FINDING INFLUENTIAL NODES IN SOCIAL NETWORKS

A dissertation

Submitted in fulfillment of the requirements for the award of the degree of

MASTER OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

Submitted by:

SANJEEV SHARMA
(2K19/CSE/20)

Under the Supervision of

MR. SANJAY KUMAR

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Bawana Road, Delhi, India-110042

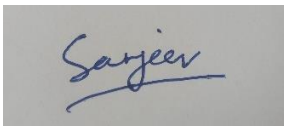
DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Bawana Road, Delhi, India-110042

CANDIDATE'S DECLARATION

I, **SANJEEV SHARMA**, Roll No. **2K19/CSE/20** pursuing M.Tech in Computer Science and Engineering, hereby declare that the project Dissertation titled **"Finding Influential Nodes in Social Networks"** which is submitted by me to the Department of Computer Science & Engineering, Delhi Technological University, Delhi is original and not duplicated from any source without proper citation in partial fulfilment of the criterion for the award of a Master of Technology degree. This work has never been used to give a degree, diploma associateship, fellowship, or any other equivalent title or recognition.



Place: Delhi **SANJEEV SHARMA**

Date: 31-08-2021 **2K19/CSE/20**

DEPARTMENT OF INFORMATION TECHNOLOGY
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi, India-110042

CERTIFICATE

I hereby certify that the Project Dissertation titled "**Finding Influential Nodes in Social Networks**" submitted by **SANJEEV SHARMA**, Roll No. **2K19/CSE/20** Computer Science and Engineering, Delhi Technological University, Delhi in partial fulfilment of the requirement for the award of a Master of Technology degree, is a record of the student's project work under my supervision. This work has never been submitted in part or in whole for a degree or diploma at this university or anywhere else, to the best of my knowledge.

Place: Delhi
Date: 31-08-2021

MR. SANJAY KUMAR
SUPERVISOR

DEPARTMENT OF INFORMATION TECHNOLOGY
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi, India-110042

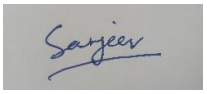
ACKNOWLEDGEMENT

Mr. Sanjay Kumar, Assistant Professor, Department of Computer Science and Engineering, Delhi Technological University, has my deepest admiration and gratitude for his wisdom, vision, expertise, guidance, enthusiastic participation, and unwavering encouragement throughout the planning and development of this research project. I particularly appreciate his thorough review and polishing of the texts, without which this project would not have been completed.

Dr. Rajni Jindal, Head of the Department of Computer Science and Engineering, deserves special recognition for providing me with all of the resources, guidance, and encouragement I needed to finish my thesis work.

Ravinder and **Usha Sharma**, my parents, for their moral support, love, encouragement, and blessings in helping me finish my endeavor.

I'd like to thank my friends **Himanshu Tiwari**, **Akhilesh Naidu**, and **Ashish Patidar**, as well as research colleagues in the department, for their support and encouragement throughout my research. In addition, I'd like to thank my friends and anyone else whose name does not appear here for assisting me directly or indirectly at all times.



Place: Delhi **SANJEEV SHARMA**
Date: 31-08-2021 **2K19/CSE/20**

ABSTRACT

We can get useful information about personal preferences, hobbies, and connections through social networks. This data could be useful in the development of recommender systems, the prediction of social influence-based outcomes, and the acquisition of knowledge. The most influential nodes in the network, also known as spreader nodes, are a strategic technique of optimizing and tracking the influence and transmission of certain information. Despite the existence of a variety of methods for identifying influential nodes in a network, recent research shows that ensuring all-round performance of selected nodes based on influence, spread, and reach is a difficult challenge. We developed a hybrid filter-based approach in which nodes are filtered based on different centrality measures and the top filtered nodes are elected as spreaders in our research. Our proposed work beats all other relevant research works when tested on a range of real-life networks across numerous judging parameters, thanks to its strategic teaming of selected spreaders and overall performance in network simulations. Another approach is also discussed where we take in the advantage of community detection and neighborhood distinctness to find out the seeds set of the social graph.

KEYWORDS

Centrality, Spreader Nodes, Influence, Social Networks, Seed set, Community Detection, Distinctness.

CONTENTS

Contents

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS, ABBREVIATIONS AND NOMENCLATURE

V = set of Nodes present in the Social Network

E = set of Edges present in the Social Network

G (V, E) = Graph of vertices attached with a set of Edges

IM = Influence Maximization

SIR = Susceptible Infected Recovered

IC = Independent Cascade

SD = Standard Deviation

CHAPTER 1

INTRODUCTION

In recent years, the analysis on social networks has evolved so much. A particular piece of information can be passed from one user to another and as there are many links between the nodes of the network, the same information can be received by a large number of users just by the ongoing process of information transmission between the adjacent nodes of the social network. But a social network can even have millions or perhaps billions of nodes, so if someone is to send a particular message to all the users by ourselves, it could be very time consuming and inefficient.

So, it would be better if small set of nodes are chosen initially, called the seed set, and let them pass the information to the major part of the remaining network. These selected nodes are also called Spreader nodes, such a set should be chosen from a large number of nodes. An approach using community detection and local structure of the nodes has been proposed to find out the seed set.

All the popular social websites like ‘Facebook’, ‘Twitter’, ‘Instagram’, ‘LinkedIn’, ‘Reddit’, etc use social networks for better user outreach, recommendations and many other use cases. Even the E-commerce websites like ‘Amazon’, ‘Flipkart’, etc use them for better product recommendations, understanding the clients’ sentiments, etc. One of the most important uses of social networks is the” Information Flow” that can happen within the network. But one cannot send the same message to all the nodes present in a social network, hence we pass the message only to the spreader nodes and let the message flow within the network via the diffusion process.

This is also called as Influence Maximization [1] because it is the influence of initially chosen seed nodes, that the same message has been transmitted to many nodes. In computer science, the process of finding out the optimal seed set, which has the maximum influence and minimum size, is termed as a NP-Hard problem as in social networks [2], there can be billions of edges between the nodes, we cannot analyse each one of those.

Influence Maximization is also used by popular companies or people to spread their desired information to most of the network users. There are two things which are most important when it comes to Influence Maximization:

- Finding out the Seed Set or Spreader Nodes
- The Information Diffusion model [3],[4] which used to simulate the process of information transmission.

The process of information diffusion can also be compared with the spread of any communicable disease, where an infected person can transmit the disease to other people who are not yet infected from the disease. This process can go on and can infect the majority of the population. In the proposed algorithm, we have used the "Susceptible Infected Recovered (SIR)" [5] model to simulate the information diffusion process. Node Centrality Measures [6] are used to rank the nodes of a social network on the basis of their importance.

More the importance, more is the chances of picking that node as one of the chosen spreader nodes from where the diffusion process shall be starting. After ordering the nodes according to their importance, we pick some top nodes from the set and make them the seed set. There are basically three types of Centrality measures available:

Local Structure based: these measures exploit the local structure around a node to find out the importance of that particular node. These have lesser time complexities than other measures as we only examine the neighborhood of a node. They can keep track of the local topologies, degrees, paths going from some particular edge, etc.

Semi-Local Structure based: these measures not only use the local structure around a node, but also the global structure to rank the nodes according to their importance.

Global Structure based: these measures exploit the global structure of the social network to find out the importance of nodes and rank them accordingly. They can use all pairs shortest paths and use it to find the importance of some edge, which will in turn help us to find the importance of the nodes attached to that edge. These are more efficient than the local structure-based centrality measures but have high time complexity.

One more important thing which needs to be considered while finding out the seed set is that it should have the least interference or the overlapped influence effect of

spreader nodes. Hence, the spreader nodes should be chosen in a way such that they are reasonably far apart from each other so that it has maximum influence over the graph with minimum overlapping [7]. We have tried to demonstrate the spreader nodes along with other normal nodes in Fig. 1. Let's now see few of the popular Centrality Measures which are used in social networks. Few abbreviations are being mentioned in Table 1.

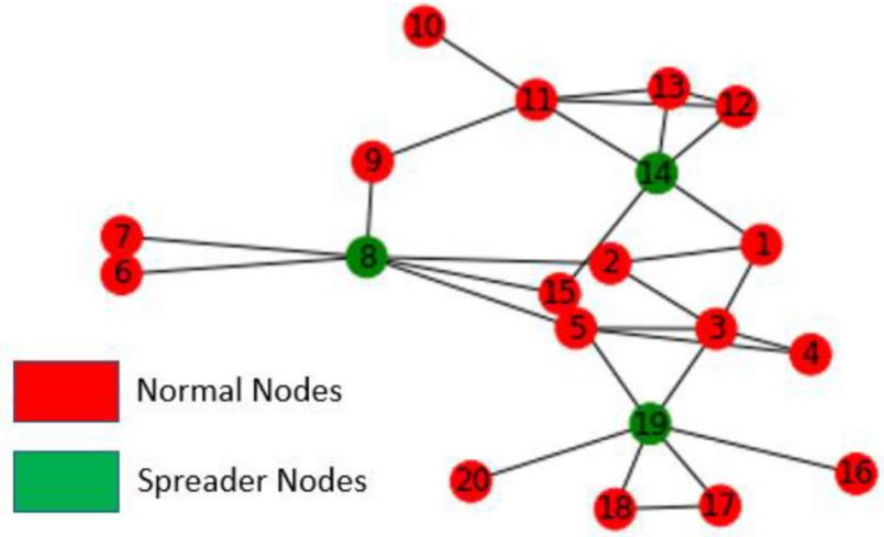


Fig. 1: Dummy Graph showing Seed set in green color

Table 1: Abbreviations used in the thesis

Serial	Variable Abbreviation	Description of the variable
1	G	The graph which represents the social network
2	V	The set of vertices in a graph, representing the users in a network
3	E	The set of edges in a graph, representing the connections in a network
4	Di	Degree of vertex 'i'
5	$e_{i,j}$ or i,j	Denotes the edge connecting vertices 'i' and 'j'
6	$eigen_i$	Eigen vector centrality of node 'i' as computed
7	CC_i	Clustering Coefficient of node 'i'

8	b_i	Between-ness of node ‘i’
9	C_i	Closeness of node ‘i’
10	$P_{j,k}$	Shortest path linking nodes ‘j’ and ‘k’
11	$Kshell_i$	K-Shell centrality of node ‘i’
12	Cr_i	Coreness centrality of node ‘i’
13	ECr_i	Extended Coreness of node ‘i’
14	PR_i	Page Rank centrality of node ‘i’
15	h_i	h-index of node ‘i’
16	Lh_i	Local-h-index or extended h-index of node ‘i’

Degree Centrality [8] is based on the degree of nodes. A node can have two types of degrees, i.e. In-degree and Out-degree. It is a local centrality measure as it takes care of only the local structure around a node [9]. More the degree of a node, more is the degree centrality. Closeness Centrality calculates the distance between a particular node and all the other nodes. It basically helps in finding out how close a node is to other vertices. It is a global structure-based centrality measure.

It has a time complexity of O(n3). Betweenness Centrality [10] also uses the global structure of the social network. It uses the concept of finding out the number of times a vertex is present between the shortest route of any two vertices. PageRank Centrality is used to sequence web pages according to their importance.

A node is considered to be important if it is connected to other important nodes. It is also used by “Google” to rank their web pages according to their importance. The web pages can be thought of as nodes of social networks and the links to other pages, contained in web pages can be considered as edges of social networks [11]. K-Shell Centrality [12] basically visualizes the network in terms of layers, the nodes which are present in the inner layers have more K-Shell Centrality and the nodes present on the outer levels or layers have less K-shells Centrality.

So, sometimes it is not able to detect the Hubs present on the peripherals of the network. H-Index Centrality [13] considers the local structure around a node; it is equal to the largest value ‘h’ such that a node has at least ‘h’ adjacent nodes and all

of them must have a degree which is greater than or equal to ‘h’. So, it performs better than Degree Centrality as it kind of takes care of a larger neighbourhood around a node.

The different centrality measures focus on different aspects of the social graph, some focus on the local structure around a node, some focus on the global structure, some applies Machine learning algorithms and some combines multiple aspects of the social graph. It is better not to restrict ourselves to a single measure and try to combine the effect of global as well as local structure around a node when we come up with a centrality measure.

Online social networks have emerged as an effective platform for sharing ideas and communicating between people. These networks have become the center of numerous research and real-life problems influence maximization, viral marketing, community detection, modeling information diffusion, fake news detection and many others [1, 2, 7].

It has led to detection, he establishment of viral-marketing, information propagation and information generation systems. Viral Marketing requires strategic selection of influential individuals who are capable of spreading information by “Word-of-Mouth” analogy, after having knowledge of the information source.

Apart from spreading of information, networks may also be used to detect and prevent epidemic propagation of information or activities by blocking access of the concerned nodes. Social Networks are represented as graphs *G (V, E)* where *G* is the graph, *V* and *E* are the set of its vertices and edges. The vertices denote the entities that belong to the social network and the edges connecting vertices denote a connection or link between two entities. A directed graph represents a network where two-way connection is not mandatory and links can be directional.

An undirected graph represents a network where a connection is treated to be bi-directional by default, or it is mandatory to have a bi-directional connection to create a valid link between two nodes. The node connectivity decides the influence one node plays among its neighbors. Determining a node’s influential properties has gained popularity in current research topics as top influential nodes play a vital role in information propagation in the network [14]. Selection of the important nodes of a network mainly depend on calculation of their centralities [15].

There are various types of centralities in a network and each one of them identifies a different aspect of the node's influential capabilities, based on the network geometry. Some of the noteworthy and popular centralities are discussed in the following paragraph.

Degree (D_i): Degree [8] centrality of a node 'i' is the total number of nodes, node 'i' is directly connected to. For a directed graph, there may be two types of degree-centralities: in-degree centrality, which is the total number of edges incident on node 'i' $\left(e_{j,i} \in E\right)$ and out-degree centrality, which is the total number of edges originating from node 'i' $\left(e_{i,j} \in E\right)$. Degree gives us a measure of reach, from one node to other parts of a network.

$$D_i = |e_{i,j} \in E| \tag{1}$$

Eigen Vector ($eigen_i$): It [16] is a measure of influence similar to degree centrality, but focused on the important nodes, one is connected to. Nodes connected to higher importance nodes have higher eigen-vector centrality. Eigen Vector centrality of node 'i' is proportional to the sum of the scores of all of its neighbours. The score is often based on the edge weights connecting two nodes. It can be mathematically expressed as:

$$eigen_i = \frac{1}{\lambda} \sum_{i,j \in E} x_j \tag{2}$$

The value of lambda is best chosen as the maximum eigen value of the adjacency matrix A (representing the graph).

Betweenness (b_i): Betweenness [10] centrality is the measure of the amount by which a vertex lies in the path between two connected vertices in a graph. It is given by the ratio of the number of times a node lies in-between the shortest path between every pair of vertices which are connected by a path in a graph, to the total number of shortest paths connecting a pair of nodes.

$$b_i = \sum_{j,k} \frac{P^i_{j,k}}{P_{j,k}} \tag{4}$$

Closeness (C_i): Closeness [17] is the reciprocal of the sum of the shortest distance between a given point and all other points in a graph to which it is connected by a path.

$$C_i = \frac{1}{\sum_{i,j} |P_{i,j}|} \tag{5}$$

K-Shell: K-Shell [12] method requires dividing the network into layers or shells, starting from the outermost nodes and progressing towards the innermost nodes or the core nodes. The outermost nodes or leaf nodes have lowest k-shell value and tend to have lower degrees. After removal of these nodes, the process is repeated for the next layer of nodes.

Coreness (Cr_i): Coreness is the sum of the k-shell indices of all the neighbouring nodes of the current node under consideration. It has similar significance to that of k-shell centrality but also emphasizes on the neighbouring nodes, while computing the importance of the current node.

$$Cr_i = \sum_{i,j \in E} Kshell_j \tag{6}$$

Extended Coreness (ECr_i): Extended coreness of node 'i' is the sum of coreness of all neighbouring nodes of node 'i'.

$$ECr_i = \sum_{i,j \in E} Cr_j \tag{7}$$

Page Rank (PR_i): Page rank [19, 20] algorithm is primarily used to rank web-pages based on the number of references it has from pages of high ranks, or in other words, of great importance. It can also be used for computing network centralities. Page rank is dependent upon the degree of the neighbouring nodes, the centrality of the nodes and strength of the links and its equation is given by:

$$PR_i = \propto \sum_{i,j \in E} \frac{x_j}{D_j} + D_i \tag{8}$$

H_Index (h_i): The usage of h-index originated as a publication citation metric, which evaluated the impact and usefulness of an article. H-index [21] of a node 'j' is defined by the maximum value of h for which node 'j' has at least 'h' neighbouring nodes, each with degree at least 'h'. H-index thus emphasizes on the neighbourhood of a node to determine its capability of being an influential node in the network. The maximum value for h-index is equal to the maximum degree of a neighbouring node.

$$h_i = \max\big(min\big(D(j)\big) \geq h\big)\, for\ all\ i,j \in E \tag{9}$$

Extended (Local) H_Index (Lh_i): Traditional h-index computation cannot recognize small changes in a node's degree, or a particular node's overall spreading abilities. Moreover, there may be multiple nodes with similar values of h-index as a solution for such drawbacks, L-H (Local H) index was introduced [22] which is an extension of h-index. This extended h-index for a node is computed by adding the h-indexes of all neighbouring nodes to the current node's h-index.

$$Lh_i = h_i + \sum_{i,j \in E} h_j \tag{10}$$

In this work, we describe a filter based superior set spreader-node selection, by using a hybrid centrality combination of page-rank, coreness and external-h index. The contribution of our work is as follows:

• A novel approach for ensuring an all-rounded selection of spreader nodes for influence maximization.

• Surveying and detailed evaluation of various real-world graph datasets on existing algorithms as well as our work.

• Our implementation proves to have a more diverse spread of important nodes and an overall better performance than existing centralities, based on various parameters of judgement.

- It identifies spreaders with good localities, maximum coverage and minimum overlap of neighbourhood.

Furthermore, even Machine learning algorithms could be used to enhance the processing of Influence Maximization algorithms, these algorithms could use the prior information which is already present and could help in making better predictions.

CHAPTER 2

LITERATURE REVIEW

After selecting the seed set, next step is to find out the influence of that seed set over the social network. It will help to find out how many new nodes, the original information transmitted from the initially chosen spreader nodes. It will help to find out the efficiency of an algorithm by comparing the performance metric “Infection Scale” later on. Information Diffusion can be thought of as an epidemic spreading situation. The Susceptible Infected Recovered (SIR) [22] model in used in this paper for the proposed algorithm to simulate the information diffusion process. In this SIR model, there are three types of nodes which are S-susceptible, I-infected nodes and R-recovered nodes.

S-susceptible nodes: these have high chances of getting infected, or in other words, there is a high chance that the information flow will reach this node.

I-infected nodes: these are already infected or have the information which needs to be multicast to many nodes.

R-recovered nodes: these nodes had been infected but now they have recovered or in other words, they cannot spread the information any more.

Another popular information diffusion model is “Independent Cascade” (IC) [3]. In, IC model, if a node ‘u’ is already infected, then there is a probability attached to every edge associated with node u, which tells the probability of successful transmission of the information from that infected node to its adjacent nodes. In a social network, there can be millions of nodes and billions of edges, so there can be regions within the network which contain the nodes having a strong connection between them, but not with other nodes outside that region. These regions can be termed as communities. Community detection [23] also helps to select the seed set [24] in such a way that spreader nodes are a bit far apart from each other [25].

One or more nodes can be picked in the seed set from a particular reasonably sized community and that may be enough to spread the information within that particular community. It can be done for all the communities. Community detection helps to find a seed set which has minimum interference and maximum influence over the network.

It can find out the communities by Brute force method, but it will not be efficient as there can be a high number of nodes in the social network. One good method to find out the communities present in our social graph is the “Girvan Newman” [26] method which uses the Betweenness centrality measure to remove the edges one by one till we have finally got our desired number of communities.

In every Iteration, it finds out the Betweenness centrality associated with every edge, then it picks up the edge having the largest Betweenness centrality and removes it from the graph. It keeps on doing this step, until it has not got the communities. An edge having the largest value of Betweenness centrality can be removed from the graph, it will help in restructuring the graph which will have proper communities. It has been tried to demonstrate the different communities present in the graph in Fig. 2.

There are various limitations in classical centrality measures for influential node selection. Nodes with same coreness values, calculated based on k-shell, may have significantly varying influence over the network. Nodes having immediate neighbours with low k-shell centrality might have the next neighbours having higher values. These issues led to the foundation of Gravity Index [27] which is a sum of the ratio of the product of the k-shell centralities of neighbouring node to the distance between the nodes.

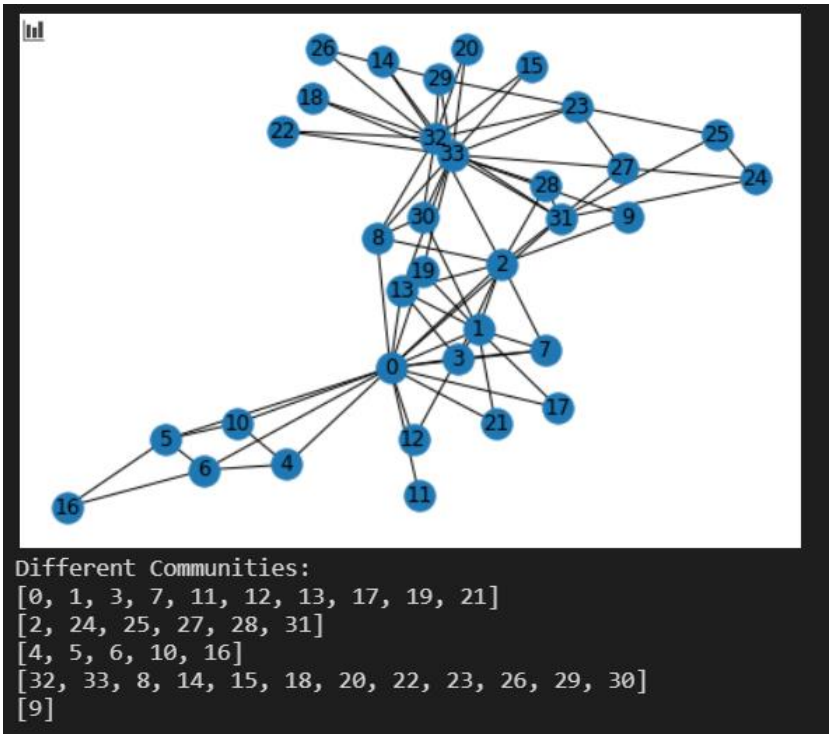


Fig. 2: Dummy Graph showing different communities present in it.

The Gravity Index and extended Gravity Index proved better at distinguishing influence of nodes in the same shell, compared to k-shell centrality index. The MINK index [28] is another influence measure introduced by Zelong Yi et al. based on k-shell centrality which aims to overcome the shortcomings of the monotonicity of k-shell. The MINK index provides efficient computation by discarding lesser important neighbour nodes which lack unique or relevant information regarding spreading capabilities.

Also, the dispersive spread of influential nodes plays an important role in influence maximization. Authors in [29] talks about heuristical clustering strategies to approach the situation and get state of the art results in influence maximization. KSC (K-Shell Community) centrality [30] was introduced as a novel idea by Qingcheng Hu et al. as a weighted metric focusing on both internal and external influence of a node, using greedy agglomerative clusters [26] for external influence calculation. Neighbourhood centrality [22] can impact a node's influence in various levels of consideration depending on the level till which neighbours are considered, and it has

varying impact, depending on the type of network chosen. Kumar et al. [9] introduced a novel method of influence maximization using the notion of modified degree centrality and mutual exclusion.

Information propagation can be carried out efficiently, by building communities in networks based on seed sets. These community-seed sets [31,32,25]. In [32], initially communities are detected inside a graph using greedy methods and then a single seed is selected based on which further seed set is expanded, with the aim of selecting the smallest, yet most effective set. Authors in [11] proposed improved WVoteRank algorithm to perform influence maximization in the weighted network by allowing the neighbors up to 2-hop in the voting process to find influential spreaders.

Our work has been inspired from [33] where an overlapping based seed set generation has been proposed. However [33] mainly focuses on community building inside networks and has been tested on small networks only. Our work uses a different algorithm than and mainly focuses on expansion within the whole network using SIR [34,35] algorithm. It demonstrates significantly better performance, compared to other centrality-based algorithms, across various relevant judgement parameters on networks of varying types and sizes.

CHAPTER 3

SOURCE OF DATA

For testing the efficiency of our algorithm, we have used the popular “Zachary’s Karate Club" [36] dataset containing 34 nodes and 78 edges as well It has an average degree of 4.5. It’s a social network within a karate club where edges represent friendships and nodes represent students of karate club at US university in the 1970s. The second dataset we have used is “Social circles: Facebook”, it contains the friend’s list of the users from Facebook. The third one is the Cond Mat Dataset. We have also showed the Zakhary’s Karate Club dataset in the Fig. 3.

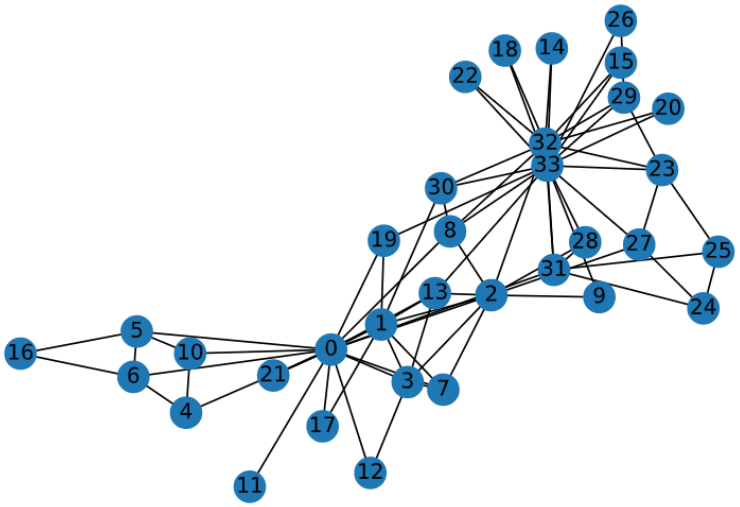


Fig. 3: Zakhary’s Karate Club Dataset demonstration.

To test Hybrid centrality algorithm’s performance, we have run it across various real-world datasets. Most of the datasets have been collected from SNAP [37] and all of them are unweighted and undirected graphs. The chosen set of graphs are quite diverse with respect to nature, sizes and scenario of usage. The description of the graphs is given in Table 2. where the given attributes provide a brief overview of the graph properties.

Table 2: Information regarding Graph Datasets for Hybrid Centrality

Graph	No. Edges	No. Nodes	Average Degree	Description
Cond Mat [38]	93497	23133	8	Condensed Matter
Facebook [39]	88234	4039	43	Facebook social circles
Enron Mail [40,24]	183831	36692	10	Communication in Enron mails

To test the performance of Distinctness algorithm, we used the datasets whose description is shown in Table 3.

Table 3: Information regarding Graph Datasets for Distinctness

Graph	No. of Edges	No. of Nodes	Average Degree
Zakhary's Karate Club	78	34	4.5
Facebook	88234	4039	43
Cond Mat	93497	23133	8

CHAPTER 4

SELECTION OF FEATURES

The performance metrics chosen in this paper focus mainly on measuring the reach, influence and variety of selected spreaders in a network.

Infection Scale

Infection scale is the fraction of the network which has been infected by the spreaders over a given period of time. The infections initiate from the spreaders and are transmitted List of concerned Centralities of every node in the Toy Network throughout the network. The infection scale increases with time and after reaching its peak, it falls down as the nodes get recovered gradually.

Recovered Nodes

A node is recovered only when it has been infected and has successfully infected its neighbours. Hence a measure of the total amount of recovered nodes, give us a sense of the fraction of network which has been able to influence the infection in the network. Recovered nodes vary with the number of spreaders initially chosen in the network. In our work, we have taken the average of all recovered nodes over 100 simulations to get a proper estimate of recovered nodes in an iteration.

Kendall's Tau Value

Kendall's tau correlation coefficient is used to check the similarity between two ranks lists. Suppose we have two rank lists R_1 and R_2 containing ranks (r_1 ,r_2 , $r_{3\ldots}$ $r_{N,}$) and (v_1 ,v_2 , v3… $v_{N,}$) for nodes (x_1 ,x_2 , $x_{3\ldots}$ $x_{N,}$). We can have N*(N-1)/2 pairings between elements in a single rank list. A pair (i, j) is said to be concordant or

matching if in R_1 $r_{i<}$ r_j implies that in R_2 $v_{i<}$ v_j. If this condition is not met, the pairs are said to be discordant. The value of Kendall's tau coefficient can range from -1 to 1, implying extreme dissimilarity and 1 implying total similarity. It is given by the formula:

$$Kdt = \frac{2*(N_c - N_d)}{N*(N-1)} \quad (12)$$

Running SIR algorithm on graph nodes individually provide us with the infection scales of individual nodes. A rank list concerned with individual spreading ability can thus be formed using SIR ranking list. The second list can be made from the ranking based on the proposed algorithm which we need to test. Then we can use Kendall's Tau to evaluate the selection of nodes by the proposed algorithm, based on the SIR rank-list.

CHAPTER 5

METHODOLOGY

Before Overlapping based spreader selection creates top-k nodes list from various centrality measures and then performs an intersection on these lists, using page, degree, clustering coefficient and eigen vector centralities to get common nodes as spreaders. While implementing we came across various limitations to this approach, especially determining optimal value of k.

The value of k varies between various networks and the intersection result hardly guarantee to provide a consistent number of spreader selection. Running the algorithm across a variety of graphs, we did not find a good selection of spreaders. However, the concept of overlapping or intersection-based centrality intuitively had potential of performing well, provided the correct parameters were chosen, with necessary changes.

Our algorithm considers multiple centrality measures to filter out most relevant spreader nodes in the network. It prepares a node filter based on multiple centrality measures and calculates a threshold for every centrality measure. We have considered the average centrality to be the threshold for every centrality. Then we filter out all graph nodes having centrality less than the threshold centrality of the corresponding list. Even if a single centrality measure fails to pass the threshold, the node is removed.

The significance of this type of filtering lies in the consideration of multiple centralities, instead of a single one, thus providing a variety of selection criteria to the process. For the selection criteria, we have chosen the following centrality measures:

Page Rank**:** Page Rank has proven to be one of the best performing centrality measures in social networks with consistently high performance in determining the most important nodes, across a variety of networks.

Extended H Index**:** Extended H-Index provides us a lower bound of optimal performance when it comes to selecting a node and its neighbours. Hence it serves as a great filtering criterion to segregate capable nodes from less influential ones.

Coreness**:** Coreness provides the neighbourhood location information inside a network as it depends directly on the shell measures.

Thus, all three centrality measures cover up almost all relevant characteristics of a node, which are required for spreader election. After the filter is applied based on these three centrality measures, we calculate a combined centrality measure of the selected nodes who have met the selection criteria. The combined measure is a sum of the normalized version of the same three centrality measures.

The normalization is required as the values of different centrality measures may have scaling issues due to greatly varying magnitudes. For normalization we have used the standard scores. After combined centrality has been calculated, the nodes are sorted in descending order of the centrality and the top-k nodes are selected as most influential spreaders for the network. In most cases, the value of k is chosen as 0.01.

There are various methods of selecting the top k nodes. The simplest way is to sort the list. However, to ensure maximum location diversity in spreader locality, is to choose the top node as the first spreader and then disable its neighbours (which will get infected by that spreader anyway) to avoid overlap, and repeat this process till all nodes are selected.

The non-neighbouring node selection has a time complexity of $O(n^2)$ where n is the total number of nodes in the graph. In every iteration we have to mark all neighbours of the current maximum node in the list and then repeat this step till we have reached the targeted number of spreaders.

However, if normal sorted selection of spreaders is opted, the time complexity becomes O ($n * \log n$).

To evaluate our model based on the fraction of network which has been influenced by the selected spreaders, we have used the SIR(Susceptible-Infected-Recovered)

algorithm. The SIR Algorithm is a real-life simulation of how an epidemy spreads in a network. It is vastly used to simulate and measure the infection and recover rates of nodes in a network and has been used in this paper as well, to evaluate the performance of our model. The SIR algorithm starts with an initialization of a fraction of network as infected nodes.

These are the initial spreaders that are responsible for transferring the infection across the network. The neighbours of the infected nodes are said to be susceptible to infection with a probability of β, which is the infection probability. For any network, there is a threshold of β which has to be maintained, in order to properly simulate an infection. The value of β has to be greater than this threshold value. The threshold is calculated as [41]:

$$\beta_T = \frac{\sum D_i}{\sum D_i^2} \tag{11}$$

In every iteration the currently infected nodes randomly infect a fraction, β of their neighbouring nodes and hence the infected set is updated with the new nodes. After infecting a node, a node is marked as recovered and becomes immune to further infection, thus ensuring that repeated infections are not made on the same node. All of the described process takes place in 1 simulation. We have run the algorithm on 100 simulations of SIR and taken the average performance of the nodes, to ensure almost every node is getting equal consideration despite the random selections. The flow of algorithm is described the Fig. 4.

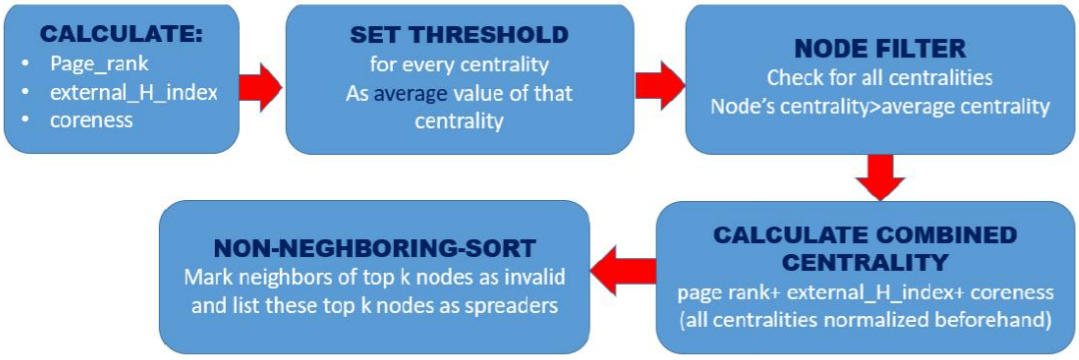


Fig. 4: The flow diagram of Hybrid Centrality

A. Algorithm for Hybrid Centrallity

Takes in the graph and outputs a map from nodes to their additive hybrid centrality score (Page Rank + Coreness + External h-index). This list only contains the filtered nodes

1. *Start*
2. *For all nodes in graph:*
   A. *Calculate External-H-Index Centrality*
   B. *Calculate Coreness Centrality*
   C. *Calculate Page Rank Centrality*
   D. *Store centralities in respective arrays*
3. *End for*
4. *Calculate average of each centrality and store them*
5. *Calculate SD of every centrality and store them*
6. *For all nodes in graph, pass them through the Hybrid Filter:*
   A. *For all calculated centralities:*

   If node centrality is less than average centrality: Discard the node

   B. *End for*
   C. *Normalize all centrality values of the node using: normalized score = (x - mean) / SD*
   D. *HybridCentrality[node] = sum of all normalized centralities of the node*

7. *End for*

   Return the HybridCentrality

   B. ***Algorithm for Non-Neighbour Spreader Selection***

   Takes in the HybridCentrality mapped list and outputs the list of spreader nodes of required length, for the given graph

8. *Start*
9. *Current_spreader_count = 0*
10. *While Current_spreader_count < number_of_spreaders:*

    A. V = node with current maximum centrality value from HybridCentrality list

    B. Add v to Spreader_list

    C. For all nodes adjacent to V in Graph:

    Mark their HybridCentrality value as -1 to discard them from being selected in next iteration

    D. End for

    E. Current_spreader_count += 1

11. *End While*
12. *Return Spreader_List*

The second proposed **Distinctness** algorithm can be used to find out the seed set for a social network using which it can have a good Influence over the network. It tries to make sure that the spreader nodes are far away from each other as it would be using community detection. It will also find out a local structure-based Centrality measure called "Distinctness Centrality" which along with a community score attached to a node, can be used to rank the nodes according to their importance. The steps are as follows:

1) First of all, community detection is performed using the "Girvan Newman" method. It used the Betweenness centrality to find out the different communities.

2) A community score is attached to each node. The idea that a node belonging to a community containing a greater number of nodes should have a higher community score.

3) Also, a Distinctness Centrality is associated with every node. It helps us to find out how many distinct nodes we can reach up to 'k' hops, if we start from the current node. Depth first Search is used to find out the Distinctness centrality, during the depth first Search, we keep on updating a 'set', containing the new nodes we have reached from the source node. It only performed the Depth first search up to K-hops.

   There can be regions where it is having nodes of more degree centrality, but actually they are strongly connected only within themselves, so it won't be able to reach to more nodes in these regions compared to the regions where it can reach to a greater number of distinct number of nodes up to k-hops, if someone starts from a source node.

   This centrality helps to rank the nodes accordingly such that the latter nodes can be used in the seed set.

4) Now, it has two scores associated with every node. i.e., community score and Distinctness score.

5) Only one or more nodes can be picked from one community which have the sum of both the scores greater than some threshold value.

6) So, community detection helped to spread the spreader nodes and Distinctness centrality helped us to find the more important nodes within a community using the local structure around that node.

7) Finally, pick top ‘X’ nodes and select them as the spreader nodes from where it will start the information diffusion process.

CHAPTER 6

RESULT

We have performed tests based on Infection Scale, Recovered Nodes and Kendall's Tau value. We compare our Hybrid Filter method with the recent state of the art centrality measures like LH-Index, Coreness, (which have already surpassed the performance of degree, between-ness, closeness and h-index centralities) as well as classical measures like k-shell, eigen vector.

The tests have been performed on real-world undirected graphs like Condensed Matter, Enron Mail and Facebook Social Circles. Each of the listed graphs have infection rate threshold β_T ranging from 0.015 to 0.1. Hence the value of β has been set between 0.06 to 0.1 for the experiments.

From Fig. 5 to Fig. 12, we plot the Infected scale of the network with respect to the time taken for the infection for all the graph datasets under consideration. The Infected scale is the fraction of the network where information has spread, starting from the initial set of spreaders. We can see that our Hybrid Filter algorithm (page rank + external h Index + coreness) outperforms all other centralities in all the datasets.

Hybrid Filter approach works especially well with Facebook social circles, Cond-Mat datasets where it beats other algorithms by a large margin. For some networks like Enron mail, the algorithm performed better with sorting-based selection, after applying hybrid filter rather than non-neighbour selection algorithm.

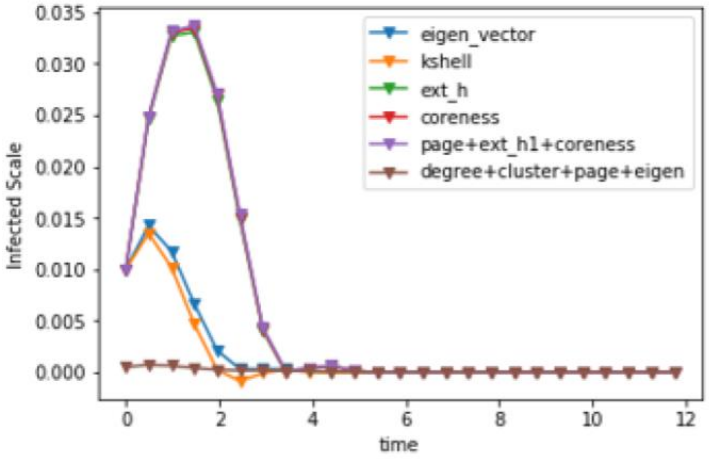


Fig. 5: Infection Scale vs Time for Amazon

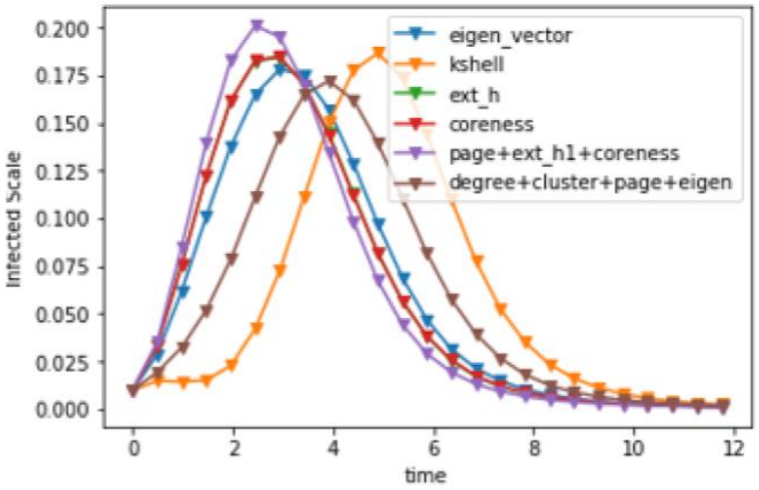


Fig. 6: Infection Scale vs Time for Astrophysics

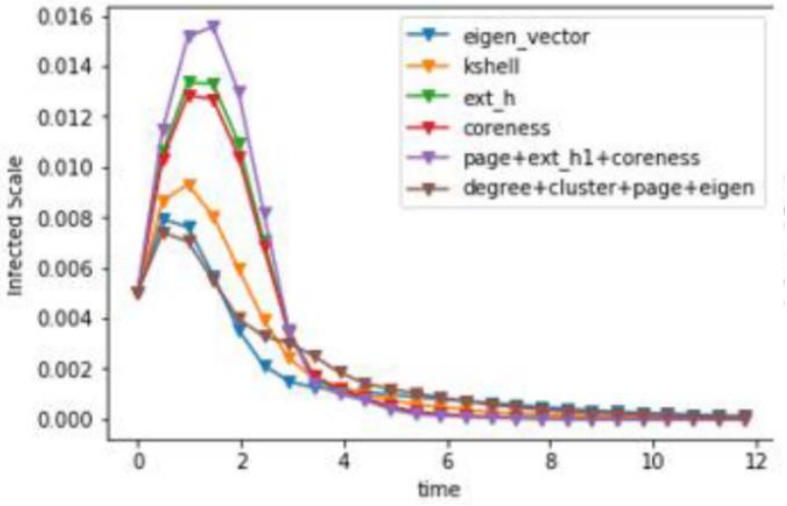


Fig. 7: Infection Scale vs Time for BrightKite

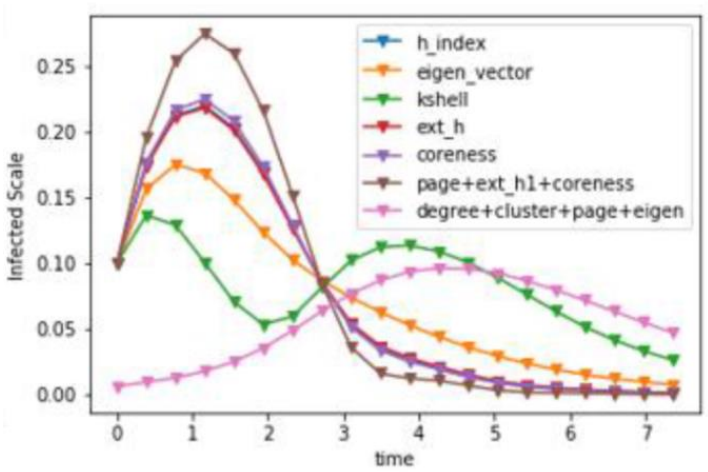


Fig. 8: Infection Scale vs Time for Cond-Mat

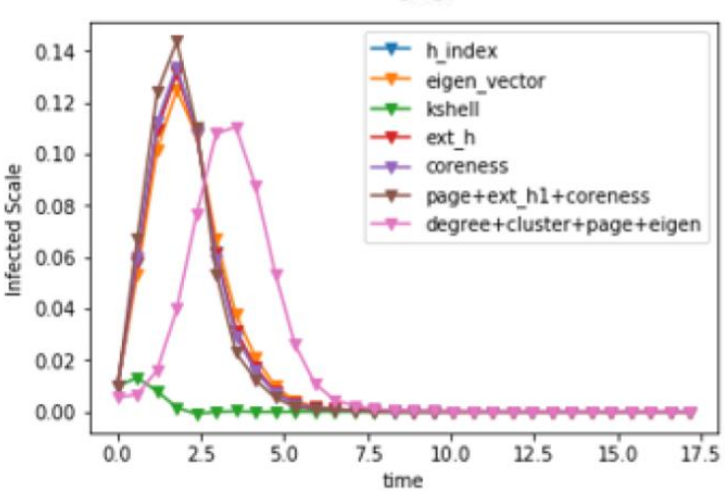


Fig. 9: Infection Scale vs Time for Enrol-Mail

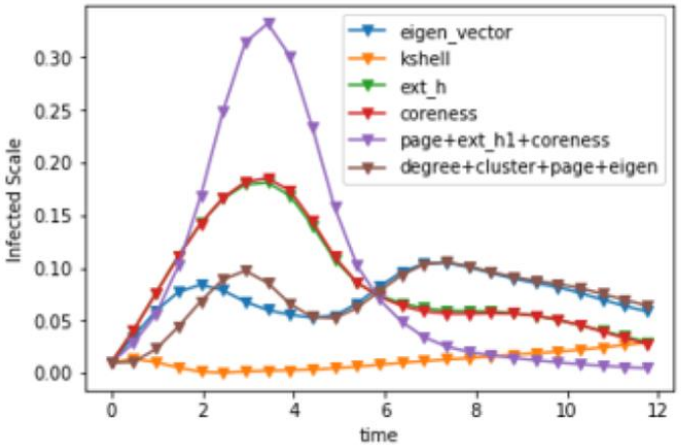


Fig. 10: Infection Scale vs Time for Facebook Social circles

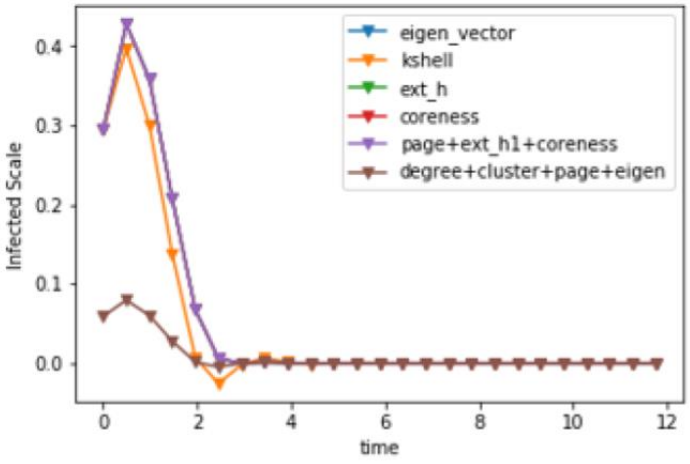


Fig. 11: Infection Scale vs Time for Karate-Club

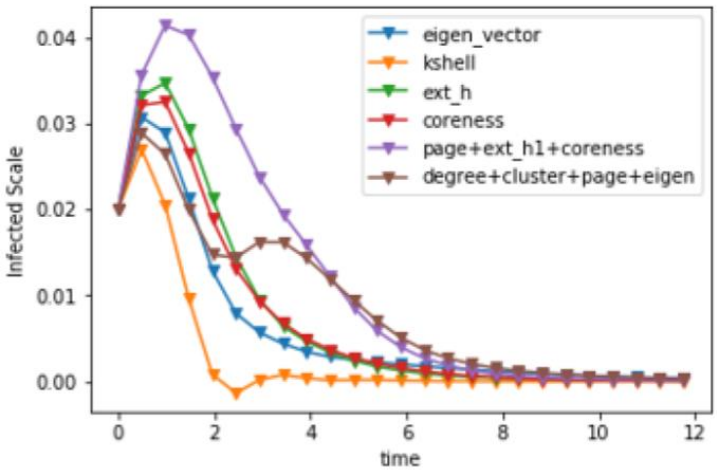


Fig. 12: Infection Scale vs Time for PGP Network

From Fig. 13 to Fig. 18, we demonstrate the algorithm's performance on the network, while changing the value of the spreaders fraction for the listed networks. Spreaders Fraction denotes the fraction of the number of nodes chosen as spreaders with respect to the total number of nodes in the entire network. We plot the total recovered nodes in a network against the initial spreaders fraction. We see that hybrid filter consistently outperforms all other algorithms, even for varying number of spreaders in every iteration, for every graph dataset.

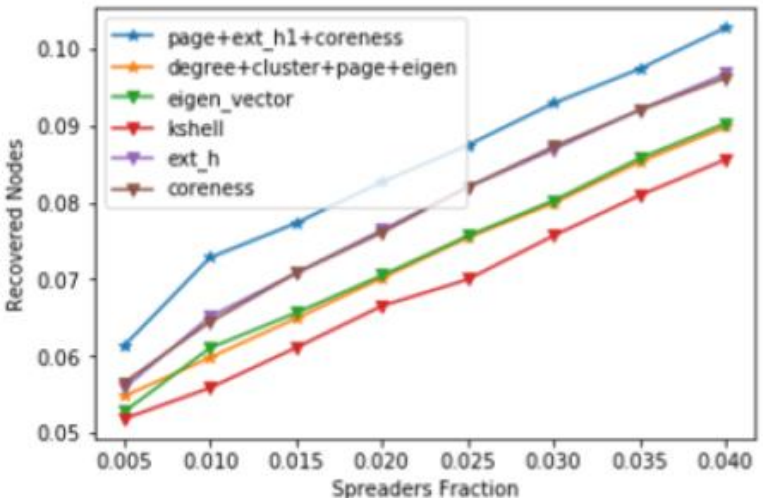


Fig. 13: Recovered Nodes vs Spreaders Fraction for AstroPhysics

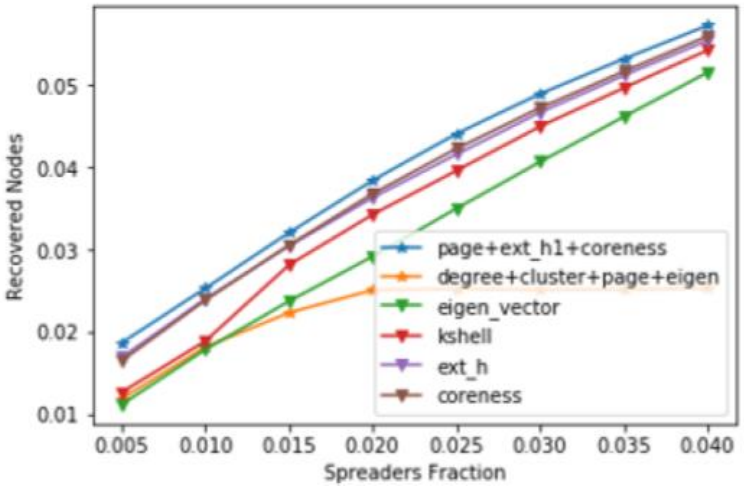


Fig. 14: Recovered Nodes vs Spreaders Fraction for BrightKite

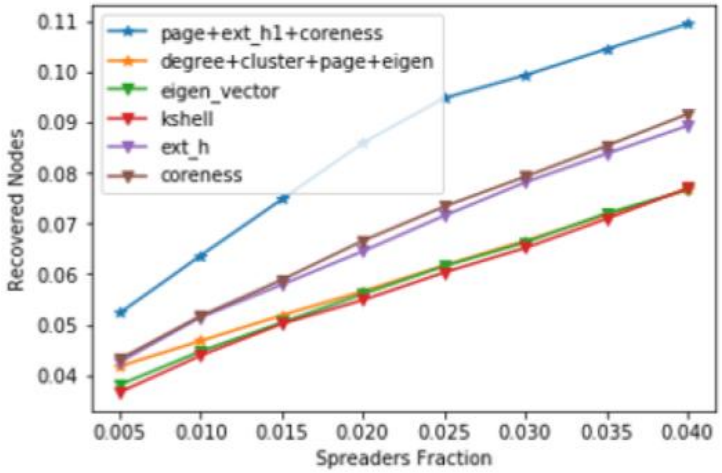


Fig. 15: Recovered Nodes vs Spreaders Fraction for Cond-Mat

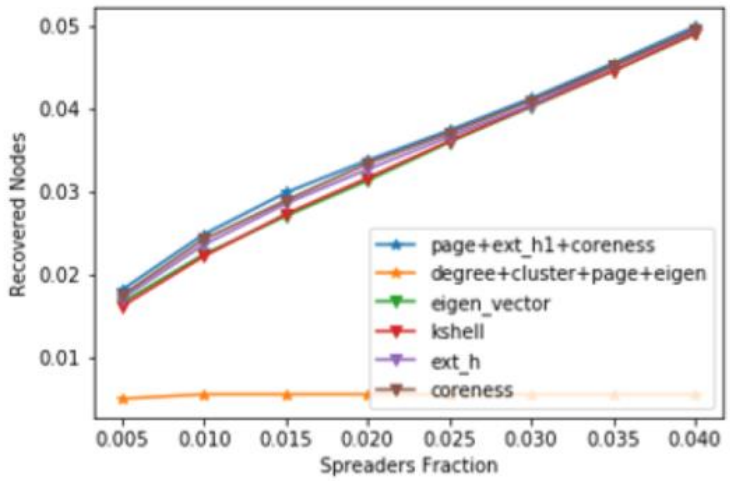


Fig. 16: Recovered Nodes vs Spreaders Fraction for Enrol-Mail

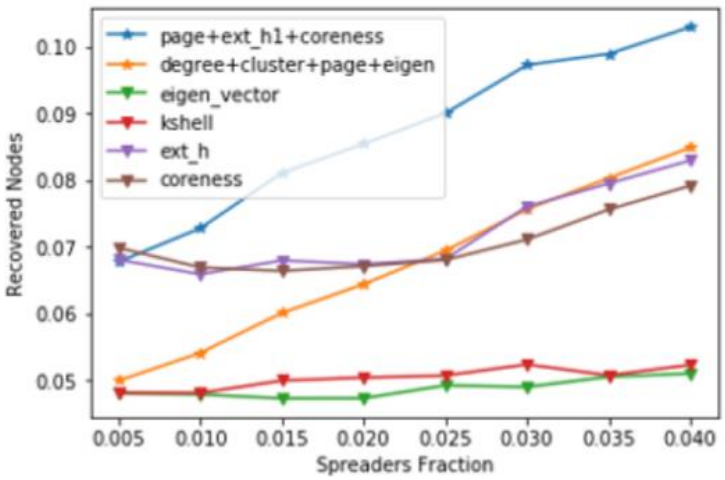


Fig. 17: Recovered Nodes vs Spreaders Fraction for Facebook Social circles

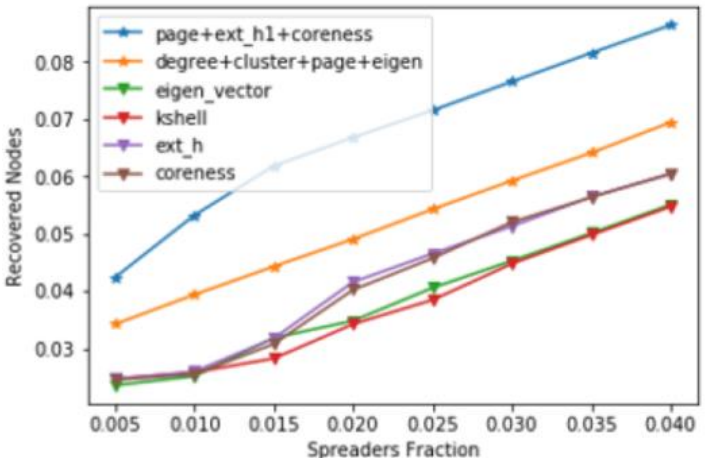


Fig. 18: Recovered Nodes vs Spreaders Fraction for PGP Network

In Table 4, we show the calculated value for Kendall's tau correlation coefficient for the same datasets. The SIR rank has been individually calculated for all nodes in the graph and the coefficient has been calculated based on the SIR ranking and hybrid filter-based ranking of nodes. The values have been calculated for β value of 0.06. We note here that our algorithm is more focused on team-based performance rather than ordering based on individual ranks. Hence hybrid filter does not always follow the ad-hoc ranking generated based on SIR ranking. Although hybrid-filter beats classical centrality measures taken separately, we find than individual correlation scores of external-h index and coreness surpass our scores by a slight margin. However, in practical scenario, our algorithm manages to surpass these algorithms, hence proving that Kendall's Tau coefficient alone, does not measure the complete capabilities of a spreader-election algorithm.

Table 4: Kendall's Tau value for different centrality algorithms corresponding to different datasets

Graph	Ours (Hybrid Filter)	Eigen Vector	K-Shell	Ext-H-Index	Coreness
Cond Mat	0.38	0.54	0.49	0.56	0.5
Facebook	0.58	0.62	0.67	0.65	0.65
Enron Mail	0.23	0.19	0.26	0.25	0.25
Karate club	0.82	0.68	0.73	0.77	0.8
Brightkite	0.41	0.24	0.49	0.49	0.49
AstroPhysics	0.021	0.025	0.047	0.014	0.020
PGP	0.49	0.49	0.49	0.56	0.54

The second proposed Distinctness algorithm has been tested based on Infection Scale and Recovered Nodes performance metrics. The Distinctness method was able to produce more infection scale and a better rate of recovered nodes per Spreaders Fraction. Fig. 19 to Fig. 21 shows the Infection Scale vs Time evaluation of the different algorithms which have mentioned in the figures.

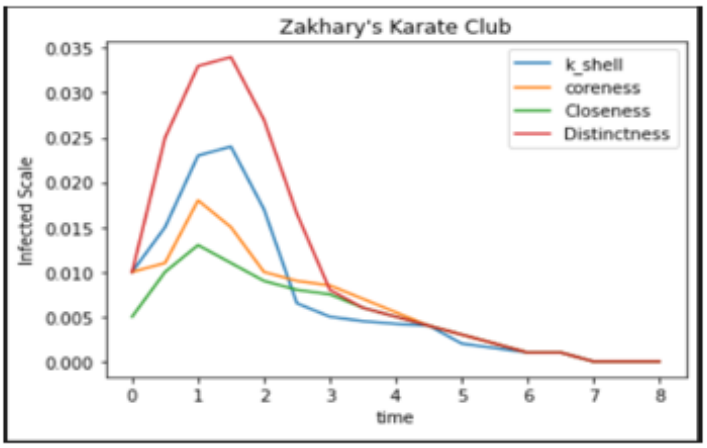


Fig. 19: Infection Scale vs Time for Zakhary's Karate Club

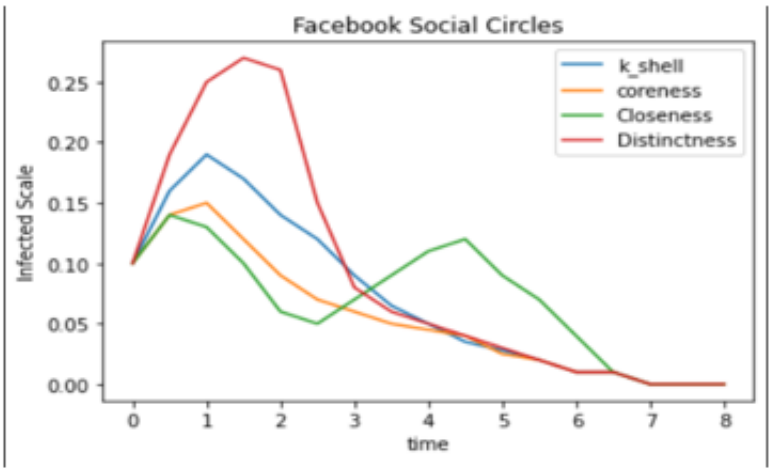


Fig. 20: Infection Scale vs Time for Facebook Social Circles

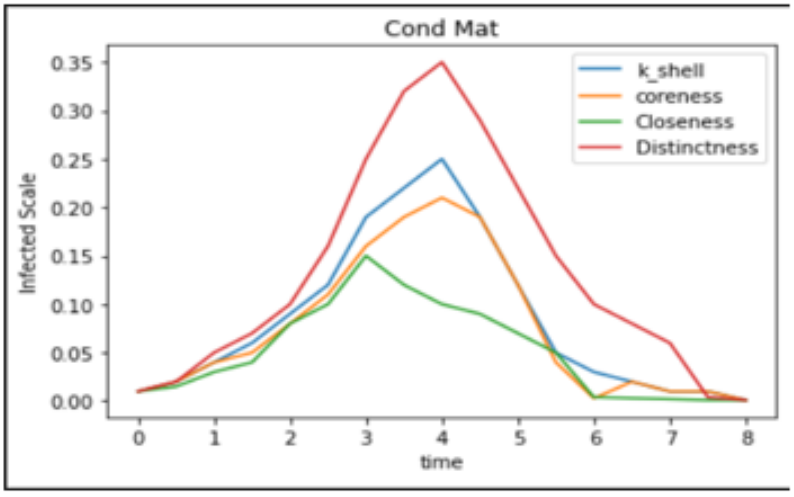


Fig. 21: Infection Scale vs Time for Cond Mat

Similarly, Fig. 22 to Fig. 24 displays the Recovered Nodes vs Spreader Fraction graphs of various algorithms. The number of nodes chosen as spreaders as a percentage of the total number of nodes in the network is known as the Spreaders Fraction. The 'Distinctness' centrality seemed to have performed better than the usual centrality measures.

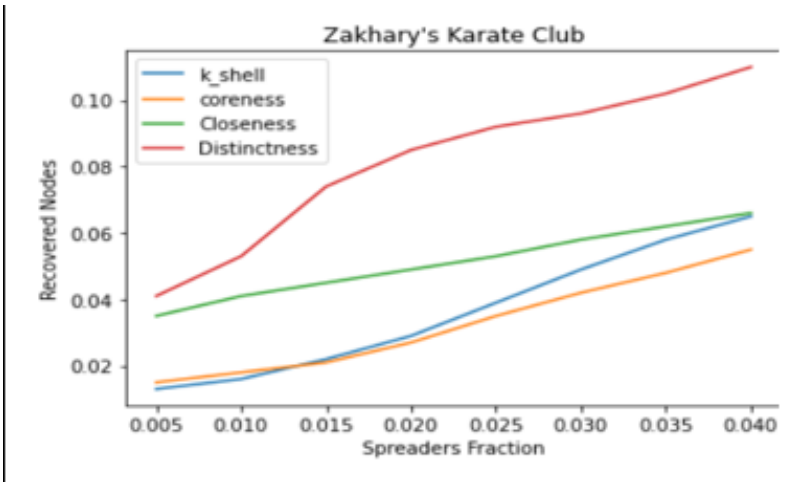


Fig. 22: Recovered Nodes vs Spreaders Fraction for Zakhary's Karate Club

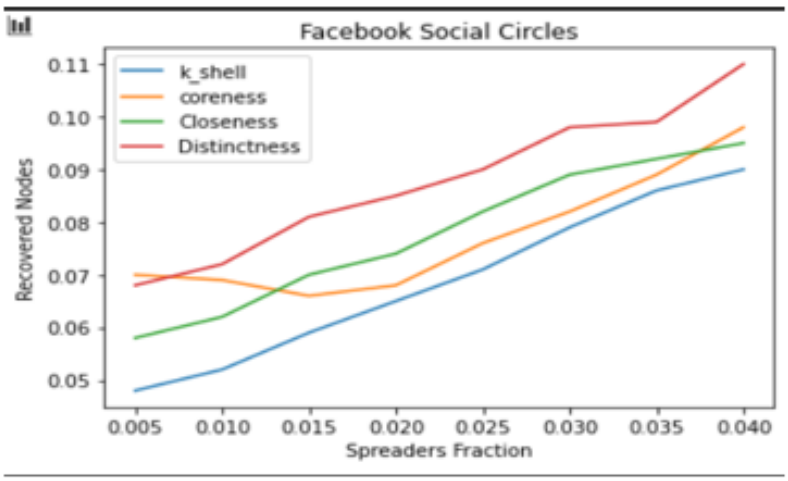


Fig. 23: Recovered Nodes vs Spreaders Fraction for Facebook Social circles

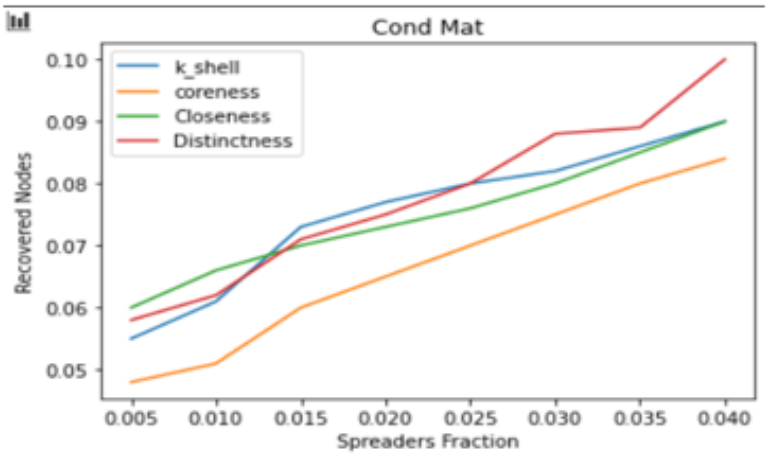


Fig. 24: Recovered Nodes vs Spreaders Fraction for Cond-Mat

CHAPTER 7

CONCLUSION

In this research work, we have proposed a novel algorithm to select spreader nodes using a hybrid centrality filter. Moreover, we have surveyed various real world, undirected, graph datasets and evaluated popular centrality measures on them. It is important to measure an algorithm based on all rounded performance, which many research works still fail to showcase. The novel approach of assigning a hybrid filter to select above average nodes as spreaders, highlight the best qualities of spreaders which are required to be a good social influencer.

Hybrid filter method has also been successful in maintaining a diverse spread of spreader nodes based on location parameter. For most cases, we have used non-neighboring spreader selection algorithm which performs on par with sorting-based algorithm. However, it has been noticed that some graphs with nodes having high average degree, perform better with sorting-based approaches

rather than the non-neighbor selection-based approach. Our work also proves that in many scenarios, nodes taking part together in an infection simulation may have different influence than if they were chosen individually and ranked based on individual performance. The non-spreader neighboring nodes of the initial set of spreaders are competent influencers too, proving that our algorithm successfully selects a good locality of influencers.

The Distinctness method performs better in the graphs which have a community relation amongst it, as it tries to select fewer nodes from a particular community. The algorithm could be enhanced for future works by introducing any other measure along with so that it could work better in the social graphs where there is no community present in the graph. Machine Learning algorithms could also be used to improve the algorithm's efficiency [39].

REFERENCES

[1] D. Kempe, J. Kleinberg, E. Tardos, “Maximizing the spread of influence through a social network” in “Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining”, Washington DC, USA, 2003, pp. 137–146.

[2] Li, Yuchen; Fan, Ju; Wang, Yanhao; Tan, Kian-Lee (2018), “Influence Maximization on Social Graphs: A Survey”. IEEE Transactions on Knowledge and Data Engineering, (), 1–1. doi: 10.1109/TKDE.2018.2807843.

[3] J. Goldenberg, B. Libai, and E. Muller, “Talk of the network: A complex systems look at the underlying process of word-of-mouth,” Marketing Letters, vol. 12, no. 3, pp. 211–223, 2001.

[4] S. Kumar, M. Saini, M. Goel, and B. S. Panda, “Modeling information diffusion in online social networks using a modified forest-fire model”. Journal of intelligent information systems, 2021, 56(2), pp.355-377.

[5] Satsuma J, Willox R, Ramani A, Grammaticos B, Carstea AS (2004), “Extending the SIR epidemic model”. Physica A: Statistical Mechanics and its Applications. 336(3-4):369-75.

[6] Kousik Das, Sovan Samanta, Madhumangal Pal, “Study on centrality measures in social networks: a survey” Social Network Analysis and Mining, December 2018 pp 8-13.

[7] S. Kumar, B.S. Panda, and D. Aggarwal, “Community detection in complex networks using network embedding and gravitational search algorithm”. Journal of Intelligent Information Systems, 2020, pp.1-22.

[8] Freeman LC, “Centrality in social networks conceptual clarification” Social Networks 1:215–239 (1978).

[9] S. Kumar, D. Lohia, D. Pratap, A. Krishna, B. S. Panda, “MDER: modified degree with exclusion ratio algorithm for influence maximisation in social networks”. Computing, 2021, pp.1-24.

[10] Freeman LC, “A set of measures of centrality based on betweenness.” Sociometry 40(1):35–41 (1977).

[11] S. Kumar, and A. Panda, “Identifying influential nodes in weighted complex networks using an improved WVoteRank approach”. Applied Intelligence, 2021, pp.1-15.

[12] M. Kitsak, et al., “Identification of influential spreaders in complex networks”, Nat. Phys. 6 (11) (2010) 888.

[13] A. Zareie, A. Sheikh Ahmadi, “EHC: Extended H-index Centrality measure for identification of users’ spreading influence in complex networks”, Physica A (2018).

[14] Kousik Das, Sovan Samanta, Madhumangal Pal “Study on centrality measures in social networks: a survey” Social Network Analysis and Mining, 2018, pp 8-13

[15] L. Lü, D. Chen, X. L. Ren, Q.M. Zhang, , T., Zhou, “Vital nodes identification in complex networks”. Physics Reports, 650, pp.1-63.

[16] Bonacich P “Factoring and weighing approaches to status scores and clique identification” J Math Sociol 2(1):113–120 (1972)

[17] Bavelas A “A mathematical model for group structures” Appl Anthropol 7:16–30 (1948)

[18] Joonhyun Bae, Sangwook Kim, “Identifying and ranking influential spreaders in complex networks by neighborhood coreness” Physica A: Statistical Mechanics and its Applications,Volume 395, 1 February 2014, Pages 549-559

[19] Page, Lawrence; Brin, Sergey; Motwani, Rajeev and Winograd, Terry, The PageRank citation ranking: Bringing order to the Web. 1999

[20] A. Langville and C. Meyer, “A survey of eigenvector methods of web information retrieval.”

[21] A. Korn, A. Schubert, A. Telcs, Lobby index in networks, Physica A388 (2009) 2221-2226.

[22] Ying Liu, Ming Tang, Tao Zhou, Younghae Do, ”Identify influential spreaders in complex networks, the role of neighborhood” Physica A: Statistical Mechanics and its Applications Volume 452, 15 June 2016, Pages 289-298.

[23] Huang H, Shen H, Meng Z, Chang H, He H (2019), “Community-based influence

maximization for viral marketing". Applied Intelligence. 49(6):2137-50.

[24] Kumar S., Hanot R. (2021) "Community Detection Algorithms in Complex Networks: A Survey. In Advances in Signal Processing and Intelligent Recognition Systems. (SIRS)". Communications in Computer and Information Science, 2020, vol 1365. Springer, Singapore.

[25] S. Kumar, L. Singhla, K. Jindal, K. Grover, B.S. Panda, "IM-ELPR: Influence maximization in social networks using label propagation-based community structure". Applied Intelligence, 2021, pp.1-19.

[26] Newman, M. E. J. (2004), "Fast algorithm for detecting community structure in networks." Physical Review E, 69(6), 066133–. doi:10.1103/physreve.69.066133.

[27] Ling-ling Ma, Chuang Ma, Hai-Feng Zhang, Bing-Hong Wang "Identifying influential spreaders in complex networks based on gravity formula" Physica A: Statistical Mechanics and its Applications Volume 451, 1 June 2016, Pages 205-212

[28] Zelong Yi, Xiaokun Wu, and Fan Li, "Ranking Spreaders in Complex Networks Based on the Most Influential Neighbors", Discrete Dynamics in Nature and Society, vol. 2018, Article ID 3649079, 6 pages, 2018

[29] Zhong-KuiBao, Jian-Guo Liu, Hai-FengZhang," Identifying multiple influential spreaders by a heuristic clustering algorithm" Physics Letters A, Volume 381, Issue 11, 18 March 2017, Pages 976-983

[30] Qingcheng Hu, Yang Gao, Pengfei Ma, Yanshen Yin,Yong Zhang, and Chunxiao Xing "A New Approach to Identify Influential Spreaders in Complex Networks" International Conference on Web-Age Information Management (WAIM 2013): Web-Age Information Management pp 99- 104

[31] Joyce Jiyoung Whang, David F. Gleich, Inderjit S. Dhillon "Overlapping Community Detection Using Seed Set Expansion" Proceedings of the 22nd ACM international conference on Information & Knowledge Management, Pages 2099-2108

[32] Xiao Li, Xiang Cheng, Sen Su, Chenna Sun: "Community-based seeds selection algorithm for location aware influence maximization" Neurocomputing 275 (2018) 1601-1613

[33] Belfin R V, E.Grace Mary Kanaga, Piotr Bródka "Overlapping Community

Detection using Superior Seed Set Selection in Social Networks",Computers & Electrical Engineering, 2018,70, pp. 1074-1083

[34] R. M. Anderson, R. M. May, "Infectious diseases in humans" Oxford University Press, Oxford, 1991.

[35] Y. Moreno, R. Pastor-Satorras, A. Vespignani, "Epidemic outbreaks in complex heterogeneous networks", Eur. Phys. J. B 26 (2002) 521

[36] https://networkx.org/documentation/stable//auto_examples/graph/plot_karate_club.html

[37] Jure Leskovec and Andrej Krevl: {SNAP Datasets}: {Stanford} Large Network Dataset Collection, Jun 2014

[38] J. Leskovec, J. Kleinberg and C. Faloutsos. "Graph Evolution: Densification and Shrinking Diameters. ACM Transactions on Knowledge Discovery from Data (ACM TKDD)", 1(1), 2007.

[39] A. Bhowmik, S. Kumar, and N. Bhat, "Eye Disease Prediction from Optical Coherence Tomography Images with Transfer Learning". Engineering Applications of Neural Networks. EANN 2019. Communications in Computer and Information

[40] J. Leskovec, K. Lang, A. Dasgupta, M. Mahoney. "Community Structure in Large Networks: Natural Cluster Sizes and the Absence of Large Well-Defined Clusters". Internet Mathematics 6(1) 29--123, 2009.

[41] Wang, W., Liu, Q. H., Zhong, L. F., Tang, M., Gao, H., & Stanley, H. E.: "Predicting the epidemic threshold of the susceptible-infected-recovered model" Scientific reports, 6, 24676, (2016).

LIST OF SCOPUS INDEXED ACCEPTANCE OF THE CANDIDATE'S WORK

[1] Sanjeev Sharma, and Sanjay Kumar. "Hybrid Centrality Filter Based Influential Spreader Selection in Social Networks" In *3rd IEEE International Conference on Advances in Computing, Communication Control and Networking–ICACCCN,* (ICAC3N-21).

[2] Sanjeev Sharma, and Sanjay Kumar. "Seed Set Selection in Social Networks using Community Detection and Neighborhood Distinctness" In *2nd Congress on Intelligent Systems*, CIS 2021.

Note: Registrations have been done for both the above mentioned papers.