

The Role of Opinion in Influence Maximization in Online Social Networks

submitted by

Subhankar Naskar

Class Roll No: 002210503044

Exam Roll: MCA244043

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 Influence Maximization in Online Social Networks” has been carried out by Subhankar Naskar (Reg. No.: 163636 of 2022-23, Roll No.: 002210503044, Exam Roll: MCA244043), 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 Influence Maximization in Online Social Networks**” is a bonafide record of work carried out by SUBHANKAR NASKAR, 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 Influence Maximization in Online Social Networks” 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.

Subhankar Naskar

Registration No.: 163636 of 2022-23

Examination Roll No.: MCA244043

Thesis Title:- The Role of Opinion in Influence Maximization in Online Social Networks.

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 her 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.

Subhankar Naskar
Class Roll No.: 002210503044
Examination Roll No.: MCA244043
Department of Computer Science & Engineering
Jadavpur University

ABSTRACT

Influence Maximization (IM) is an optimization problem in social networking to identify the most influential users in social networks. IM problem, with large-scale data, faces many challenges like time-efficiency, accuracy, and effectiveness of seed. To solve these challenges, we propose a Community-based Context-aware Influence Maximization algorithm. Community-based Context-aware Influence Maximization uses a community-based framework to improve time efficiency, significantly reducing search space. It considers the user's interests to address the effectiveness of the seed.

The traditional information diffusion models (i.e., Independent cascade model) have been extended to the Context-aware Context-aware Independent Cascade model for influence spreading. To identify the most influential users, first propose a Community Detection Algorithm (CDA) to partition the network into sub-networks based on the community label. After analyzing the dataset, a total of 3 communities BJP, TMC, and Sanjukta Morcha are formed. First, the Greedy approach implements the IM algorithm and then the CELF approach. Then the comparison result of the execution time of each algorithm is represented by the Python module matplotlib. After finding the result on the main dataset the Context-aware Graph Reduce Algorithm (Section: 4.4) is used to reduce the size of the main graph. After graph reduction, the IM algorithm (Section: 4.1) with the Greedy approach (Section: 4.2) and the CELF approach (Section: 4.3), and then the comparison result of both the algorithms is represented by the Python package matplotlib.

So, this paper highlights how the context-sensitive framework enables faster influence maximization over the network using existing Greedy and CELF models of influence spread.

CONTENTS

Chapter Name	Page
1. Chapter I: Introduction	01
1.1 Motivation and Challenges.	02
1.2 Objectives.	02
1.3 Organization of the Report.	03
2. Chapter II: Literature Survey	04
3. Chapter III: Data Description	07
3.1 Node Labeling.	08
3.2 Label Propagation.	09
4. Chapter IV: Methodology	10
4.1 Influence Maximization Problem.	10
4.2 Greedy Models on Influence Maximization.	11
4.3 CELF Models on Influence Maximization.	12
4.4 Proposed Graph Reduced Algorithm.	14
5. Chapter V: Experiments & Results	15
5.1 Experiment on the First Week of February 2021 Dataset.	15
5.2 Experiment on the Second Week of February 2021 Dataset.	17
5.3 Experiment on the First Week of March 2021 Dataset.	20
5.4 Experiment on the Second Week of March 2021 Dataset.	22
5.5 Experiment on the First Week of April 2021 Dataset.	24
5.6 Experiment on the Second Week of April 2021 Dataset.	26
5.7 Overall Observations.	28
6. Chapter VI: Conclusion	29
6.1 Limitations.	29
6.2 Future Work.	30
References	32

LIST OF TABLES

Table Caption	Page
➤ Table No: 3.1 Representation of data in file1.csv	07
➤ Table No: 3.2 Representation of data in file2.csv	08
➤ Table No: 5.1 Influencing Person Identification of February Week - 1	16
➤ Table No: 5.2 Influencing Person Identification of February Week - 2	18
➤ Table No: 5.3 Influencing Person Identification of March Week - 1	21
➤ Table No: 5.4 Influencing Person Identification of March Week - 2	23
➤ Table No: 5.5 Influencing Person Identification of April Week - 1	25
➤ Table No: 5.6 Influencing Person Identification of April Week - 2	27

LIST OF FIGURES

Figure Caption	Page
➤ Figure No: 5.1.1 Feb 1st Week Main Graph.	15
➤ Figure No: 5.1.2 Feb 1st Week Reduced Graph.	15
➤ Figure No: 5.1.3 CELLF vs Greedy Computation Time Comparison of February - 1 Dataset. .	17
➤ Figure No: 5.2.1 Feb 2nd Week Main Graph	18
➤ Figure No: 5.2.2 Feb 2nd Week Reduced Graph	18
➤ Figure No: 5.2.3 CELLF vs Greedy Computation Time Comparison of February - 2 Dataset. . .	19
➤ Figure No: 5.3.1 March 1st Week Main Graph.	20
➤ Figure No: 5.3.2 March 1st Week Reduced Graph.	20
➤ Figure No: 5.3.3 CELLF vs Greedy Computation Time Comparison of March - 1 Dataset.	21
➤ Figure No: 5.4.1 March 2nd Week Main Graph	22
➤ Figure No: 5.4.2 March 2nd Week Reduced Graph.	22
➤ Figure No: 5.4.3 CELLF vs Greedy Computation Time Comparison of March - 2 Dataset.	23
➤ Figure No: 5.5.1 April 1st Week Main Graph.	24
➤ Figure No: 5.5.2 April 1st Week Reduced Graph.	24
➤ Figure No: 5.5.3 CELLF vs Greedy Computation Time Comparison of April - 1 Dataset.	25
➤ Figure No: 5.6.1 April 2nd Week Main Graph	26
➤ Figure No: 5.6.2 April 2nd Week Reduced Graph.	26
➤ Figure No: 5.6.3 CELLF vs Greedy Computation Time Comparison of April - 2 Dataset.	27

LIST OF ABBREVIATIONS

BTC	Bitcoin.
CELF	Cost Effective Lazy Forward.
CSV	Comma Separated Value
DAG	Direct Acyclic Graph.
ICM	Independent Cascade Model.
IM	Influence Maximization.
LTM	Linear Threshold Model.
MIA	Maximum Influence Arborescence.
SA	Simulated Annealing.

Chapter I - Introduction

In today's interconnected world, information and ideas spread rapidly through social networks, online communities, and communication platforms. Understanding how influence propagates within these networks is crucial for various applications, from marketing campaigns and viral content dissemination to public health interventions and social movements. This field of study, known as influence maximization, seeks to identify the most influential nodes within a network that can trigger the largest cascade of influence when activated.

The importance of influence maximization stems from the inherent structure of networks. They are not simply collections of isolated nodes, but rather interconnected systems where the behavior of one node can significantly impact others. This interconnectedness allows for efficient information diffusion, but it also highlights the power of strategically placed individuals who can act as catalysts for change. Identifying these influential nodes allows us to optimize resource allocation and maximize the impact of campaigns.

The field of influence maximization draws upon concepts from graph theory, network science, and computational complexity. Researchers have developed various algorithms to identify influential nodes, each with its strengths and weaknesses. These algorithms analyze network properties such as connectivity, centrality, and community structure to predict the potential spread of influence from a given set of seed nodes. In this research study, the IM (Influence Maximization) algorithm for both the Greedy Approach and the CELF approach is implemented.

Chakraborty et al. (2024) [28], proposed a work which develops a framework for opinion-aware network formulation from online topical networks and its subsequent analysis. Novel concepts of opinion class and opinion compatibility are proposed for a context-aware graph-reduce approach. Experiments highlight how the context-sensitive framework enables faster influence maximization over the network using existing Greedy and CELF models of influence spread. They worked on 2022's Ukraine VS Russia War dataset but in this work, a different dataset is used i.e., West Bengal 2021's Election 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 [29]*. This data has been used to form a network of users on Twitter regarding the said topic. The paper is organized as follows: (A) A Brief study of influence maximization-related algorithms is

given in Chapter II, (B) The data used for network formation in this work is described in Section 3, (C) Section 4 gives details of the Influence Maximization related algorithms to the studied network, (D) A comparison of the influence maximization by greedy approach and CELF approach performance of is provided in Section 5, (E) Finally the concluding remarks are given in Section 6.

1.1 Motivation and Challenges

Elections stand as pivotal moments in democratic societies, molding the course of governance and the formulation of public policy. By analyzing the dynamics of electoral processes, including candidate selection, campaigning strategies, and voter behavior, researchers gain insights into the underlying mechanisms driving political outcomes. Influence maximization techniques offer a lens through which to it can be observed how information spreads within the electorate, potentially impacting voter preferences and decision-making.

In the context of election campaigns, identifying influencing individuals or groups can be instrumental in maximizing the reach and impact of campaign messages. By leveraging influence maximization models, political strategists can strategically allocate resources to target key segments of the electorate, thereby optimizing their chances of electoral success. Understanding who holds influence over voter opinions and behaviors can inform campaign strategies and resource allocation decisions.

To find the most influencing person from the huge number of election datasets the biggest challenge is related to computation time. If the dataset and the seed set are too large of a graph then the computation time to find the most influencing person is very complex and time-consuming. However, the computation time will be reduced if the graph is reduced by the Graph Reduction Algorithm based on some context.

To find the most influencing person from the context-based community first, label all the users. Labeling all the users of the dataset is sometimes a very time-consuming process, this is also one of the main challenges.

1.2 Objectives

The current work meets the following objectives:

- There are many campaigns in the era of social media through online interaction. Selecting the right community from the campaign and identifying the most influencing persons from that community is the main objective of this project.

- Another objective is to give a comparative result of the Greedy and CELF approaches on the main and reduced datasets.

1.3 Organization of the Report

The rest of the report is organized as follows:

- Chapter 2 gives a study of the recent research on Context-aware Influence Maximization approaches.
- Chapter 3 illustrates the general techniques used to identify the most influential person in a social network like the Greedy Approach and CELF approach.
- Chapter 4 elaborates on the implementation details of the current work undertaken, a comprehensive discussion on the results of individual applied models and noteworthy observations is provided in Chapter 5.
- Lastly, the concluding remarks are given in Chapter 6 along with limitations of the current work and future scope.

Chapter II - Literature Survey

In today's digitally interconnected world, social networks have become powerful platforms for communication, information dissemination, and influence propagation. The last decades have witnessed the booming of online social networks where hundreds of millions interact with each other and produce unprecedented content. The prevalence of online social networks has prompted much attention on information diffusion, as a piece of information could quickly become pervasive through the “word-of-mouth” propagation among friends in the network.

For example, In early 2021, Elon Musk had the cryptocurrency market wrapped around his finger. A tweet about dogecoin by the Tesla and SpaceX founder sent prices flying as much as 50%. The tweet was, “**You can now buy a Tesla with bitcoin**”: One of the largest companies in the world accepting BTC as payment helped push the asset to a new high. BTC would reach nearly \$65,000 within a month of the tweet.

This diffusion phenomenon is powerful in many applications, such as the adoption of political standpoints and technical innovations. As such, information diffusion in online social networks has attracted extensive research efforts from multiple fields, including computer science, physics, etc. Apart from this, it is used in various domains like marketing, public health, and social sciences as well.

As a key algorithmic problem in information diffusion research, influence maximization (IM) has been extensively studied recently due to its potential commercial value. The IM problem was originally studied as an algorithmic problem by *Domingos et al. [1]*, and *Richardson et al. [2]*, while *Kempe et al. [3]* were the first to formulate the problem as a discrete optimization problem.

A common limitation of existing approaches is computational inefficiency on large networks. Thus, two major types of solutions have been proposed.

First, a line of heuristic algorithms has been proposed. *Chen et al. [4]* proposed degree discount heuristics under the uniform IC (Independent Cascade) model, in which all edge probabilities are the same. Experimental results indicate that the new heuristics are efficient with satisfactory influence spread. Following the idea, *Chen et al. [5]* proposed a Maximum Influence Arborescence (MIA) model to evaluate the influence spread of a user. *Kimura et al. [6]* proposed the shortest path-based influence cascade models and efficient algorithms to compute influence spread under modified IC models.

For the LT (Linear Threshold) model, *Chen et al. [7]* observed that the spread can be computed in linear time on directed acyclic graphs (DAGs). They experimentally

showed that this heuristic is significantly faster than the greedy algorithm and achieves high-quality seed set selection, measured in terms of the spread achieved. Similarly, *Goyal et al. [8]* proposed SIMPATH, an efficient and effective algorithm for influence maximization under the LT (Linear Threshold) model that addresses these drawbacks by incorporating several clever optimizations. The work by *Narayanam et al. [9]*, proposed a Shapley value-based heuristic SPIN for the LT model. However, SPIN only relies on the evaluation of influence spreads of seed sets, and thus does not use specific features of the LT model. Moreover, SPIN is not scalable, with running time comparable as shown in [9] or slower than the optimized greedy algorithm.

Second, several optimized greedy algorithms have been proposed. *Lescovec et al. [10]* proposed the CELF (Cost Effective Lazy Forward) algorithm, which achieves 700 times speed improvement for the simple greedy algorithm. Later, *Goyal et al. [11]*, proposed CELF++ (The advanced version of the CELF model), an extension to CELF that further reduces the number of spread estimation calls, which achieves approximately 35% – 55% speedup compared with CELF. Besides, *Chen et al. [12]* proposed the NewGreedy and MixedGreedy algorithms in the IC (Independent Cascade) model with uniform probabilities. However, their performance is nonsteady, sometimes even worse than CELF. *Wang et al. [13]* discussed the influence maximization from the view of social community and proposed a new community-based greedy algorithm for mining top-k influential nodes.

Besides these two types of solutions, *Jiang et al. [14]* proposed a different approach based on Simulated Annealing (SA) for the influence maximization problem. *Guo et al. [15]* investigated the personalized influence maximization problem. *Rodriguez et al. [16]* studied the influence maximization problem in continuous time diffusion networks. The competitive influence maximization problem in social networks was studied in the works [17], [18]. The problem of minimizing the spread of misinformation in social networks was discussed in the works of [19], and [20]. Recently, data-driven methods were proposed to solve the influence maximization problem by maximizing likelihood functions w.r.t. real propagation cascades.

Data-driven methods are based on the assumption that the pairwise transmission probabilities are unknown in a network. In contrast, simulation-based methods are based on the assumption that the pairwise transmission probabilities are known prior. In data-driven methods, *Barbieri et al. [21]* studied social influence from the topic modeling perspective. *Bonchi [22]* discussed what to learn and how to learn from trace data of past propagations to identify influential users. *Goyal et al. [23]* proposed to directly use the past available data instead of assuming influence probabilities are given as input.

The complementary problem of learning influence probabilities from the available data was studied in the works [24], [25], [26], and [27].

IM (Influence Maximization) aims to select a set of 'k' number of users in an online social network, aka. seed set with the maximum influence spread, i.e., the expected number of influenced users through the seed set in information diffusion is maximized. A well-known application of IM is a political campaign on social media where a political party may wish to spread its campaign on social media. Apart from this, another application of IM is viral marketing, where a company may wish to spread the adoption of a new product from some initially selected adopters through social links between users. Besides viral marketing, IM has many other important applications such as network monitoring, rumor control, social recommendation, etc.

Influence maximization is the process of identifying a subset of individuals (seeds) whose activation yields the maximum influence spread in a network. *Kempe et al. [3]* in 2003 were the first to formalize IM as the following combinatorial optimization problem: Given a network with 'n' nodes and given a "spreading" or propagation process on that network, choose a "seed set" S of size $k < n$ to maximize the number of nodes in the network that are ultimately influenced.

Some of the challenges in finding the most influencing nodes (seed nodes) in the social network are: The first challenge is how to model the information diffusion process in a social network, which would heavily affect the influence spread of any seedset in IM. Second, the IM problem is theoretically complex in general. It has been proven that the optimal solution of IM is NP-hard under most of the diffusion models.

So, in this comparative study, the main focus is on a context-aware graph reduction approach. These experiments highlight how the context-sensitive framework enables faster influence maximization over the network using existing Greedy and CELF models of influence spread.

Chapter III - Data Description

The prerequisite for this work was the collection of Twitter data related to the West Bengal State Assembly Elections 2021. First three months of twitter data is collected (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 a total of 33 fields or columns which were: id, conversation_id, created_at, date, timezone, place, tweet, language, hashtags, 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 of 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 further will be used in this work to find the seed nodes in the context-based influence maximization algorithms.

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

In the current dataset, the opinions expressed by the users of the opinionated network can be grouped into 4 crisp classes - Supports BJP, Supports TMC, Supports Sanjukta Morcha, 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 (OCI).

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 this dataset, the texts with hashtags such as "#BengalElection2021" and "#WestBengalAssemblyElection2021" are classified as Neutral parties, whereas '#BangleNijerMeyeKeiChay' and '#BengalWithTCM' are considered positive supporters of a political party TMC and '#BengalWithModi' and '#ModirSatheiAsolPoribarton' are determined to be positive supporters of BJP and '#vote4sanjuktamorcha' and '#vote4left' are considered positive supporters of Sanjukta Morcha.

3.1 Node Labeling

The tweets in the current dataset should be ideally classified into one of the four possible sentiments, namely, “Supports BJP”, “Supports TMC”, “Supports Sanjukta Morcha”, and “Undetermined”. In this dataset, a semi-supervised learning approach is taken as it is not feasible to manually label a total of 29,445 tweets as described in above of this section. Thus, a semi-supervised approach is undertaken where a small fraction of the data is labeled. Out of the 29,445 tweets, 5000 tweets were manually labeled by a team of volunteers. The number of labels assigned in the dataset was four, corresponding to the 4 types of tweets in the dataset, “Supports BJP”, “Supports TMC”, “Supports Sanjukta Morcha”, and “Undetermined”. Next, the Label Propagation algorithm is used to label the remaining unlabeled data. The label propagation algorithm is a semi-supervised machine learning algorithm that is used to propagate labels in a network. It is based on the idea that nodes in a network are more likely to adopt the labels of their neighbors, and it aims to propagate labels from a small set of seed nodes to the rest of the network.

3.2 Label Propagation

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 helps in classifying a large dataset efficiently and effectively 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.

Chapter IV - Methodology

The opinions expressed by the users of the opinionated network can be grouped into 4 crisp classes - “Supports BJP”, “Supports TMC”, “Supports Sanjukta Morcha”, and “Undetermined”. 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 (OCI).

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 . 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 as discussed.

The modified network was obtained after the detection of Opinion Class. In contrast to the prior step, data is now represented as {node, ocl} along with Adjacency List, AL[]. By leveraging the OCL (Opinion Compatibility List) information for each node. next, the Opinion Compatibility is proposed, which serves as the measure of whether the opinions expressed by two neighboring nodes ‘n1’ and ‘n2’ are similar or not at time t_i in the network. The Opinion Compatibility measure is tagged to each edge $e_{i,j}$ between two neighboring nodes n_i and n_j , which carries a value 1 if the OCL_i and OCL_j are equal, else it is set to 0.

4.1 The Influence Maximization Problem

Influence Maximization (IM) is first modeled as an algorithmic problem by *Kempe et al. in 2003 [3]*. This problem studies a social network represented as a graph $G = (V, E)$, where ‘V’ is the set of nodes in G (i.e., the pair of source users, src and destination users, dst) and ‘E’ is the set of edges (directed or undirected) in the graph, G (i.e., social links between users). The main goal of the IM problem is to find a k-sized set of users with the maximum influence in graph G. The influence of any seed set is defined based on the information diffusion process among the users. An example of information

diffusion is a political campaign in social media, where a political party spreads its campaign on social media.

Algorithm of Influence Maximization Problem

Input: Graph object (G), set of seed nodes (S), propagation probability (p), and the number of Monte-Carlo simulations (mc).

Output: Average number of nodes influenced by the seed nodes.

Initialization: Empty list spread[].

for each monte-carlo simulation **do**

Initialization: new_active[] list with S[:] and A[] list with S[:]

while each new_active node **do**

Initialization: Empty list new_ones[].

for each node from the new_active list **do**

 np.random.seed(i)

 success=np.random.uniform(0, 1, len(g.neighbors(node, mode="out"))) < p

 new_ones += list(np.extract(success, g.neighbors(node, mode="out")))

end for

 new_active = list(set(new_ones) - set(A))

 A = A + new_active

end while

 append (len(A)) to spread

end for

return np.mean(spread)

4.2 Greedy Model on Influence Maximization

The Greedy model is one of the simplest and most commonly used approaches for solving the Influence Maximization problem in social networks which is proposed by *Kempe et. al. (2003) [3]*. The basic idea behind this Greedy model is to iteratively select nodes as seed individuals in the network based on their potential to maximize influence spread. This finds the node with the biggest influence spread, adds it to the seed set, and then finds the node with the next biggest influence spread over and above the original influence spread, and so on until 'k' seed nodes are found. The greedy model only needs to calculate the influence spread of the sum of (n-i) where i=0 to 'k' nodes which is approximately equal to 'kn'.

Algorithm of Greedy Model

Input: Graph object (g), number of seed nodes (k), propagation probability (p), and the number of Monte-Carlo simulations (mc).

Output: Optimal seed set, resulting spread, time for each iteration.

Initialization: Empty list S[], spread[], timelapse[], and set the current system time to start_time.

for each seed node 'k' **do**

Initialization: best_spread = 0

for each node in set(range(g.vcount())) - set(S) **do**

 s = IC(g, S + [j], p, mc)

if s > best_spread **then**

 best_spread = s

 node = j

end if

end for

 append node to S

 append best_spread to spread

 append (time.time() - start_time) to timelapse

end for

return (S, spread, timelapse)

4.3 CELF Model on Influence Maximization

The **CELF (Cost Effective Lazy Forward)** model was developed by *Leskovec et al. in 2007 [10]*. Although the Greedy algorithm works fast, it still needs to be faster when used on realistically sized networks. CELF was one of the first significant subsequent improvements.

CELF exploits the sub-modularity property of the influence spread function, which implies that the influence spread of a given node in one iteration of the Greedy algorithm cannot be any larger than its influence spread in the previous iteration. This helps to choose the nodes for which the influence spread function will evaluate in a sophisticated manner, rather than simply evaluating the influence spread for all nodes.

More specifically, in the first round, it calculates the influence spread for all nodes (like Greedy) and stores them in a list. Naturally, the top node is added to the seed set in the first iteration, and then removed from the list. In the next iteration, only the influence spread for the top node is calculated. After restoring, that node remains at the top of the list, then it must have the highest marginal gain of all nodes. If the marginal

gain of all other nodes is calculated, they would be lower than the value currently in the list (due to submodularity) and therefore the “top node” would remain on top. This process continues, finding the node that remains on top after calculating its marginal spread, and then adding it to the seed set. By avoiding calculating the spread for many nodes, the CELF model turns out to be much faster than the Greedy model.

Algorithm of CELF Model

Input: Graph object (g), number of seed nodes (k), propagation probability (p), and the number of Monte-Carlo simulations (mc).

Output: Optimal seed set, resulting spread, time for each iteration.

Initialization: Set the current system time to start_time.

start_time = time.time()

marg_gain = [IC(g, [node], p, mc) for node in range(g.vcount())]

Q = sorted(zip(range(g.vcount()), marg_gain), key = lambda x: x[1], reverse = True)

S = [Q[0][0]]

spread = [Q[0][1]]

SPREAD = [Q[0][1]]

Q = Q[1:]

LOOKUPS = [g.vcount()]

timelapse = [time.time() - start_time]

for each range from 0 to (k-1) **do**

Initialization: check = False and node_lookup = 0

while not check **do**

 node_lookup = node_lookup + 1

 current = Q[0][0]

 Q[0] = (current, IC(g, S+[current], p, mc) - spread)

 Q = sorted(Q, key = lambda x: x[1], reverse = True)

 check = (Q[0][0] == current)

end while

 spread = spread + Q[0][1]

 append (Q[0][0]) to S

 append (spread) to SPREAD

 append (node_lookup) to LOOKUPS

 append (time.time() - start_time) to timelapse

 Q = Q[1:]

end for

return (S, SPREAD, timelapse, LOOKUPS)

4.4 Proposed Graph-Reduce Algorithm

Finally, the problem of Influence Maximization in Opinionated Networks is proposed, where a set of seed nodes 'S' is determined for activating neighboring nodes with given information, and where the Opinion Compatibility between each set of neighbors is known. The original Adjacency List $AL[]$ and the Opinion Compatibility List $OC[]$ are both essential for representing such a graph object.

The influence propagation probability during context-aware influence maximization is directly dependent on the Opinion Compatibility between neighbors, and nodes cannot activate their neighbor in the absence of such compatibility. The below algorithm shows how an input graph 'G' is reduced to graph G' by leveraging the 'Opinion Compatibility' information.

The reduced graph can be obtained by using the new adjacency list $AL'[]$, which contains a reduced number of entries 'r' in contrast to the original list $AL[]$ with 'k' entries. This context-aware reduction in the graph can optimize seed node selection, as well as overall influence, spread over the network, in terms of computational time required as seen in the experiments next.

Algorithm for Graph Reduction

Input: Adjacency List $AL[k]$ and $OC[k]$ for Graph $G = (V, E)$.

Output: Adjacency List $AL'[r]$ for Reduced Graph $G' = (V', E')$.

Initialization: Empty List $AL'[]$.

```
for each i-th set of neighbors in  $AL[k]$  do
    if  $OC[i] = 1$  then
        append  $n_s, n_d$  to  $AL'[]$ 
    end if
end for
return  $AL'[r]$ 
```

Chapter V - Experiments & Results

The Influence Maximization problem states that for a network constituting ‘n’ nodes, a seed set ‘S’ of size k ($k < n$) is to be chosen, to maximize the number of nodes that can be influenced, as a result of an influence spreading process. The Greedy model and the CELF model for the Influence maximization problem have been used in this experimentation.

5.1 Experiment on the First Week of February 2021 Dataset:

First, the first week of February 2021 dataset is taken where the dataset contains a total of 2193 tweets. From 2193 tweets, it is observed that a total of 1661 nodes are unique and the remaining nodes are duplicates so duplicate nodes are dropped. Then the Python package matplotlib is used to visualize the main graph G , which is shown in Figure 5.1.1, Then the main graph is reduced based on the user’s sentiment by graph-reduce algorithm, which is already discussed in section 4.1. After reducing the graph, it is observed that a total of 824 nodes are unique and again the Python package matplotlib is used to visualize the reduced graph G' , which is shown in Figure 5.1.2.

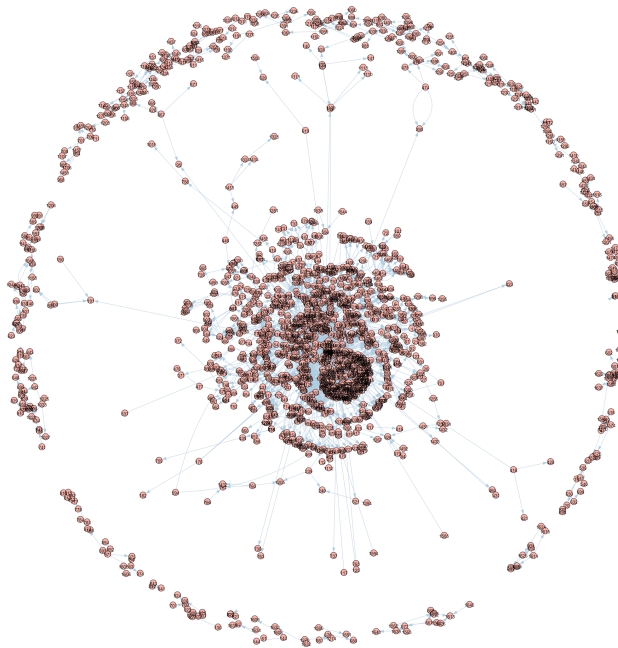


Figure 5.1.1: Feb 1st Week Main Graph

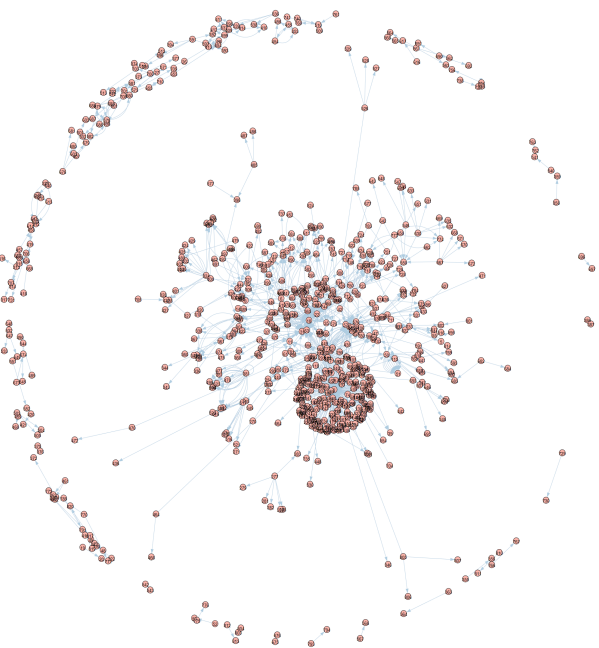


Figure 5.1.2: Feb 1st Week Reduce Graph

Now, the IM (Influence Maximization) algorithm is implemented by the Greedy method and the CELF method on the main graph and the Reduced Graph, and a total of 5 seed nodes are found. Now, the 5 seed nodes i.e., 5 influencing persons are shown in Table 5.1.

February First Week			
Main Graph (Seed / Person Name)		Reduced Graph (Seed / Person Name)	
1142	VK_bhatia2014	712	TalenganaBest1
642	TV9Bharatversa	381	TV9Bharatversa
1300	TalenganaBest1	409	prabumxian
832	SanjayJK	6	pathakBJPMP
1462	Ujjwalnetaji	330	TheUntamedFire

Table 5.1: Influencing Person Identification of February Week - 1

From Table No. 5.1, It is observed that two influencing persons (VK_bhatia2014, TV9Bharatversa) out of 5 persons are the same in the main graph and the opinion-based reduce graph.

The reason for this difference is that no particular opinion has been given priority in the first graph, but in the second graph, the graph has been created keeping only those who have the same opinion, that is why all the seed nodes of the two graphs are not the same.

It is observed that the computational time requirements of the main graph and the reduced graph with fixed seed sets for the two graphs using both greedy and CELF methods, which is shown in Figure 5.1.3.

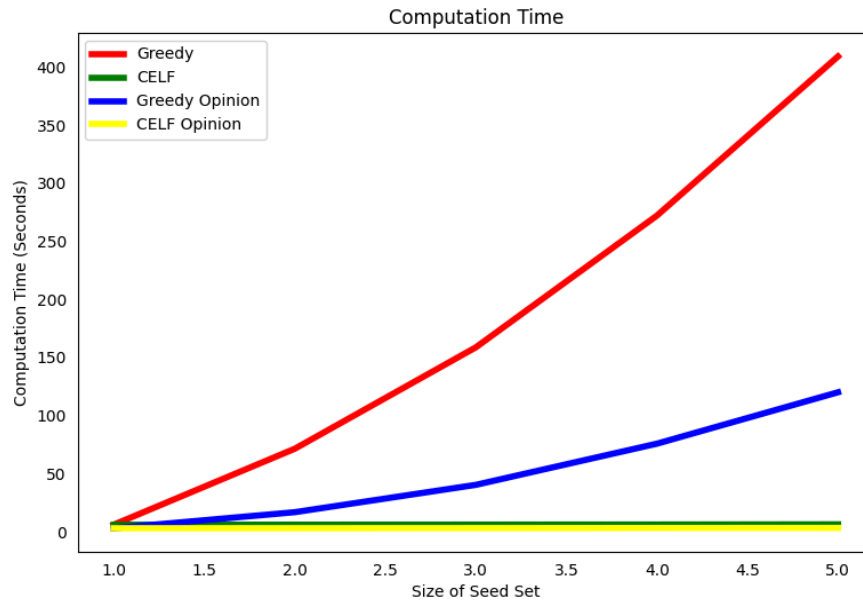


Figure 5.1.3: CELF vs Greedy Computation Time Comparison of February - 1 Dataset

Some observations that follow are discussed below:

- For the 1st graph, the determination of 5 seed nodes takes 400 seconds for the greedy algorithm, while the CELF algorithm identifies 5 seeds within 10 seconds. On the other hand, the greedy algorithm on the reduced graph takes around 100 seconds and the CELF output is obtained within 5 seconds.
- So, in contrast, using the context-aware reduced graph, the Greedy algorithm performs the same task within 100 seconds, and the CELF approach identifies all 5 seeds within merely 5 seconds, i.e. 5 to 7 times faster.

5.2 Experiment on the Second Week of February 2021 Dataset:

Now, the dataset of the second week of February 2021 is taken. The dataset contains a total of 1789 tweets. From 1789 tweets, it is observed that a total of 1707 nodes are unique and the remaining nodes are duplicates so, the duplicate nodes are dropped. Now, the Python library matplotlib is used to visualize the main graph G , which is shown in Figure 5.2.1. Now the main graph is reduced based on the user's sentiment by graph-reduce algorithm, which is already discussed in section 4.1. After reducing the graph, it is observed that a total of 785 nodes are unique and again the Python package matplotlib is used to visualize the reduced graph G' , which is shown in Figure 5.2.2.

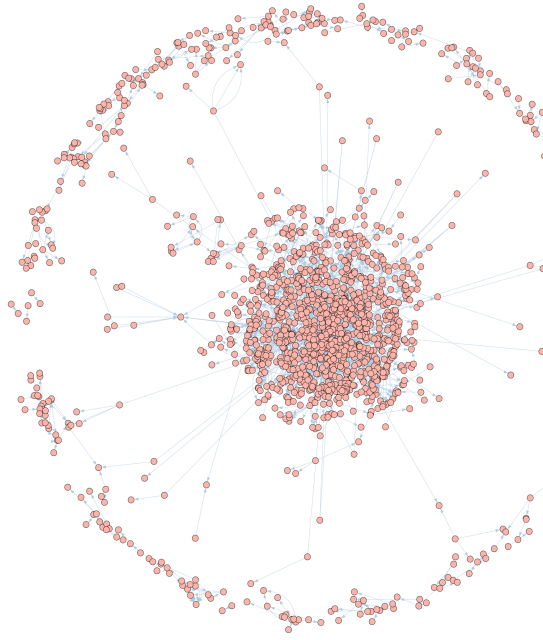


Figure 5.2.1: Feb 2nd Week Main Graph

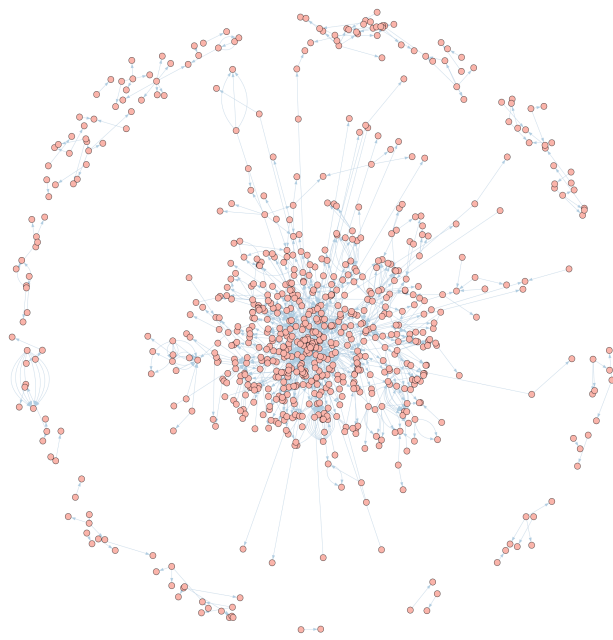


Figure 5.2.2: Feb 2nd Week Reduce Graph

Again, the IM (Influence Maximization) algorithm is implemented by the Greedy method and the CELF method on the main graph and the Reduced Graph, and a list of 5 seed nodes is found. The seed nodes i.e., the list of influencing persons are shown in Table 5.2.

February Second Week			
Main Graph (Seed / Person Name)		Reduced Graph (Seed / Person Name)	
319	mani24sjcpl	195	CNNnews18
365	kmmanoj1968	299	sushi_jivi
1270	Prashan35134700	653	SubhashMarothia
614	TV9Bharatvarsh	173	rKrishnadas510
415	CNNnews18	236	Amit69468898

Table 5.2: Influencing Persons Identification of February Week - 2

From Table 5.2, it is observed that only one influencing person (CNNnews18) out of 5 persons are the same in the main graph and the opinion-based reduce graph. The reason for this difference is that no particular opinion has been given priority in the first graph, but in the second graph, the graph has been created keeping only those who have the same opinion, that is why all the seed nodes of the two graphs are not same which is already discussed in the previous stage.

It is observed that the computational time requirements of the main graph and the reduced graph with fixed seed sets for the two graphs using both greedy and CELF methods, which is shown in Figure 5.2.3.

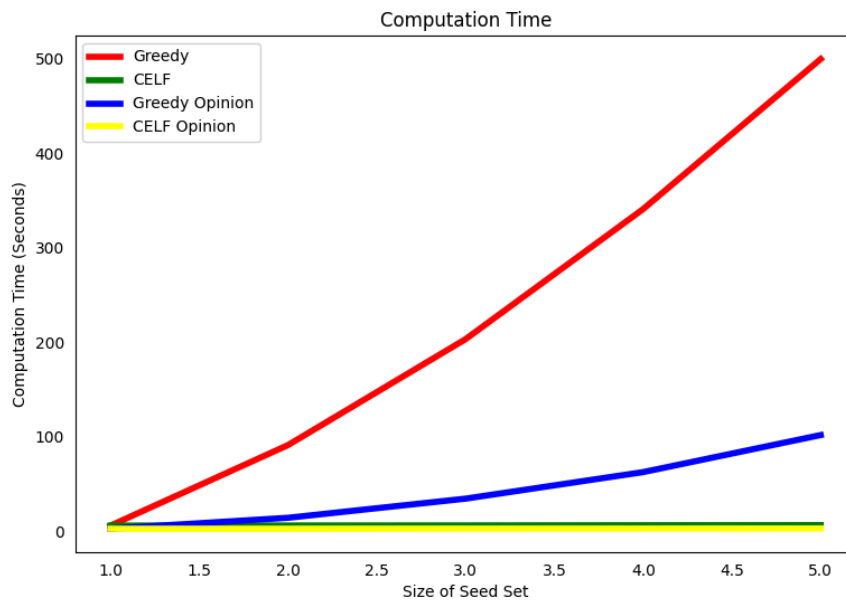


Figure 5.2.3: CELF vs Greedy Computation Time Comparison of February - 2 Dataset

Some observations that follow are discussed below:

- For the 1st graph, the determination of 5 seed nodes takes around 500 seconds for the greedy algorithm, while the CELF algorithm identifies 5 seeds within 10 seconds. On the other hand, the greedy algorithm on the reduced graph takes around 90 seconds and the CELF output is obtained within 5 seconds.
- So, in contrast, using the context-aware reduced graph, the Greedy algorithm performs the same task within 90 seconds, and the CELF approach identifies all 5 seeds within merely 5 seconds, i.e. 5 to 7 times faster than the greedy method.

5.3 Experiment on the First Week of March 2021 Dataset:

Now, the dataset of the first week of March 2021 is taken. The dataset contains a total of 5638 tweets. From 5638 tweets, it is observed that a total of 3495 nodes are unique and the remaining nodes are duplicates so, the duplicate nodes are dropped. Now, again Python package matplotlib is used to visualize the main graph G , which is shown in Figure 5.3.1. Then the main graph is reduced based on the user's sentiment by graph-reduce algorithm, which is already discussed in section 4.1.

After reducing the graph, it is observed that a total of 1621 nodes are unique and again a Python package matplotlib is used to visualize the reduced graph G' , which is shown in Figure 5.3.2.

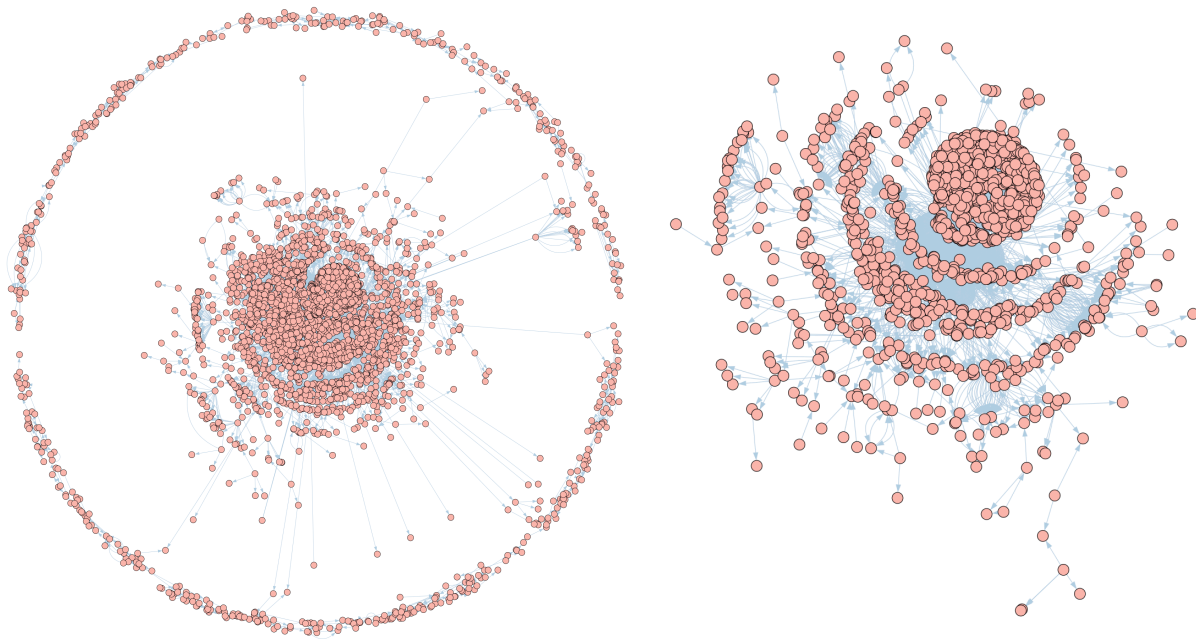


Figure 5.3.1: March 1st Week Main Graph Figure 5.3.2: March 1st Week Reduced Graph

Again, the IM algorithm is implemented by the Greedy method and the CELF method on the main graph and the Reduce Graph and a list of 5 seed nodes are found. The seed nodes i.e., the list of influencing persons are shown in Table 5.3.

March First Week			
Main Graph (Seed / Person Name)		Reduced Graph (Seed / Person Name)	
253	senthilvenk1	795	dutta70
2169	AnthSald	371	naamitabjp
919	dutta70	873	iyeraganesh
543	TV9Bharatvarsh	446	TV9Bharatvarsh
1337	souvikaitc	251	latestly

Table 5.3 Influencing Person Identification of March Week - 1

From Table: 5.3, it is observed that only one influencing person (TV9Bharatvarsh) out of 5 persons is the same in the main graph and the opinion-based reduced graph. The reason behind it is already discussed in the previous section.

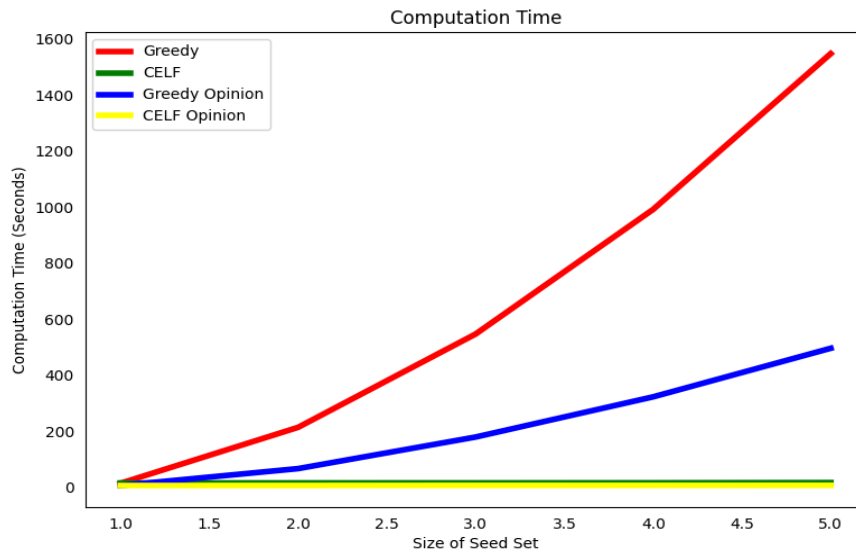


Figure 5.3.3: CELF vs Greedy Computation Time Comparison of March - 1 Dataset

All observations that are made are already discussed in the previous two stages.

5.4 Experiment on the Second Week of March 2021 Dataset:

Now, the dataset of the second week of March 2021 is taken. The dataset contains a total of 7456 tweets. From 7456 tweets, it is observed that a total of 2891 nodes are unique and the remaining nodes are duplicates so, the duplicate nodes are dropped.

Now, again Python package matplotlib is used to visualize the main graph G , which is shown in Figure 5.4.1. Then the original graph is reduced based on the user's sentiment by the graph-reduce algorithm, which is already discussed in section 4.1. After reducing the graph, it is observed that a total of 2016 nodes are unique, and again Python package matplotlib is used to visualize the reduced graph G' , which is shown in Figure 5.4.2.

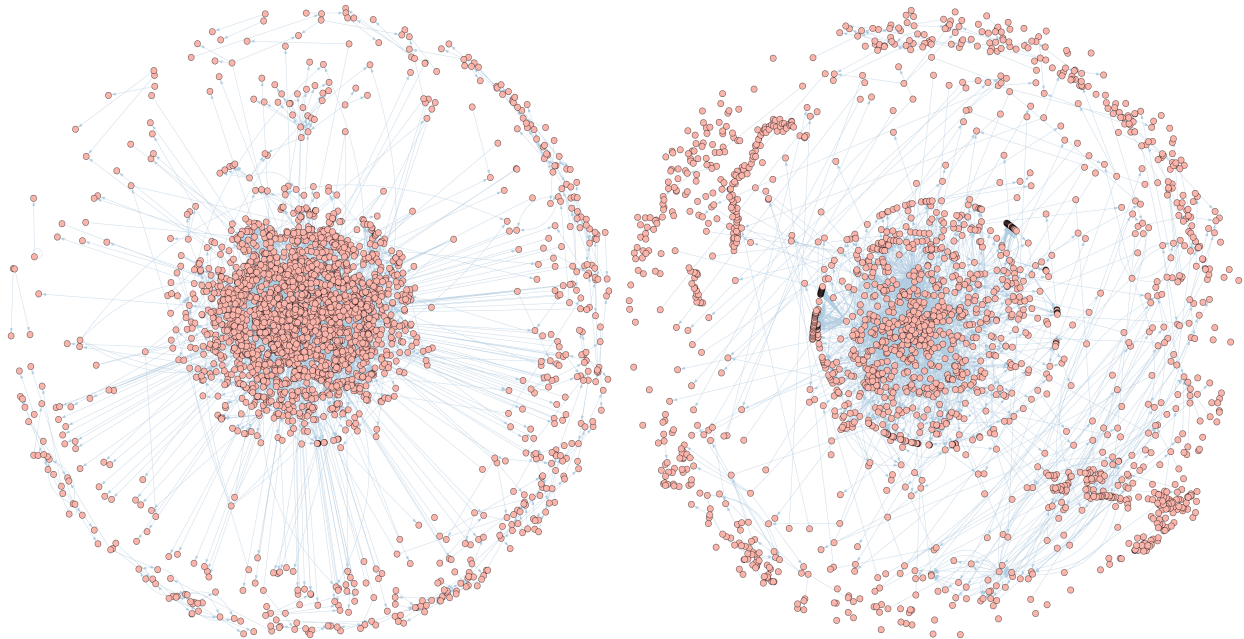


Figure 5.4.1: March 2nd Week Main Graph Figure 5.4.2: March 2nd Week Reduced Graph

Again, the IM algorithm is implemented by the Greedy method and the CELF method on the main graph and the reduced Graph, and a list of 5 seed nodes are found which is shown in Table 5.4.

March Second Week			
Main Graph (Seed / Person Name)		Reduced Graph (Seed / Person Name)	
1893	iamComutpal	1183	HarpreetKaurG18
1863	dutta70	1296	Madhumi69263117
1018	sanju1234_sanju	423	BholadasSATNAM
1763	sureshseshardi1	1057	UnSubtleDesi
2762	KolleolsBack	948	CTRavi_BJP

Table 5.4: Influencing Person Identification of March Week - 2

From Table: 5.4, here it is observed that no person is the same in the main graph and the opinion-based reduced graph. The reason behind it is already discussed in the previous stage.

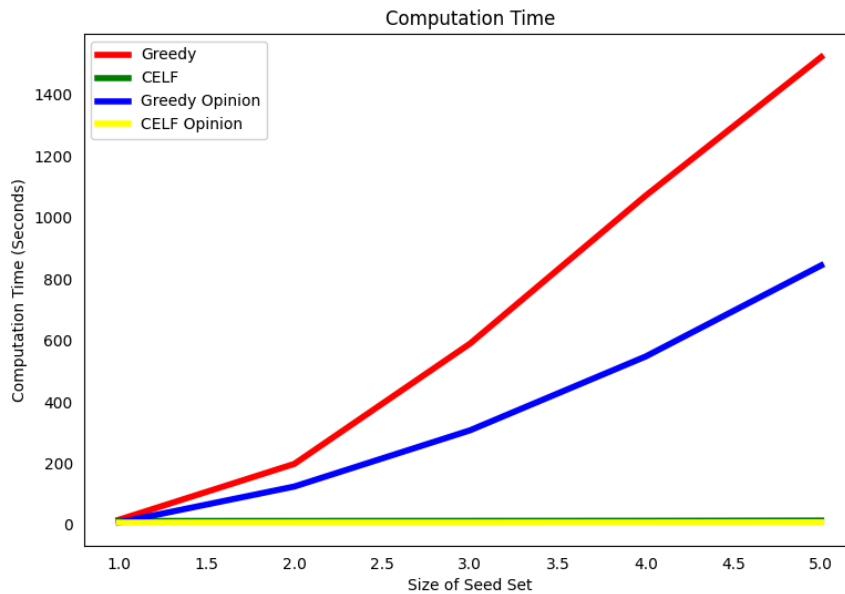


Figure: 5.4.3 CELF vs Greedy Computation Time Comparison of March - 2 Dataset

All observations that are made are already discussed in the previous two stages.

5.5 Experiment on the First Week of April 2021 Dataset:

Now, the dataset of the first week of April 2021 is taken. The dataset contains a total of 7798 tweets. From 7798 tweets, it is observed that a total of 4710 nodes are unique and the remaining nodes are duplicates so, the duplicate nodes are dropped.

Now, again Python package matplotlib is used to visualize the main graph G , which is shown in Figure 5.5.1. Then the main graph is reduced based on the user's sentiment by the graph-reduce algorithm, which is already discussed in section 4.1.

After reducing the graph, it is observed that a total of 3066 nodes are unique and again the Python package matplotlib is used to visualize the reduced graph G' , which is shown in Figure 5.5.2.

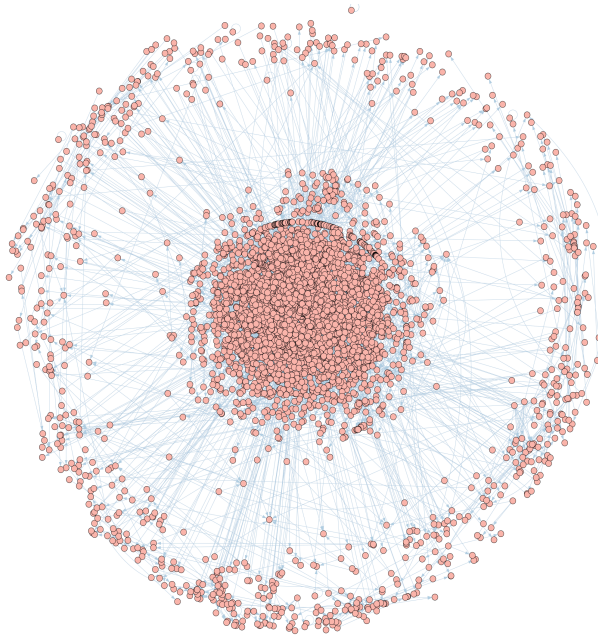


Figure: 5.5.1 April 1st Week Main Graph

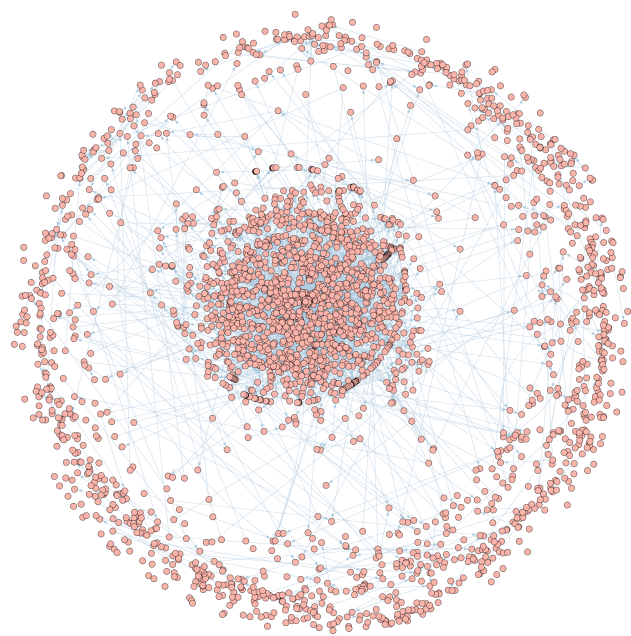


Figure: 5.5.2 April 1st Week Reduced Graph

Again, the IM algorithm is implemented by the Greedy method and the CELF method on the main graph and the reduced Graph, and a list of 5 seed nodes is found which is shown in Table 5.5.

April First Week			
Main Graph (Seed / Person Name)		Reduced Graph (Seed / Person Name)	
727	NonBhaktIndian	431	NonBhaktIndian
250	raunakarora02	2806	KolleolsBack
4453	KolleolsBack	98	raunakarora02
979	RD01989	1680	sureshseshari1
1096	RashmiShriJS	2872	NotNewToAny1

Table 5.5: Influencing Person Identification of April Week - 1

From Table: 5.5, here it is observed that three influencing persons (NonBhaktIndian, KolleolsBack, raunakarora02) out of 5 persons are the same in the main graph and the opinion-based reduced graph. The reason behind it is already discussed in the previous section.

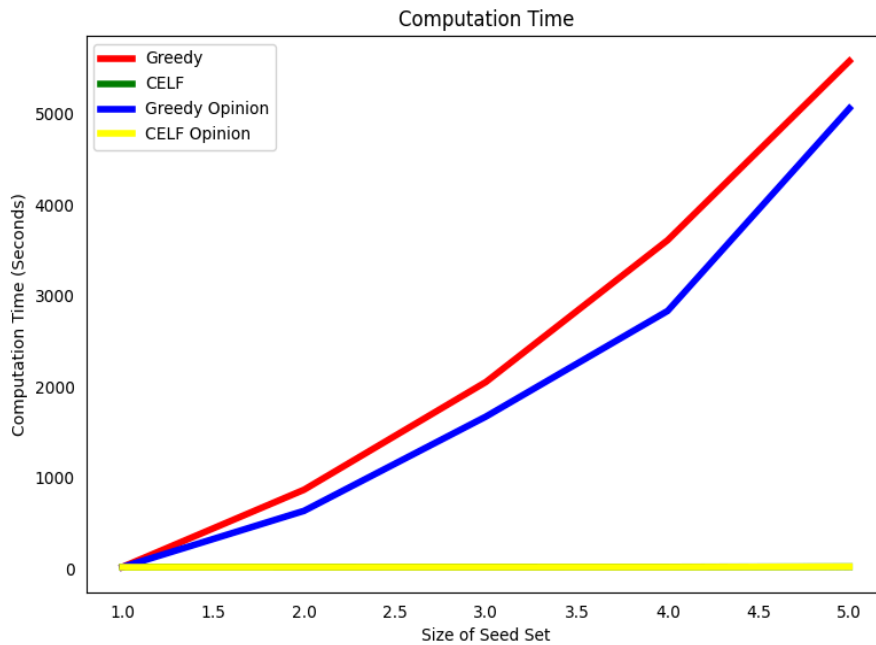


Figure 5.5.3: CELF vs Greedy Computation Time Comparison of April - 1 Dataset

All observations that are made are already discussed in the previous two stages.

5.6 Experiment on the Second Week of April 2021 Dataset:

Now, the dataset of the second week of April 2021 is taken. The dataset contains a total of 7195 tweets. From 7195 tweets, it is observed that a total of 6209 nodes are unique and the remaining nodes are duplicates so, the duplicate nodes are dropped.

Now, again the Python package matplotlib is used to visualize the main graph G , which is shown in Figure 5.6.1. Then the main graph is reduced based on the user's sentiment by the graph-reduce algorithm, which is already discussed in section 4.1.

After reducing the graph, it is observed that a total of 2744 nodes are unique, and again Python library matplotlib is used to visualize the reduced graph G' , which is shown in Figure 5.6.2.

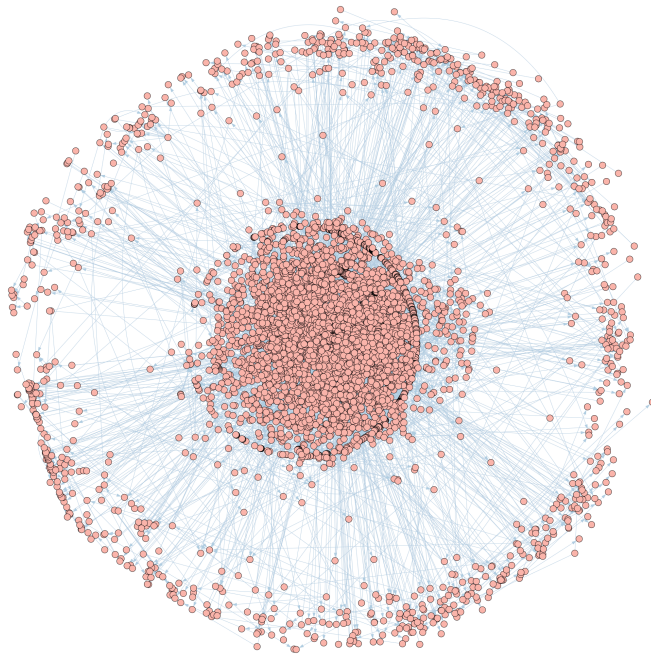


Figure: 5.6.1 April 2nd Week Main Graph

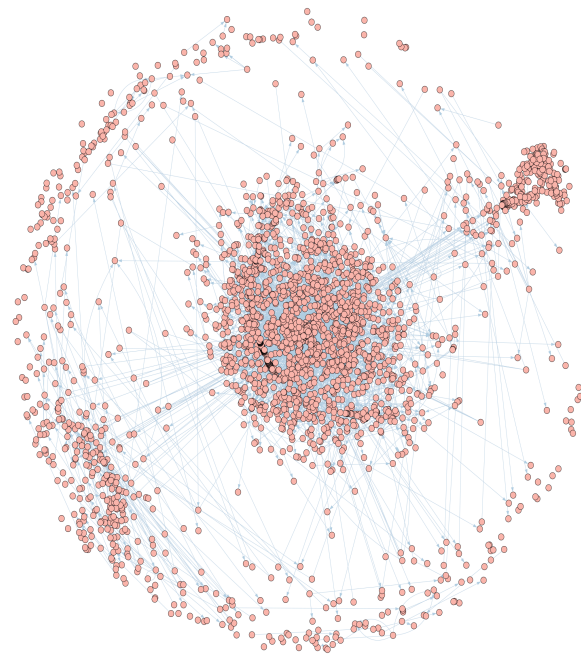


Figure 5.6.2: April 2nd Week Reduced Graph

Again, the IM algorithm is implemented by the Greedy method and the CELF method on the main graph and the reduced Graph, and a list of 5 seed nodes is found. All the seed nodes i.e., influencing persons are shown in Table 5.6.

April Second Week			
Main Graph (Seed / Person Name)		Reduced Graph (Seed / Person Name)	
314	bismay_inc	1764	AjitsinhJagirda
240	raunakarora02	2673	latestly
4908	Dharma2X	2354	RealChowkidar
4849	AjitsinhJagirda	969	pravingandhino1
3274	vig_anil	814	bismay_inc

Table 5.6: Influencing Person Identification of April Week - 2

From Table: 5.6, it is observed that a total of two influencing persons (AjitsinhJagirda, bismay_inc) out of 5 persons are the same in the main graph and the opinion-based reduced graph. The reason behind it is already discussed in the previous stage.

It is also observed that all the influencing persons from the previous months are different in this month, this is because of the change in opinion every month.

The computational time requirements of the main graph and the reduced graph with fixed seed sets for the two graphs using both greedy and CELF methods, which is shown in Figure 5.6.3.

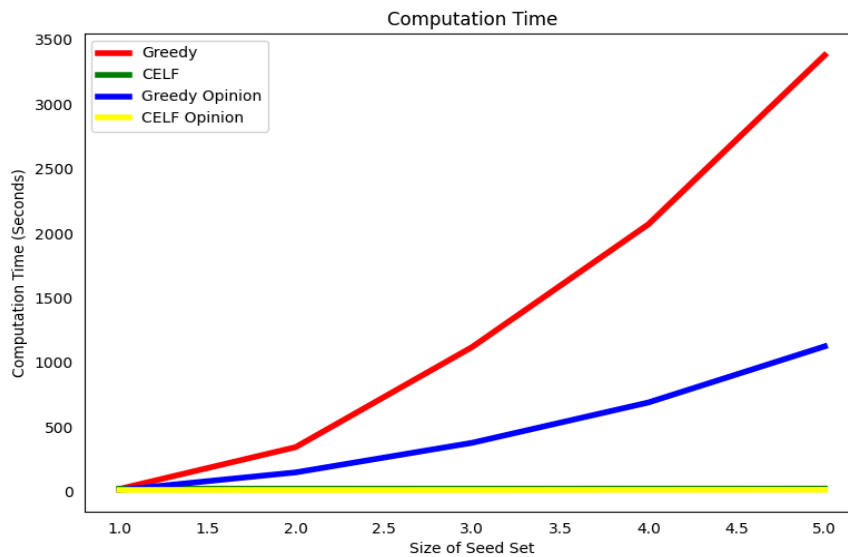


Figure 5.6.3: CELF vs Greedy Computation Time Comparison of April - 2 Dataset

All observations that are made are already discussed in the previous two stages.

5.7 Overall Observations:

Some overall observations that are made from this experiment are-

- IM with the CELF method will produce the fastest result (According to computational time) as compared to the Greedy method.
- If the size of the graph is large even if the size of the seed nodes are the same, the Greedy method's computational time grows exponentially compared to the CELF method.
- By analyzing the output of each month, it is observed that the influencing person is changing every month, the reason behind it is the prioritizing of those nodes who have the same opinion.

Chapter VI - Conclusion

Thus, the proposed context-aware graph-reduce approach results in optimized performance in the case of the influence maximization problem. It is observed that the graph-reduce technique aids in informed influence spread over the network by using the Greedy and CELF techniques. The proposed framework can be further utilized to address other network analysis challenges such as community detection, etc.

6.1 Limitations:

Opinion-based influence maximization is a fascinating area of research, especially in the context of social networks and marketing. However, it has several limitations that need to be considered:

- Models often oversimplify opinion dynamics by assuming binary opinions or uniform influence, neglecting the complexity of human behavior where influence varies among individuals and over time.
- Accurate data on social interactions for studying opinion dynamics is challenging, often reliant on synthetic or incomplete data. This limitation, coupled with privacy concerns over detailed data collection, constrains the scope and quality of available data, hindering comprehensive understanding and accurate representation of real-world scenarios.
- Influence maximization problems pose computational challenges, often necessitating approximation algorithms or heuristics that may not ensure optimal solutions.
- Static assumptions in many models overlook the dynamic nature of social networks, which constantly evolve. Additionally, these models often fail to consider adaptive behavior, where individuals adjust their actions in response to influence campaigns, highlighting a gap in understanding real-world dynamics.

6.2 Future Work:

- The current opinion-based influence maximization project is based on a specific language; in the future, it can be implemented in a multi-language scenario for more accurate results.
- In this project, ICM (Independent Cascade Model) is implemented by the Greedy Method and the CELF Method, so the computation time for finding the seed node is slightly higher. The modified greedy and the CELF++ models take very little time to find the seed nodes. So, in the future, it can be implemented by the Optimized Greedy and the CELF++ model.
- In this project, not all information is equally persuasive. The concept of "opinion strength" associated with messages and "resistance" levels for nodes has been introduced. Stronger opinions are more likely to influence others, while resistant nodes are less likely to change their minds.
- Various approaches can be used to solve the problem in the future of large undetermined nodes that result in a large size of community 4.

- **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.

- **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 affiliation 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. Conversely, 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.

REFERENCES

- [1] P. Domingos and M. Richardson, "Mining the network value of customers," in KDD 2001.
- [2] M. Richardson and P. Domingos, "Mining knowledge-sharing sites for viral marketing," in KDD 2002.
- [3] D. Kempe, J. M. Kleinberg, and E. Tardos, "Maximizing the spread of influence through a social network," in KDD 2003.
- [4] W. Chen, Y. Wang, and S. Yang, "Efficient influence maximization in social networks," in KDD 2009.
- [5] W. Chen, C. Wang, and Y. Wang, "Scalable influence maximization for prevalent viral marketing in large-scale social networks," in KDD 2010.
- [6] M. Kimura and K. Saito, "Tractable models for information diffusion in social networks," in PKDD 2006.
- [7] W. Chen, Y. Yuan, and L. Zhang, "Scalable influence maximization in social networks under the linear threshold model," in ICDM 2010. doi/10.1109/ICDM.2010.118
- [8] A. Goyal, W. Lu, and L. V. S. Lakshmanan, "SIMPAT: An Efficient Algorithm for Influence Maximization under the Linear Threshold Model," in ICDM 2011.
- [9] R. Narayanam and Y. Narahari, "A shapley value-based approach to discover influential nodes in social networks," in TASAE 2010. doi/10.1007/978-3-319-13129-0_8
- [10] J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. VanBriesen, and N. Glance, "Cost-effective outbreak detection in networks," in KDD 2007.
- [11] A. Goyal, W. Lu, and L. V. S. Lakshmanan, "CELFF++: Optimizing the Greedy Algorithm for Influence Maximization in Social Networks," in WWW 2011.
- [12] W. Chen, Y. Wang, and S. Yang, "Efficient influence maximization in social networks," in KDD 2009.
- [13] Y. Wang, G. Cong, G. Song, and K. Xie, "Community-based greedy algorithm for mining top-k influential nodes in mobile social networks," in KDD 2010.
- [14] Q. Jiang, G. Song, G. Cong, Y. Wang, W. Si, and K. Xie, "Simulated Annealing Based Influence Maximization in Social Networks," in AAAI 2012.
- [15] J. Guo, P. Zhang, C. Zhou, Y. Cao, and L. Guo, "Personalized influence maximization on social networks," in CIKM 2013.
- [16] M. G. Rodriguez and B. Scholkopf, "Influence maximization " in continuous time diffusion networks," in ICML 2012.

- [17] S. Bharathi, D. Kempe, and M. Salek, "Competitive influence maximization in social networks," in WINE 2007.
- [18] A. Borodin, Y. Filmus, and J. Oren, "Threshold models for competitive influence in social networks," in WINE 2010.
- [19] C. Budak, D. Agrawal, and A. E. Abbadi, "Limiting the spread of misinformation in social networks," in WWW 2011.
- [20] X. He, G. Song, W. Chen, and Q. Jiang, "Influence blocking maximization in social networks under the competitive linear threshold model," in SDM 2012.
- [21] N. Barbieri, F. Bonchi, and G. Manco, "Topic-aware Social Influence Propagation Models," in ICDM 2012.
- [22] F. Bonchi, "Influence propagation in social networks: A data mining perspective," IEEE Intelligent Informatics Bulletin, Vol.12, No.1, 2011.
- [23] A. Goyal, F. Bonchi, and L. V. Lakshmanan, "A data-based approach to social influence maximization," in PVLDB 2012.
- [24] A. Goyal, F. Bonchi, and L. V. Lakshmanan, "Learning influence probabilities in social networks," in WSDM 2010.
- [25] M. G. Rodriguez, J. Leskovec, and A. Krause, "Inferring networks of diffusion and influence," in KDD 2010.
- [26] K. Saito, R. Nakano, and M. Kimura, "Prediction of information diffusion probabilities for independent cascade model," in KES 2008.
- [27] J. Tang, J. Sun, C. Wang, and Z. Yang, "Social influence analysis in large-scale networks," in KDD 2009.
- [28] Amartya Chakraborty and Nandini Mukherjee, 2024. Information spreads in opinionated social networks. In 7th Joint International Conference on Data Science & Management of Data (11th ACM IKDD CODS and 29th COMAD) (CODS-COMAD 2024), January 4–7, 2024, Bangalore, India. ACM, New York, NY, USA.
- [29] 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.