible communities, as it only considers the edges or connections between the nodes or users in this case. So, in this project report the main objective is to incorporate these contexts for each user into these edges or connections and finally detect the communities for the Collection of twitter dataset.

In this report, a collection of tweets related to the "West Bengal State Assembly Election 2021" were taken, which were used by *Chakraborty et al.* [21]. This data has been used to form a network of users on Twitter regarding the said topic. The network is further analyzed and then a context-aware community detection approach was implemented on the network. This similar work is also done by *Chakraborty et al.* [4], on a collection of data from Twitter based on online interactions that feature the "Russia-Ukraine war".

1.4. Outline of the Report

In this project report, the data used for the network formation is described in Chapter III. The details of the Louvain algorithm, as well as the Context-aware Louvain algorithm, are discussed in Chapter IV. The comparison and results of both the Louvain algorithm and the Context-aware Louvain algorithm are discussed in Chapter IV. The results and discussions related to it are in Chapter V. The drawbacks of the Context-aware Louvain algorithm are discussed in Chapter VI. The project report concludes with a summary of findings and potential future work in Chapter VII.

CHAPTER II - LITERATURE SURVEY

At present, as social media networks have become a huge pool of information, community detection in these complex networks has grown in importance as a research topic. Using community detection algorithms, various insights about the complex network can be found. There are various community detection algorithms that can be used to extract various useful information from these large-scale networks. Choosing the appropriate community detection algorithm depends on the scenarios, the requirements and the constraints. Some of the most commonly used community detection algorithms include hierarchical clustering, modularity maximization, and spectral clustering.

In this section, some of the relevant works on community detection on different networks are discussed:

For Hierarchical clustering: It is a popular method for grouping nodes. It creates communities so that nodes within a community are similar to each other and different from nodes in other communities. This group of methods discover hierarchical community structures (i.e., dendrogram) in three ways: divisive, agglomerative and hybrid. Given are the few works based on hierarchical clustering:

In 2002, Girvan et al. [16] first highlighted a feature of community structure, where network nodes are joined together in tightly knit groups, between which there are only looser connections. So, they proposed a divisive algorithm to detect these groups or communities, built around the idea of using centrality indices to find community boundaries.

In 2003, Tyler et al. [25] proposed a modification of the Girvan–Newman algorithm, to improve the speed of the calculation. They proposed to calculate the contribution to edge betweenness only from a limited number of centers, chosen at random, deriving a sort of Monte Carlo estimate.

In 2004, Newman et al. [14] proposed an algorithm which used a divisive technique to iteratively remove edges from the network thereby breaking the whole network into communities, this breaking up was done based on edge betweenness measures. It also included a recalculation step in which betweenness scores are reevaluated after the removal of every edge.

In 2004, Newman [11] proposed an agglomerative hierarchical clustering algorithm to maximize the modularity of the network. It joins edges to shape larger communities in order to increase the modularity for the network.

In 2004, Clauset et al. [15] proposed an agglomerative hierarchical clustering algorithm which gradually merges nodes, each of which are initially regarded as a community. It enhances the running time of the algorithm for a large network, which makes it faster compared to other algorithms.

In 2008, Fernández et al. [23] proposed an algorithm which tries to solve the non-uniqueness problem in agglomerative hierarchical clustering, which is generally dealt with distinct criteria, which mostly consist of the selection of one out of various resulting hierarchies. Their proposed variable-group algorithm for agglomerative hierarchical clustering solves the ties in proximity problem by merging the two clusters that form a tie into a supercluster.

In 2018, Zarandi et al. [17] proposed an algorithm based on Structural Similarity (CDASS), which applies divisive and agglomerative strategies in a hybrid way. In the first phase it randomly removes some low similarity edges and thus breaks the network into several primary communities, then it merges the communities to get the final community structure. Then, in the second phase it uses an evaluation function to select the best communities between overall random generated communities.

For Spectral Clustering: Spectral clustering techniques make use of the spectrum (eigenvalues) of the similarity matrix of the data to perform dimensionality reduction before clustering in fewer dimensions..

In 2013, Amini et al. [18] proposed a method to group nodes in a network by using the normalized Laplacian and regularized adjacency matrix. Their work presents a faster technique to identify communities in networks using a pseudo-likelihood approach, which works well even when networks are large and sparse. They also improve the initial grouping with a modified version of spectral clustering, making the process more effective. This method reliably identifies community structures, particularly in simpler cases with two groups, solving some of the common challenges in analyzing large and sparse networks.

In 2014, Siemon et al. [19] analyzed the Laplacian spectra of the macroscopic anatomical neuronal networks of the macaque and cat. The findings were similar to the traditional graph metrics, revealing an integrative community structure within the neural brain networks.

For Modularity Maximization: The main goal of modularity maximization algorithm is to maximize the modularity value of the network, that is, the number of edges lying within communities in a network relative to the number of edges expected on the basis of chance. Given are the few works based on modularity maximization:

In 2006, Danon et al. [24] suggested that normalizing the modularity variation produced by the merging of two communities by the fraction of edges incident to one of the two communities, in order to favor small clusters. This led to better modularity optima especially for the communities that are different in size.

In 2008, Bakillah et al. [23] enhanced the fast-greedy optimization of modularity clustering algorithm with semantic similarity to deal with the complex social graphs extracted from the twitter and thus investigate how graph clustering can be applied to support the detection of geo-located communities in a typhoon oriented network.

In 2008, Blondel et al. [2] proposed a heuristic method based on modularity optimization to extract the community structure in data from a Belgian mobile phone network. The popular Louvain algorithm for modularity maximization was proposed in this work.

In 2011, De Meo et al. [6] presented a strategy for uncovering the community structure within networks, (potentially large ones). The method relies on optimizing network modularity and introducing a unique measure of edge centrality based on k-paths.

In 2013, Kim et al. [3] proposed a partitioned Louvain algorithm that was validated on an actual Twitter dataset. Their proposed method performed better for bigger data size compared to other algorithms with respect to the modularity of the network where the performance of other algorithms degraded for larger dataset.

In 2013, Waltman et al. [10] proposed a modularity based community detection algorithm for the large networks, that uses the modularity value after the local movement of nodes to neighboring communities in a more sophisticated way. It not only merges communities, but also splits communities if needed in order to increase modularity value.

In 2015, Que et al. [7] proposed a novel graph mapping and data representation method to improve the body of existing results by designing a highly scalable parallel Louvain algorithm for distributed memory systems that can analyze graphs with hundreds of billions of edges.

In 2015, Traag [12] proposed the Random Neighbor Louvain algorithm which is an enhanced version of the Louvain algorithm [2], where instead of searching for the best neighbor, which gives the high modularity value and selecting it, it selects a random neighbor. This way it tries to reduce the higher computation time of the Louvain algorithm.

In 2015, Ozaki et al. [9] proposed a simple acceleration algorithm, the Louvain Prune algorithm, which was based on the Louvain algorithm [2]. Their proposed algorithm enables the

analysis of large networks up to ten times faster without quality deterioration by pruning the inefficient process on almost every kind of network.

In 2018, Zhang et al. [13] implemented a new algorithm known as Random Self-adaptive Neighbor Louvain algorithm[2], which is based on the principle of small probability events to enhance the performance of Louvain algorithm[2]. The algorithm picks up a few neighbors at random based on the probability and one node is selected, which maximizes the modularity.

In 2020, Kim et al. [1] focused on community detection based on geosocial co-clustering in networks. By incorporating geographic constraints, their proposed method outperforms the modified Louvain algorithm and can identify communities that have both social and spatial coherence.

In 2020, Seifikar et al. [8] proposed a dynamic community detection algorithm based on Louvain algorithm[2], which from the previous snapshots of the communities and network changes generates a compressed graph which is much denser and informative compared to the original graph.

From the study of these above-mentioned works, it is observed that none of these works have incorporated the sentiment of the nodes in a network for community detection. It completely detects the community based on the structural properties of the network.

So, in 2023, Chakraborty et al. [4] proposed an algorithm, which incorporated the sentiment of each node in the network by adding edges between nodes having the same sentiment and modifying weight values for the edges between the nodes based on some criteria.

CHAPTER III - DATA DESCRIPTION

The prerequisite for this work was the collection of Twitter data related to West Bengal State Assembly Elections. A collection of tweets for six weeks were taken (the first and third weeks of February, March, and April 2021), where the last 2 weeks coincided with the election dates, stored as a csv file. For the first and second week of February, there were 2193 and 1788 tweets respectively. For the first and second week of March, there were 5637 and 4834 tweets respectively. For the first and second week of April, there were 7798 and 7195 tweets respectively. For all the tweets there were total 33 fields or columns which were: *id, conversation_id, created_at, date,timezone, place, tweet, language, hashtags, cashtags, user_id, user_id_str, username, name, day, hour, link,urls, photos, video, thumbnail, retweet, nlikes, nreplies, nretweets, quote_url, search, near, geo, source, user_rt_id, user_rt, retweet_id, reply_to, retweet_date, translate, trans_src and trans_dest*. Which generally had the description about the tweets like username, tweet content, time zone, hashtags, date, time and so on.

As, all the fields were not needed, so except the fields username, hashtags and tweet fields all other fields were dropped. Here, the username field has the name of the user by whom the tweet was done, the content of the tweet is stored in the tweet field and the hashtags field contains all the hashtags used in the tweets. From these tweets all the users mentioned in that tweet were extracted and stored in a list as 'lst' in another csv file (say, *file1.csv*, shown in Table 3.1) along with 'src' which is the value in username.

src	lst
user1	[user2, user3, user4]

Table 3.1: Representation of data in *file1.csv*

Now, from the new csv file each mentioned user was taken from the list for a row and were put in different rows making separate pairs with the 'src' username and stored them in another csv file. For example if a row had 'user1' in 'src' field and ['user2', 'user3', 'user4'] in 'lst' field, then for the new csv file 'user1' as 'src' and 'user2' as 'dst' for a row were taken, then 'user1' as 'src' and 'user3' as 'dst' for the next row and 'user1' as 'src' and 'user3' as 'dst' for the next row and then move on to the next row for the former csv file. In this way, all the edges, i.e., pairs of nodes where the values at 'src' and 'dst' of each row is a node and they pair up together to form an edge are found. This is done for the whole collection of tweets and is stored in another csv file (say, *file2.csv*, shown in Table 3.2). Which will be the fundamental elements to build up a network for the collection of tweets.

src	dst
user1	user2
user1	user3
user1	user4

Table 3.2: Representation of data in file2.csv

Now, keeping this csv file for edge aside, the most important file, i.e., file which will contain labels of each user to their community based on their tweet content has to be created. This labeled dataset will be used to identify the label of each user in src .

To enhance the results obtained from the original Louvain algorithm, the context of the tweets are incorporated and the already constructed graph is modified. More precisely, the sentiment of a user is fetched by analyzing his/her tweets.

In the dataset used, the opinions expressed by the users of the opinionated network can be grouped into 4 crisp classes - Supports X, Supports Y, Supports Z, and Neutral. For example, assuming support for X as supporting party X, support for Y as supporting party Y, support for Z as supporting party Z, and posts only on view as considered neutral, and the posts made by each user of the network can be assigned an Opinion Class. This file has been used in Context Aware Louvain Algorithm as described in Section 4.3.

CHAPTER IV - METHODOLOGY

4.1. Using Traditional Louvain Algorithm

The Louvain algorithm focuses on maximizing the modularity for each detected community, aiming to enhance the strength of connections within communities while reducing the strength of connections between different communities. Modularity, a crucial metric in network analysis, is a scalar value ranging from -1 to 1 that quantifies the density of links inside each community compared to links between communities. A high modularity value indicates a network with strong, well-defined communities, where the connections between nodes of the same community are significantly denser than the connections between nodes of different communities.

In the context of weighted networks, where edges have associated weights representing the frequency of interactions (such as the number of communications between two users in a mobile phone network), modularity is adapted to account for these weights. This adaptation allows for a more detailed and accurate representation of the network's structure, reflecting not just the presence of connections but their varying strengths. The formal definition of modularity in weighted networks considers the sum of the weights of edges within communities and compares it to the expected sum if the edges were distributed randomly, adjusted for the weight distribution across the entire network.

This sophisticated measure of calculating modularity ensures that communities are identified not merely based on the number of connections but also on the significance of those connections, thereby providing a more comprehensive understanding of the network's community structure. The main goal of the traditional Louvain algorithm is to achieve a partitioning that maximizes modularity, thereby highlighting communities with dense internal connections and sparse external connections, which in turn, reveals the underlying organization and dynamics of complex networks.

This traditional Louvain algorithm operates in an iterative manner, involving two main phases: modularity optimization and community aggregation. These phases are repeated until the modularity gain stabilizes or reaches a maximum, indicating that an optimal or near-optimal community structure has been identified.

4.1.1. Modularity Maximization Phase

Modularity [26] compares the actual density of edges within communities to the expected density if edges were distributed randomly, thus offering a quantitative assessment of the community structure. The formal definition of modularity Q is given by:

$$Q = \frac{1}{2m} \sum_{i,j} (A_{ij} - \frac{k_i k_j}{2m}) \delta(C_i, C_j)$$

where A_{ij} represents the weight of the edge between nodes i and j , k_i and k_j denote the degrees of nodes i and j respectively, m is the total weight of all edges in the network, and $\delta(C_i, C_j)$ is the Kronecker delta, which is 1 if nodes i and j are in the same community, and 0 otherwise. This equation essentially measures the difference between the actual edge weight and the expected edge weight for each pair of nodes, summing these differences for all pairs within the same community and normalizing by the total edge weight in the network.

4.1.2. Community Aggregation Phase

In the second phase, known as community aggregation, the Louvain algorithm constructs a new network where each node represents a community identified in the previous phase. This process aggregates communities into super-nodes, facilitating the detection of hierarchical structures within the network. The weights of edges between these new nodes are determined by summing the weights of edges between nodes in the corresponding original communities. By treating communities as single entities, the algorithm aims to identify higher-level structures and relationships between communities, enabling a more comprehensive understanding of the network's organization. The aggregation phase is crucial for uncovering nested community structures and refining the partitioning of the network into distinct clusters.

Algorithm for Traditional Louvain:

Input:

- *Initial Graph G.*

G is the initial Graph

1. Put each node of G in its own community (partition).
2. Q = modularity of G with the initial partition.
3. **while** some nodes are moved **do**
 - a. **for** each node u of G **do**
 - i. Place u in its neighboring community including its own community that it belongs to, on condition that it maximizes the modularity gain.
 - b. **end for**
4. **end while**
5. Q_1 = modularity of G with current partition
6. **if** $Q_1 > Q$ **then**
 - a. $G \leftarrow$ the network between communities of G with respect to the final partition.
7. **else**
 - a. **Return**
8. **end if**

Output:

- *Community structure with maximum modularity.*

4.2. Weakness of the Louvain Algorithm

The output graph from this algorithm is constructed using data from tweets, focusing specifically on interactions between users on the platform. This graph is undirected and any edges present initially are weighted as 1, meaning that it treats all interactions as equally important and does not consider the direction or intensity of these interactions. Additionally, the graph lacks any contextual information beyond the basic interactions.

The Louvain algorithm, which is applied to this graph, seeks to maximize modularity, a measure of the quality of the division of the graph into communities. Modularity quantifies how well a graph is partitioned by comparing the density of edges within communities to the density of edges between communities. The algorithm works by iteratively grouping nodes into communities to improve modularity, eventually identifying

distinct communities within the graph. For instance, in this study, different connected components, or clusters of nodes, represent different communities.

However, in this context, these connected components reflect interactions centered around the same topic, each associated with a different sentiment. Identifying these communities is difficult as it helps to reveal the underlying structure of user interactions and the sentiments they express. The fundamental issue with the Louvain algorithm is its reliance on the structural properties of the network, such as edge weights and connectivity patterns, while neglecting node attributes and external factors. This focus on structure alone means that the algorithm might overlook important contextual information that could influence community assignments of the components.

4.3. Using Context-aware Louvain Algorithm

As in the Traditional Louvain Algorithm discussed in Section 4.1, the lack of context while creating the network is noticed. So, to overcome this limitation, the edges in the graph can be redefined to include more meaningful data. One potential approach is to use node labeling based on sentiment for community detection. By assigning labels to nodes according to the sentiment of their tweets, the algorithm can incorporate this contextual information, leading to more meaningful and accurate community identification.

For example, if two users frequently share similar sentiments, the edges between them can be weighted more heavily, reflecting a stronger connection. Conversely, interactions between users with differing sentiments can be given lower weights or even ignored. This adjustment allows the Louvain algorithm to consider the content and sentiment behind the interactions, not just the structure of the network.

By using this improved method, the algorithm can more effectively capture the complexities of social interactions on Twitter, resulting in a deeper and more detailed understanding of the communities within the network. The following section delves deeper into these alternative methods, exploring how they can enhance community detection by integrating additional contextual insights and ultimately providing a more accurate representation of user interactions and sentiments on the platform.

4.3.1. Node Labeling

The tweets in the current dataset should ideally be classified into one of the four possible sentiments: "Supports BJP," "Supports TMC," "Supports Sanjukta Morcha" and "Undetermined." However, manually labeling all 29,445 tweets in the

used dataset is not practical due to the immense time and effort required, as described in Chapter III.

To address this challenge, a strategic approach is taken by initially focusing on a smaller subset of the dataset. In the paper, a team of volunteers manually labeled 5,000 out of the 29,445 tweets. These labels were assigned to the four sentiment categories: "Supports BJP," "Supports TMC," "Supports Sanjukta Morcha", and "Undetermined." This subset of labeled data provides a crucial foundation for the next steps in the process.

Each node 'n' in the network, irrespective of its significance in the network structure, is a part of the 'Opinionated Network' because of their social media post 'p' regarding the same topic/context at time t_i . From the dataset, the texts with hashtags such as "#BengalElection2021" and "#WestBengalAssemblyElection2021" are classified as Neutral parties, whereas '#BangleNijerMeyeKeiChay' and '#BengalWithTMC' are considered positive supporters of a political party TMC, '#BengalWithModi' and '#ModirSatheiAsolPoribarton' are determined to be positive supporters of BJP and '#vote4sanjuktamorcha' and '#vote4left' are considered positive supporters of Sanjukta Morcha.

4.3.2. Label Propagation

To label the remaining 24,445 tweets, the Label Propagation algorithm [27] was employed, a semi-supervised machine-learning technique. This algorithm works on the principle that nodes (in this case, tweets) in a network are likely to share the same label as their neighbors. Essentially, it propagates labels from the small set of manually labeled tweets to the rest of the network.

The Label Propagation algorithm starts by assigning initial labels to the seed nodes. Then, it iteratively updates the labels of the unlabeled nodes based on the labels of their neighboring nodes. Over several iterations, the algorithm spreads the labels throughout the network until it converges, meaning that the labels stabilize and do not change with further iterations.

This method allows us to efficiently and effectively classify a large dataset with minimal manual labeling effort. By leveraging the relationships and similarities between tweets, the Label Propagation algorithm helps us extend the initial manual labeling to the entire dataset, ensuring that all tweets are categorized into one of the four sentiment groups. This approach not only saves time and resources but

also enhances the accuracy and consistency of the sentiment classification across the dataset.

4.3.3. Incorporating Context in Louvain Algorithm

Having the context of the orientation of each person from the node labeling and label propagation approach, the edge weights can be modified based on this labeled data for the Louvain algorithm to yield a better division of the network into different communities. Context information from the node labeling approach is utilized along with the Louvain algorithm to achieve more precise and meaningful results. The node labeling approach categorizes the graph into four distinct labels, each representing a specific context or orientation of the nodes.

The input to the algorithm is an undirected and unweighted graph G and the context (or label) of each node C . Where C is a dictionary which has key as the node and the community label as value. Given that G may be disconnected, it is necessary to ensure that all nodes can be considered within a unified framework. To address this, a head node u is chosen for each label, and all other nodes in G are traversed. For each traversed node v , if $(label[u] = label[v])$, then, an edge $e(u, v)$ is added between this u and v . This step is crucial because the Louvain algorithm cannot merge components that are not connected. By connecting nodes of the same label to a central head node, the graph G is made fully connected within each label category, ensuring that the Louvain algorithm can process the entire network cohesively.

The next step involves converting the unweighted graph G into a weighted graph G' . This conversion ensures that the edge weights reflect the contextual information, thereby enhancing the community detection process. By incorporating context into the edge weights, the algorithm can more effectively identify communities that reflect the true nature of the relationships within the network. This approach allows the Louvain algorithm to perform more accurately and meaningfully, taking into account the inherent contextual relationships between nodes.

For the said approach, the label information C is used to modify the edge weights. Specifically, for an edge $e(u, v)$, if the labels of nodes u and v are the same $(label[u] = label[v])$, i.e., if both of their context are same, then the edge weight of the edge $e(u, v, w)$ is set to $w = 1$; otherwise, $e(u, v, w)$ is set to $w = 0$.

i.e.,

$$e(u, v) \leftarrow e(u, v, w),$$

where,

$$w = \begin{cases} 1, & \text{if } \text{label}[u] = \text{label}[v] \\ 0, & \text{else} \\ \end{cases}$$

By applying this modified Louvain algorithm, the resulting community structure aligns closely with that derived from the node labeling approach. This indicates that the context-aware modification enhances the algorithm's ability to detect communities that are more coherent and relevant to the underlying social structure.

Algorithm for Context-aware Louvain:

Input:

- *Initial graph G, where*
 - *u and v represent nodes,*
 - *e represents an edge,*
 - *e(u, v) represents an unweighted edge between nodes u and v, and*
 - *e(u, v, w) represents a weighted edge between the same, where w is the weight of the edge.*
- *Color of each node C, where*
 - *C[u] represents label of the node u, and*
 - *c represents a label in C*

G is the initial graph and C is the label of each node

1. **for** each distinct label c in C **do**
 - a. Choose a node u with label c
 - b. head = c
 - i. **for** u in G **do**
 1. **if** C[u] = c, **then**
 - a. $G \leftarrow G \cup e(\text{head}, u)$
 2. **end if**
 - ii. **end for**
 - c. **end for**

2. Initialize an empty graph H
 - a. **for** $e(u, v)$ in G **do**
 - i. **if** $C[u] = C[v]$ **then**
 1. $w \leftarrow 1$
 - ii. **else**
 1. $w \leftarrow 0$
 - iii. **end if**
 - iv. $H \leftarrow e(u, v, w)$
 - b. **end for**
3. Initialize a disjoint set D , where each node u are put in different set $S[u]$ of their own
4. $Q \leftarrow \text{modularity}(H, D)$
5. $\text{moved} \leftarrow \text{true}$
6. **while** moved **do**
 - a. $\text{moved} \leftarrow \text{false}$
 - b. **for** v in H **do**
 - i. $\text{newedge} \leftarrow \text{false}$
 - ii. **for** each neighbor u of v in H **do**
 1. $D_1 \leftarrow D$
 2. $D_1 \leftarrow D_1(S[v] \leftarrow S_1[v] \cup S_1[u])$
 3. $X \leftarrow \text{modularity}(H, D_1)$
 4. **if** $X > Q$
 - a. $Q \leftarrow X,$
 - b. $Xu \leftarrow u, Xv \leftarrow v$
 - c. $\text{newedge} \leftarrow \text{true}$
 5. **end if**
 - iii. **end for**
 - iv. **if** newedge **then**
 1. $D \leftarrow D(S[Xv] \leftarrow S[Xv] \cup S[Xu])$
 2. $\text{moved} \leftarrow \text{true}$
 - v. **end if**
 - c. **end for**
7. **end while**
8. $\text{Comm} \leftarrow u: \text{name}(S[u]), \text{ for all } u \text{ in } H$
9. **return** Comm

Output:

- Community structure with maximum modularity w.r.t. context.

CHAPTER V - RESULTS AND DISCUSSION

From the results of both the algorithms it is observed that for all the results generated for traditional Louvain, the number of communities is much much higher than that of the results generated for Context-aware Louvain Algorithm. Whereas, in the results of the Context-aware Louvain the number of distinct communities are 4. For all the results for Context-aware Louvain, the nodes in each graph are labeled according to the communities they belong to. The colors, red, green and blue represent any of the political parties, and the yellow represents the nodes for which the party couldn't be decided which may be due to their lack of tweeting or them being against one of the parties. Also it is observed that the modularity of graphs generated with the community partition by Context-aware Louvain algorithm is higher than that of the modularity of the graph generated with community partition by traditional Louvain algorithm. In the graph , distinct color means distinct community.

5.1. Result for February 1st Week Dataset



Fig 5.1.1: Output Graph for February 1st week by Traditional Louvain

First, the dataset of the first week of February was taken. The dataset contained a total of 2193 tweets.

Then the dataset was modified to d as discussed in Chapter III. After which, the traditional Louvain algorithm was applied on it as discussed in Section 4.1 and a dictionary with community partition was generated. Now, Gephi [20] software was used to visualize the graph generated with that community partition, which is shown in *Fig 5.1.1*. Then the Context-aware Louvain algorithm was applied on the modified dataset d as discussed in Section 4.3 and got a dictionary with community partition. Now, the Gephi [20] software was again used to visualize the graph generated with



Fig 5.1.2: Output Graph for February 1st week by Context-aware Louvain

that community partition, which is shown in Fig 5.1.2. It is observed in Fig 5.1.1. the number of distinct colors is higher than that of Fig 5.1.2. As distinct color means distinct community, it is observed that the nodes of same colors are closer to each other for Fig 5.1.1. But in Fig 5.1.2. it is observed that not necessarily the neighboring nodes are of the same color. Also the modularity for Fig 5.1.2. which is 0.8198738882413366, is more than that of Fig 5.1.1. which is 0.7270871858274518. This shows that the density of the community is higher for Fig 5.1.2.

As this was the time before the declaration of the election so the number of tweets were lesser compared to other months.

5.2. Result for February 2nd Week Dataset



Fig 5.2.1: Output for February 2nd week by Traditional Louvain

Then, the dataset for the second week of February was taken. The dataset contained a total of 1788 tweets.

Then, the dataset was modified to d as discussed in Chapter III. After which, the traditional Louvain algorithm was applied on it as discussed in Section 4.1 and a dictionary with community partition was generated. Now, Gephi [20] software was used to visualize the graph generated with that community partition, which is shown in Fig 5.2.1. Then the Context-aware Louvain algorithm was applied on the modified dataset d as discussed in Section 4.3



Fig 5.2.2: Output Graph for February 2nd week by Context-aware Louvain

and got a dictionary with community partition. Now, Gephi [20] software was used to visualize the graph generated with that community partition for this month, which is shown in *Fig 5.2.2*. It is observed that in *Fig 5.2.1* that the number of distinct colors is higher than that of *Fig 5.2.2*. As distinct color means distinct community, it is seen that the nodes of same colors are closer to each other for *Fig 5.2.1*. But in *Fig 5.2.2* it is observed that not necessarily the neighboring nodes are of the same color. Also the modularity for *Fig 5.2.2* which is 0.8143037508476152, is more than that of *Fig 5.2.1* which is 0.7113977906794182. This shows that the density of the community is higher for *Fig 5.2.2*.

As the election was declared at the end of the month, which was at the end of this period, so, the number of tweets were still lesser compared to other

months.

5.3. Result for March 1st Week Dataset

The dataset of the first week of March was taken. The dataset contained a total of 5637 tweets. Then, the dataset for this month was modified to d as discussed in Chapter III. After which, the traditional Louvain algorithm was applied on it as discussed in Section 4.1 and got a dictionary with community partition. After that, Gephi [20] software was used to visualize the graph generated with that community partition, which is shown in *Fig 5.3.1*. Then the Context-aware Louvain algorithm was applied on the modified dataset d as discussed in Section 4.3 and a dictionary with community partition was generated. Now, according to the community partition Gephi [20] software was used to visualize the graph generated with that community partition, which is shown in *Fig 5.3.2*. It is observed that in *Fig 5.3.1*. the number of distinct colors is higher than that of *Fig 5.3.2*. As distinct color means distinct community, it is observed that the nodes of same colors are closer to each other for *Fig 5.3.1*. But in *Fig 5.3.2*. it is observed that not necessarily the neighboring nodes are of the same color. Also the modularity for *Fig 5.3.2*. which is



Fig 5.3.1: Output Graph for March 1st week by Traditional Louvain algorithm.

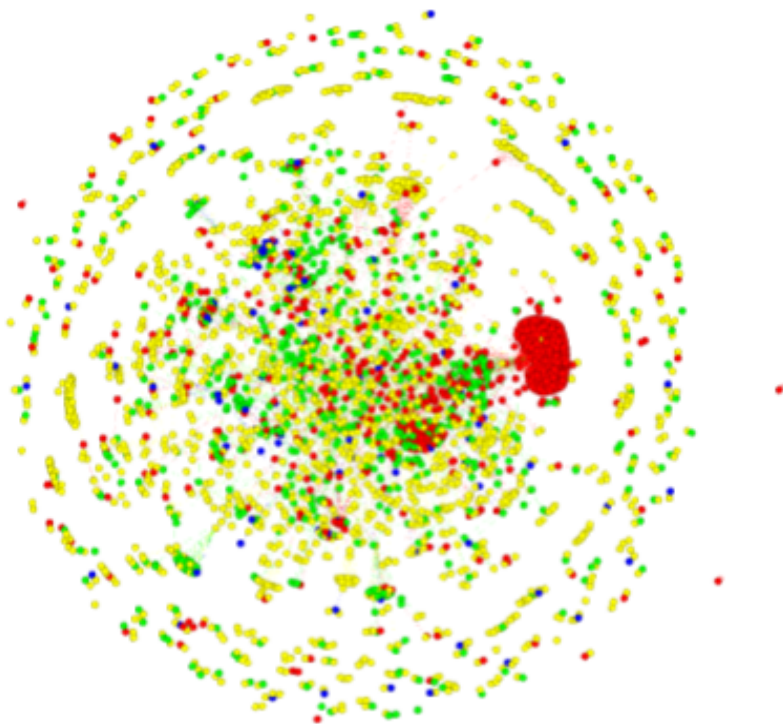


Fig 5.3.2: Output Graph for March 1st week by Context-aware Louvain algorithm.

0.8492870106952973, which is more than that of Fig 5.3.1. which is 0.703277126736111. This shows that the density of the community is higher for Fig 5.3.2.

As the election was declared before this month, so, the number of tweets increased rapidly on the election topic. Most of the tweets were related to pandemic as during this period the pandemic due to COVID-19 was at its peak, so most of the people opposed the idea of organizing elections. But, they were not supporting any community so they were labeled as 4, for which in Fig 5.3.2 it is noticed that the yellow color is more frequent. Also, it is observed that the number of red and green nodes in Fig 5.3.2 were much more, which is because the tweets were more related to online campaigning and debates between two communities. This idea of debate could be deducted from the observation that the red and green nodes are seen together often. Also, few of the blue nodes can also be observed, which are much lesser in frequency which indicates the less number of participation by 3rd community.

5.4. Result for March 2nd Week Dataset

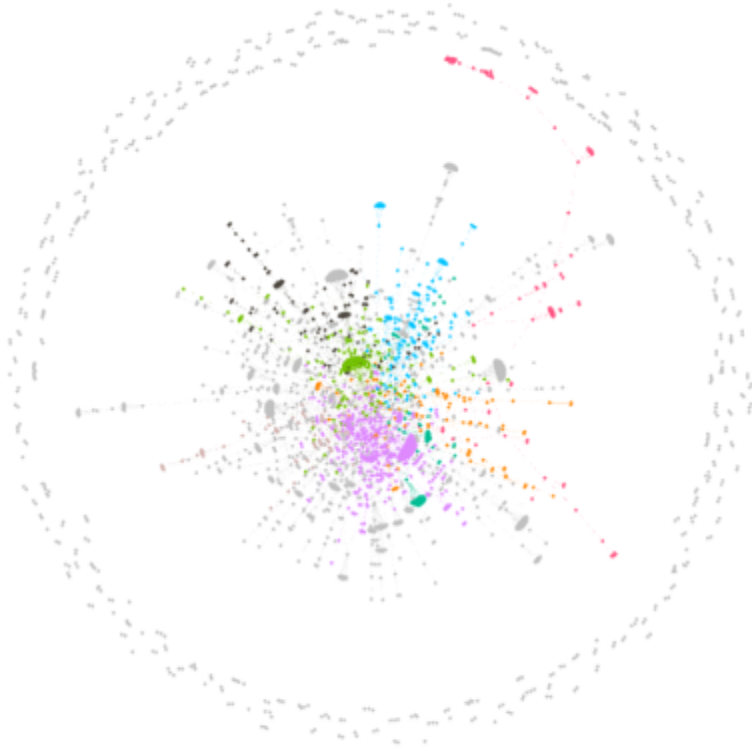


Fig 5.4.1: Output Graph for March 2nd week by Traditional Louvain algorithm.

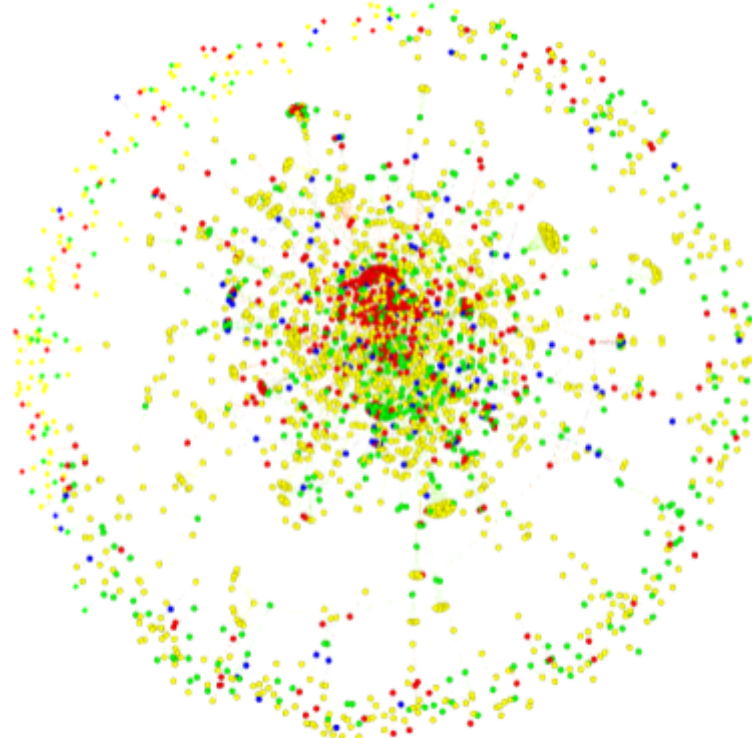


Fig 5.4.2: Output Graph for March 2nd week by Context-aware Louvain algorithm.

The dataset of the second week of March was taken. The dataset contained a total of 4834 tweets. Then, the dataset for this month was modified to d accordingly as discussed in Chapter III. After which, the traditional Louvain algorithm was applied on it as discussed in Section 4.1 and a dictionary with community partition was generated. After that, Gephi [20] software was used to visualize the graph generated with that community partition, which is shown in Fig 5.4.1. Then the Context-aware Louvain algorithm was applied on the modified dataset d as discussed in Section 4.3 and got a dictionary with community partition. Now, according to the community partition Gephi [20] software was used to visualize the graph generated with that community partition, which is shown in Fig 5.4.2. It is observed that in Fig 5.4.1. the number of distinct colors is higher than that of Fig 5.4.2. As distinct color means distinct community, it is observed that the nodes of same colors are closer to each other for Fig 5.4.1. But in Fig 5.4.2. it is observed that not necessarily the neighboring nodes are of the same color. Also the modularity for Fig 5.4.2. which is 0.8320993713521705, which is more than that of Fig 5.4.1. which is 0.7587334472201814. This shows that the density of the community is higher for Fig 5.4.2.

As all the political parties started their campaign various news and views for or

and against many topics were posted by the users. So, the number of tweets increased rapidly on the election topic. Most of the tweets were mostly users from different communities blaming and pointing out mistakes of the other political parties. So, it is observed in *Fig. 5.4.2* that the number of green and red nodes are almost visually equal, also the number of unidentified community nodes are lesser compared to other months. Also, few of the blue nodes can also be observed, which are much lesser in frequency which again indicates the less number of participation by 3rd community.

5.5. Result for April 1st Week Dataset



Fig 5.5.1: Output Graph for April 1st week by Traditional Louvain algorithm.

The dataset of the first week of April was taken. The dataset contained a total of 7798 tweets. Then, the dataset for this month was modified to d accordingly as discussed in Chapter III. After which, the traditional Louvain algorithm was applied on it as discussed in Section 4.1 and a dictionary with community partition was generated. After that, Gephi [20] software was used to visualize the graph generated with that community partition, which is shown in *Fig 5.5.1*. Then the Context-aware Louvain algorithm was applied on the modified dataset d as discussed in Section 4.3 and got a dictionary with community partition. Now, according to the community partition Gephi [20] software was used to visualize the graph generated with that community partition, which is shown in *Fig 5.5.2*. It is observed that in *Fig 5.5.1*, the number

of distinct colors is higher than that of *Fig 5.5.2*. As distinct color means distinct community, it is observed that the nodes of same colors are closer to each other for *Fig 5.5.1*, which shows the high number of interactions between the users. But in *Fig 5.5.2*, it is observed that not necessarily the neighboring nodes are of the same color. Also, the modularity for *Fig 5.5.2*, which is 0.8303278827391466, which is more than that of *Fig 5.5.1*, which is 0.7526738268985951. This shows that the density of the community is higher in *Fig 5.5.2* for communities generated by Context-aware Louvain.

In the *Fig. 5.5.2* the increase in the number of yellow nodes is observed and is scattered throughout the Graph. This indicates the increase in the number of users

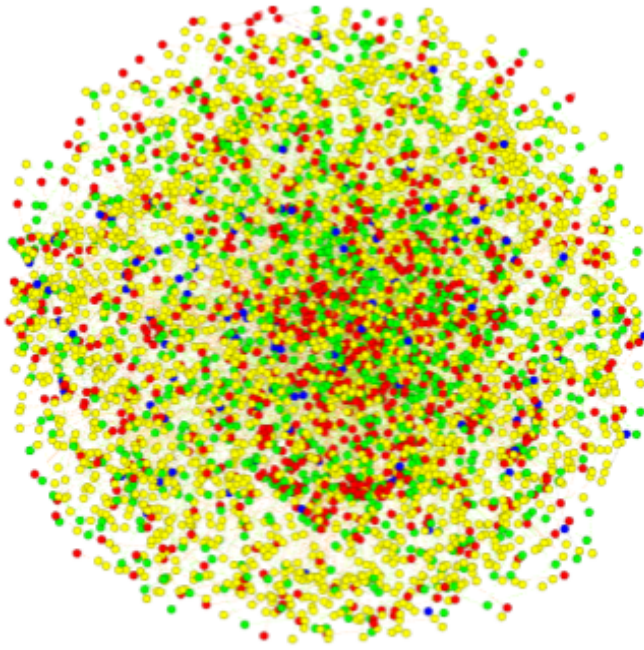


Fig 5.5.2: Output Graph for April 1st week by Context-aware Louvain algorithm.

tweeting against multiple political parties. The compactness of the Graph in Fig 5.5.2 indicates the huge number of interactions among users of different communities. This can be justified by the fact that during this time, people were gathered by various political parties for rallies, to which some users opposed while some supported, which resulted in online arguments among the users. The blue nodes can also be observed, which are much lesser in frequency compared to the frequency of nodes for other communities, but it is higher than its own frequency compared to other months, which indicates that the users supporting the 3rd political party have started participating more at this time compared to other months..

5.6. Result for April 2nd Week Dataset



Fig 5.6.1.: Output Graph for April 2nd week by Traditional Louvain algorithm.

The dataset of the second week of April was taken. The dataset contained a total of 7195 tweets. Then, the dataset for this month was modified to d accordingly as discussed in Chapter III. After which, the traditional Louvain algorithm was applied on it as discussed in Section 4.1 and a dictionary with community partition was generated. After that, Gephi [20] software was used to visualize the graph generated with that community partition, which is shown in Fig 5.6.1. Then the Context-aware Louvain algorithm was applied on the modified dataset d as discussed in Section 4.3 and got a dictionary with community partition. Now, according to the community partition Gephi [20] software was used to visualize the graph generated with that

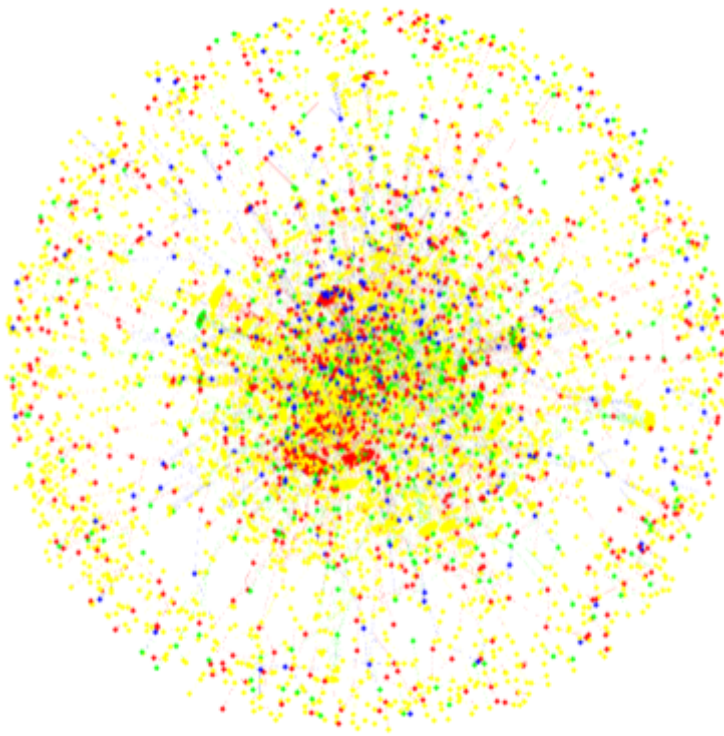


Fig 5.6.2: Output Graph for April 2nd week by Context-based Louvain algorithm.

community partition, which is shown in Fig 5.6.2. It is observed that in Fig 5.6.1. the number of distinct colors is higher than that of Fig 5.6.2. As distinct color means distinct community, it is observed that the nodes of same colors are closer to each other for Fig 5.6.1, which shows the high number of interactions between the users. But in Fig 5.6.2. it is observed that not necessarily the neighboring nodes are of the same color. Also, the modularity for Fig 5.6.2. which is 0.7403712457553524, which is more than that of Fig 5.6.1. which is 0.6516939402461368. This shows that the density of the community is higher in Fig 5.6.2 for communities generated by Context-aware Louvain algorithm.

In Fig. 5.6.2 it is observed that the yellow nodes are more in number.

That is, most of the users posted tweets which were related to the election but did not support any particular communities or political parties. This can be justified by the fact that during this time arose the scarcity of various healthcare equipment like oxygen, hospital beds and so on. So, people were posting tweets, criticizing the political parties for their lack of concern for common people.

5.7. Overall Observation

The overall modularity for all the datasets for each algorithm are recorded in *Table 5.7*. It is noticed that for each dataset the modularity of the network with respect to the community partitions given by Context-aware Louvain algorithm is always higher than that of the modularity of the network with respect to the community partitions given by traditional Louvain algorithm. This high modularity is the result of the addition of extra edges during the first phase of the Context-aware Louvain algorithm.

It is also observed that the number of communities in the traditional Louvain algorithm is much higher compared to that of the Context-aware Louvain algorithm. This is because the Node Labeling and Label Propagation algorithm mentioned in Section 4.3.1 and 4.3.2 puts a label for each user(or node), thus, limiting the number of communities to be formed in the network.

Dataset for	Modularity for Context-aware Louvain	Modularity for traditional Louvain
February 1	0.8198738882413366	0.7270871858274518
February 2	0.8143037508476152	0.7113977906794182
March 1	0.8492870106952973	0.703277126736111
March 2	0.8320993713521705	0.7587334472201814
April 1	0.8303278827391466	0.7526738268985951
April 2	0.7403712457553524	0.6516939402461368

Table 5.7: Comparison of Modularity Value for each collection of dataset for Context-aware Louvain and traditional Louvain algorithm.

CHAPTER VI - DRAWBACKS OF CONTEXT-AWARE LOUVAIN

The proposed node labeling and label propagation approach for graph coloring demonstrates a reasonably effective method for visualizing community structures within a network. However, there are few drawbacks. One significant limitation is that a huge number of nodes are labeled as 4, indicating indeterminate sentiment and represented by the yellow color. This indeterminacy arises primarily because these nodes, or users, either did not tweet at all within the network or their tweets were ambiguous, i.e., either opposing the community or not showing support for any specific community.

Consequently, this leads to a significant problem of overlapping communities. If a user opposes one community, it is possible that they might support the ideologies of the other two communities. However, the applied Context-aware Louvain algorithm, similar to the traditional Louvain algorithm, only considers a crisp set of communities. As a result, users with overlapping or unclear community affiliations are also labeled as 4, highlighting a limitation in effectively capturing the accurate community structures within the network.

Additionally, the proposed Context-Aware Louvain algorithm has a higher computation time compared to the traditional Louvain algorithm. This increased computational time is a significant drawback, as it can limit the algorithm's scalability and efficiency, especially when applied to large-scale networks. The higher computation time stems from the additional complexity introduced by context-awareness, which requires more intensive processing to analyze interactions and determine probabilistic relationships between nodes.

CHAPTER VII - CONCLUSION

Despite the challenges mentioned in Chapter VI, the Context-Aware Louvain algorithm represents an important advancement in the field of community detection and network analysis. It provides a more clear understanding of user interactions and community structures by incorporating context into the clustering process. However, the identified drawbacks highlight the need for further optimization and refinement. Future work could focus on developing more efficient algorithms that maintain the benefits of context-awareness while reducing computation time. Additionally, exploring alternative methods for handling indeterminate nodes, such as incorporating more sophisticated probabilistic models or machine learning techniques, could enhance the accuracy and reliability of the clustering results.

7.1. Future Scope

There are various approaches that can be used to solve the problem of large undetermined nodes that results in large size of community 4.

7.1.1. Probabilistic Existence of Edge Approach:

The issue of indeterminate nodes can be mitigated to some extent by delving deeper into the analysis of user interactions within the network. One potential approach to improve the determination of node labels involves calculating the probability of an edge existing between two nodes with specific labels, denoted as (C_i, C_j) . Here, C_i and C_j represent possible labels within the community. If the label of a node cannot be conclusively determined through direct analysis, an indirect method can be employed. This involves finding an edge to an already decisively labeled node and then assigning the label to the indeterminate node aware on the maximum probability of interaction, $P[x][y]$, where $P[x][y]$ represents the likelihood of an edge existing between labels x and y . Although this probabilistic approach may yield better clustering results and a more accurate representation of community structures, it is important to note that it does not produce deterministic outcomes. The results can vary depending on the network's characteristics and the specific interactions analyzed.

7.1.2. Keyword Classification and Rough Sets Approach:

A method can also be developed that sorts tweets by username to ensure all tweets from the same user are considered together. By using a predefined

collection of keywords or phrases that indicate support for or against each political party, a membership score can be assigned to the users. These scores, denoted as μ , reflect the degree of a user's association with each party based on the presence of specific keywords in their tweets. If a tweet contains keywords supporting multiple parties, the membership score is proportionally divided among those parties. Also, if a tweet contains keywords against a party, the membership score is redistributed among the remaining parties.

By applying the Rough Set theory, the users can be categorized into distinct communities based on their highest membership scores. Users with a clear preference for one party are assigned to that party, while those with divided support are placed in overlapping communities. Users with equal scores across all parties are considered neutral. This refined method not only addresses the indeterminate labeling issue but also provides a more precise visualization of overlapping communities, enhancing the understanding of the complex dynamics within the network.

7.2. Pros and Cons of the proposed approaches:

Probabilistic Existence of Edge Approach:

This method makes use of the network's structure and interactions to deduce information that may be missing, providing a clearer picture of user association with a community. Its ability to adjust dynamically to the network's features can lead to better clustering outcomes. This flexibility ensures that the analysis remains relevant as the network evolves. Ultimately, this approach offers valuable insights into user association with a community and community formations within the network.

One drawback of this approach is that it doesn't always give the same results. It can vary based on how the network is set up and the interactions that are looked at. This makes it harder to depend on the outcomes for making decisions. Also, using this method might need a lot of computing power. It can take a long time to accurately figure out probabilities and study how people interact in the network. This complexity might make it tough to use for big networks with lots of data.

Keyword Classification and Rough Set Approach:

One key advantage of this approach is its systematic classification of users based on their tweet content. By organizing tweets by username and employing predefined keyword sets, users can be accurately assigned membership scores reflecting their association with different political parties. Additionally, using Rough Set theory provides a clear and rule based method for determining community membership, by efficiently handling overlapping communities and neutrality. This approach of integrating content analysis and mathematical classification method gives more clearer and interpretable results, which helps in better understanding of the network behavior.

However, a notable drawback of this approach is its reliance on the quality of the predefined keyword sets. The accuracy of the method is completely dependent on the accuracy of these keywords in capturing user associations. Inaccurate or outdated keyword sets may result in misclassifications or incomplete representations of user associations, compromising the reliability of the analysis. Additionally, the manual effort required to update and manage the keyword sets can be huge, making it a challenging task to efficiently implement and maintain the classification system over time.

Both the approaches have their own set of advantages and disadvantages, yet each offers potential solutions to the challenges like undetermined nodes and overlapping communities. However, it is important to acknowledge that despite these efforts, the possibility of encountering overlapping communities and undetermined nodes cannot be entirely removed. However, the implementation of the mentioned approaches is expected to significantly alleviate these challenges, thus enhancing the overall effectiveness of community detection and network analysis methodologies.

REFERENCES

1. Jungeun Kim, Jae-Gil Lee, Byung Suk Lee, and Jiajun Liu. 2020. "Geosocial co-clustering: A novel framework for geosocial community detection." *ACM Transactions on Intelligent Systems and Technology (TIST)* 11, 4 (2020), 1–26.
2. Vincent D. Blondel, Jean-Loup Guillaume, Renaud Lambiotte and Etienne Lefebvre. 2008. "Fast unfolding of communities in large networks." *J. Stat. Mech.* (2008) P10008.
3. Yong-Hyuk Kim, Sehoon Seo, Yong-Ho Ha, Seongwon Lim and Yourim Yoon. 2013. "Two Applications of Clustering Techniques to Twitter: Community Detection and Issue Extraction." *Discrete Dynamics in Nature and Society*. Volume 2013 | Article ID 903765.
4. Amartya Chakraborty, Deepamoy Pal, and Nandini Mukherjee. 2023. "Context-aware Community Detection in the Russia-Ukraine Conflict Network." *ICDCN '24*, January 4–7, 2024, Chennai, India.
5. Pasquale De Meo, Emilio Ferrara, Giacomo Fiumara, Alessandro Provetti. 2013. "Enhancing community detection using a network weighting strategy." *Information Sciences*, 222:648-668, 2013.
6. Pasquale De Meo, Emilio Ferrara, Giacomo Fiumara, and Alessandro Provetti. 2011. "Generalized Louvain method for community detection in large networks." *Proceedings of the 11th International Conference On Intelligent Systems Design And Applications*, pp. 88-93, 2011.
7. Xinyu Que, Fabio Checconi, Fabrizio Petrini and John A. Gunnels. 2015. "Scalable Community Detection with the Louvain Algorithm." *2015 IEEE International Parallel and Distributed Processing Symposium*, May 25-29, 2015, Hyderabad, India.
8. Mahsa Seifikar, Saeed Farzi and Masoud Barati. 2020. "C-Blondel: An Efficient Louvain-Based Dynamic Community Detection Algorithm." *IEEE Transactions on Computational Social Systems*, Vol. 7, No. 2, April 2020.
9. Naoto Ozaki, Hiroshi Tezuka and Mary Inaba. 2015. "A Simple Acceleration Method for the Louvain Algorithm." *International Journal of Computer and Electrical Engineering*.
10. Ludo Waltman and Nees Jan van Eck. 2013. "A smart local moving algorithm for large-scale modularity-based community detection." *Eur. Phys. J. B* 86, 471 (2013).
11. M. E. Newman. 2004. "Fast algorithm for detecting community structure in networks." *Physical review* E69 (6) (2004) 066133
12. V. A. Traag. 2015. "Faster unfolding of communities: speeding up the Louvain algorithm." *Physical Review E* 92(3) (2015) 032801.

13. Ziqiao Zhang, Peng Pu, Dingding Han and Ming Tang. 2018. "Self-adaptive Louvain algorithm: Fast and stable community detection algorithm based on the principle of small probability event." *Physica A: Statistical Mechanics and its Applications* Volume 506, 15 September 2018, Pages 975-986
14. M. E. Newman and M. Girvan. 2004. "Finding and evaluating community structure in networks." *Physical Review E*, vol. 69, no. 2, p. 026113, 2004.
15. A. Clauset, M. E. Newman, and C. Moore. 2004. "Finding community structure in very large networks." *Physical Review E*, vol. 70, no. 6, p. 066111, 2004.
16. M. Girvan and M. E. Newman. 2002. "Community structure in social and biological networks." *PNAS*, vol. 99, no. 12, pp. 7821–7826, 2002.
17. F. D. Zarandi and M. K. Rafsanjani. 2018. "Community detection in complex networks using structural similarity." *Physica A: Statistical Mechanics and its Applications*, vol. 503, pp. 882–891, 2018.
18. Arash A. Amini, Aiyu Chen, Peter J. Bickel and Elizaveta Levina. 2013. "Pseudo-likelihood methods for community detection in large sparse networks." *The Annals of Statistics*, vol. 41, no. 4, pp. 2097–2122, 2013.
19. Siemon C. de Lange, Marcel A. de Reus, and Martijn P. Van Den Heuvel. 2014. "The Laplacian spectrum of neural networks." *Frontiers in Computational Neuroscience*, vol. 7, p. 189, 2014.
20. Mathieu Bastian, Sebastien Heymann, and Mathieu Jacomy. 2009. *Gephi: An Open Source Software for Exploring and Manipulating Networks*. (2009).
<http://www.aiai.org/ocs/index.php/ICWSM/09/paper/view/154>
21. Amartya Chakraborty, Nikhil Badyal, Aman Sharma and Nandini Mukherjee. 2022. "A Novel Centrality-based Measure for Election Network Analysis." 2022 IEEE Silchar Subsection Conference (SILCON), 04-06 November, 2022, Silchar India.
22. Mohamed Bakillah, Ren-Yu Li, and Steve HL Liang. 2014. Geo-located community detection in Twitter with enhanced fast-greedy optimization of modularity: the case study of typhoon Haiyan. *International Journal of Geographical Information Science*, DOI: 10.1080/13658816.2014.964247
23. Fernandez, A. and Gomez, S., 2008. "Solving non-uniqueness in agglomerative hierarchical clustering using multidendrograms." *Journal of Classification*, 25, 43–65. doi:10.1007/s00357-008- 9004-x
24. L. Danon, A. Díaz-Guilera and A. Arenas. 2006. "The effect of size heterogeneity on community identification in complex networks." *J. Stat. Mech.* 11 (2006) 10.

25. J.R. Tyler, D.M. Wilkinson and B.A. Huberman. 2003. "Email as spectroscopy: Automated discovery of community structure within organizations." In: *Communities and Technologies*, Kluwer, B.V., Deventer, The Netherlands, 2003, pp. 81–96.
26. M.E.J. Newman. 2004. "Analysis of weighted networks." *Phys. Rev. E* 70 (5) (2004) 056131.
27. Xiaojin Zhu and Zoubin Ghahramani. 2002. "Learning from Labeled and Unlabeled Data with Label Propagation." URL: <https://api.semanticscholar.org/CorpusID:15008961>