Machines and Algorithms
http://www.knovell.org/mna

Research Article.

Unveiling Hidden Communities: A Graph Clustering Approach to User Interactions and Closeness

Haroon Ahmad[1], Muhammad Sanaullah[1], Muhammad Sajid[1, *], Faheem Mazhar[1], Muhammad Fuzail[2], Tauqeer Safdar Malik[3]

[1] Department of Computer Science, Air University, Islamabad, 44230, Pakistan

[2] Department of Computer Science, NFC Institute of Engineering and Technology, Multan, 59030, Pakistan

[3] Department of Information & Communication Technology, Bahauddin Zakariya University, Multan, 60800, Pakistan

[*]Corresponding Author: Muhammad Sajid. Email: msajid@aumc.edu.pk

Received: 04 December 2024; Revised: 30 December 2024; Accepted: 07 February 2025; Published: 20 March 2025

AID: 004-01-000047

Abstract: The growth of social networking sites (SNS) and the expansion of the web have facilitated easy communication among people on a single platform. A graph containing nodes and edges linking the nodes can be used to depict a social network. While the nodes represent the people or entities, the edges depict how these entities interact with one another. People who tend to associate with one another in social networks who have similar choices, tastes, and preferences form virtual clusters or communities. Finding these communities can be helpful for a variety of purposes, including locating a shared research area in cooperative networks, locating a user base for marketing and recommendation, and locating protein interaction networks in biological networks. This study presents a new way to locate communities that uses local knowledge and node space similarity. We use graph embedding to improve Community Discovery (CD) in social networks by combining eigenvector centrality and closeness measurements. Tests on six real-world datasets, including DBLP, Amazon, and Ego-Facebook, reveal that the suggested hybrid model does better than classic algorithms like Louvain, Walktrap, and Infomap. It gets a maximum NMI of 0.91 and a modularity of 0.86. These results show that the method is strong and can be used on a broad scale, making it a good way to find significant community structures in big networks.

Keywords: Clustering; Communities; Social Network; Closeness and Eigenvector Centrality; Strong and Weak Entities;

1. Introduction

As more and more of our daily activities are conducted online, there is an increasing need for social data. People can interact and voice their thoughts on goods and policies via social media platforms [1]. Therefore, everyone from heads of state to small business owners uses them as a source of information. Social media platforms make everything available to a global audience without regard to demographic limitations. People now congregate in communities and organizations to communicate and exchange information in a virtualized social environment made possible by the widespread use of social media [2]. A social network is a kind of networking that goes by this name. In the present world, a few of the most well-known ones are Instagram, LinkedIn, Facebook, Twitter, and so forth. These networks' research pushes the limits of transdisciplinary fields. The network grows more complicated every day as new linkages and contents are added without any clear definition because social media data is so different. Due to its huge data, researchers and scientists must undertake considerable amounts of data computation because of how frequently this means of communication is used [3]. Social network analysis (SNA) allows social phenomena to be studied within a particular social environment. The majority of the study is carried out with data from a small community or social networking group [4, 5].

A group of readers interested in reading publications on the same topic and age range intends to sign up for an introductory college course [6]. A well-liked technique for simulating the connections and interactions among elements or entities in actual systems is graph theory. In mathematics, a graph comprises a collection of nodes connected by a collection of edges [7]. Graph theory features are applied to understand user behavior, consumer interests, and interactions [8]. Moreover, learner interactions in social learning settings [9, 10] are characterized by graph techniques. It enables scholars to mine complex networks for valuable data while improving their understanding of these networks' basic properties and structure. It is necessary to comprehend network science and its applications to represent and evaluate the data coming from social networks [11, 12]. These nodes are referred to as leaders who are remarkably adept at building communities [13, 14]. Nowadays, the most studied topic in SNA is identifying communities and important nodes because of its applications in recommender systems [15], e-learning [16], and healthcare [17]. CD is the process of finding groups of users on the network who have similar characteristics. To determine the network's structure and functionality, community detection is utilized to extract the unique link between the nodes [18]. To achieve this, three approaches can be used: using topological features, using additional node and edge data, or merging the two [19].

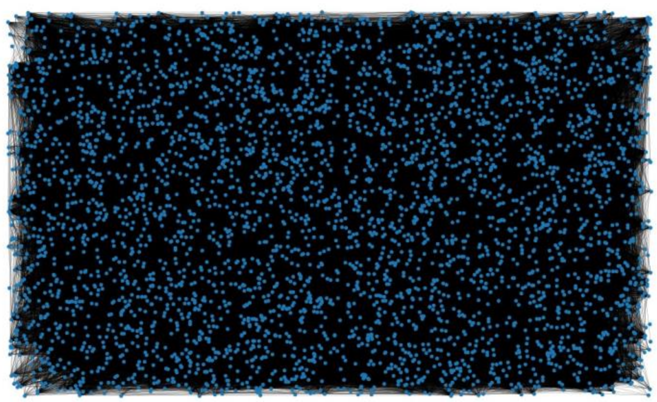

Figure 1: A Graph illustrated with Communities

The graph in Figure 1 serves as an example, displaying edges between and among different communities. As social networks, in particular, have temporal complexity and size constraints, choosing a suitable community structure is a difficult challenge. While some approaches from the previously mentioned categories may analyze largescale graphs rather quickly, they may also reveal low-quality community structures [20]. High modularity indicates that the community detection process successfully grouped the nodes into high-density, functionally well-isolated communities [21]. As seen in Figure 2, communities inside a network are identified using the proposed approach. Starting with an input network represented by an adjacency matrix, the process proceeds. Subsequently, significant nodes are determined by their huge number of connections and multiple interactions [22]. Consequently, the first stage of our concept is solely intended for node modeling in an embedding space and significance level computation. After that, the fundamental community structure is created by comparing nodes and utilizing their influence in addition to the Jaccard coefficient similarity. It has proven to be quite effective in converting high-dimensional graphs into continuous, dense, low-dimensional vector spaces [15].

Graph embedding is a highly effective approach for addressing issues in network analysis. Furthermore, the goal of graph embedding is to transform a network into a lower-dimensional vector space while preserving the network's structural properties [23]. Additionally, in a space with few dimensions, it is possible to generally depict the nodes that are close to the network by an identical vector. This simplifies duties associated with identifying and categorizing communities. The suggested model has three distinct phases. Initially, we establish an embedding space where the nodes are represented as vectors. We discover nodes with an outstanding ability to create communities and great influence over others using degree centrality metrics.

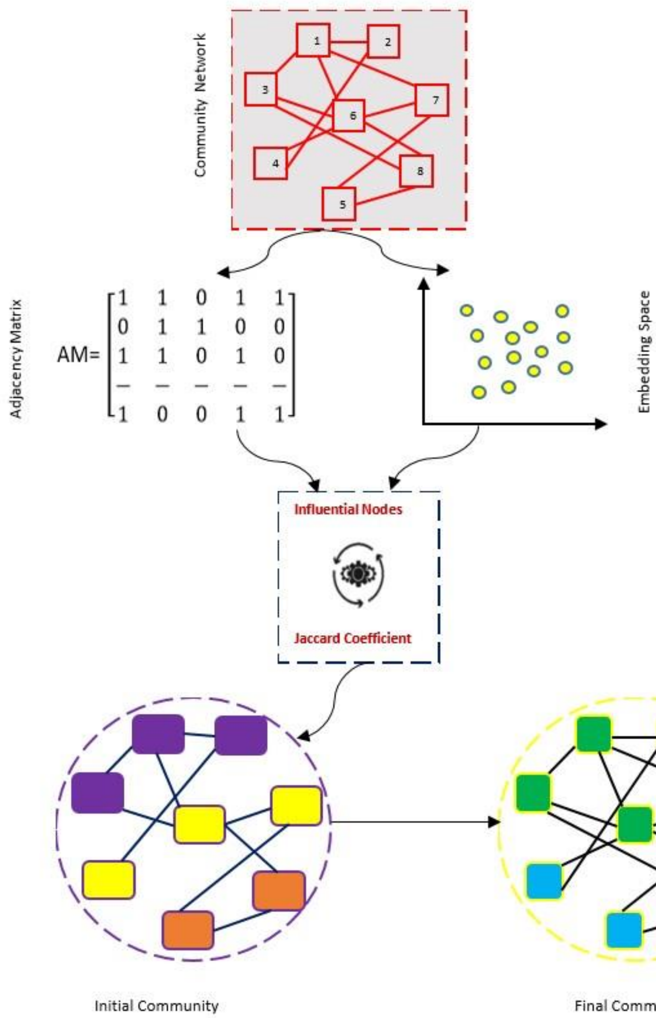

Figure 2: Proposed Hybrid Model Ideology

Next, we create an initial community structure by clustering the nodes that are most comparable to the well-known ones in the same community according to the Jaccard coefficient similarity in the embedding space. In the final step, the strong communities are united with the weak communities that were removed from the first community structure that was established in the second phase.

1.1. Significance of Research

By combining eigenvector and proximity centrality with graph embedding approaches, the proposed hybrid methodology makes a big step forward in the field of community discovery. This model integrates centrality and embedding instead of treating them as separate phases like most standard techniques do. This makes it better at showing how social relationships work. This makes it easier to find cohesive, well-defined communities, even in big, noisy networks. This level of accuracy is very important for things like targeted marketing, recommendation systems, e-learning networks, and biological community analysis. The method's robustness, scalability, and practical usefulness in finding hidden patterns in complicated networks are supported by real-world data from a variety of sources.

1.2. Motivation

This proposed approach aims to identify potential personalities that might influence social network communities. Influencer targets may include users with high eigenvector centrality to establish brand connections inside the communities. Another source of motivation was discovering the connections between various cultures. Individuals with high closeness centrality can act as a bridge between different communities, encouraging communication and the sharing of knowledge. Understanding these links might help develop strategies to encourage collaboration between groups or more effectively distribute information throughout the network. By combining graph clustering with eigenvector and proximity centrality, we may gain greater insight into the information flow inside these communities as well as the interactions between users and who impacts whom.

1.3. Research Problem

Despite the abundance of user interaction data provided by online platforms, understanding how individuals connect and build communities remains a challenge. Conventional methods for community detection might not fully capture the nuances of user behavior and information flow. This research proposal aims to develop a more comprehensive approach to identifying hidden communities inside user interaction networks.

- How many times do members of communities have with one another?
- Who are a community's main actors? Who is in a position of power?
- Can we employ a mix of eigenvector and closeness centrality to improve the accuracy and interpretability of community recognition in user interaction networks?

1.4. Contribution

The primary innovations and contributions of our work are summarized as follows:

- A brand-new, five-phase hybrid approach is recommended for the identification of social network communities. It starts at well-known nodes and spreads outward to locate communities.
- Unlike earlier techniques, our hybrid model makes use of a combination of eigenvector centralities and proximity, as well as graph statistical inference and graph embedding features.
- We present a unique centrality metric that can effectively leverage eigenvector centrality and closeness to improve community detection methods.

1.5. Organization

The following outlines how this research is arranged: A detailed Related Work is presented in section 2. Comprehensive work and discussion of the hybrid technique are explained section 4. Experiment results of a hybrid model are presented in section 6. This study is concluded in section 7 with discussions of future work.

2. Related Work

We have primarily categorized related work in five sub-sections, such as briefed in subsequent sections:

2.1. An Index of Community Detection Techniques

The topic of community detection has been the focus of numerous studies [20], and a variety of algorithms [24] are available for community detection. These methods can be broadly categorized as follows:

modularity-based methods, spectral analysis-based methods, hierarchical structures, clustering methods, random walk methods, label-propagation methods, graph-based methods, and information-theoretic measure methods [25].

2.2. Modularity-Based Group Recognition

As stated in Equation 1, the Girvan Newman algorithm [26] used modularity, a well-known standard metric, to identify the communities inside the network. Later, modularity was used as the basis for the creation of several more algorithms. These algorithms yield strong comparative findings and find extensive use in several domains, including product recommendations and research group identification [27]. The modularity metric is adjusted and connected to the spanning tree to detect the communities [28].

$$Q_m = \frac{1}{2n} \sum_{in,jn} \left[A_{injn} - \frac{Km_{in} * Km_{jn}}{2n} \right] \delta\left(C_{in}, C_{jn}\right) \tag{1}$$

A_{injn} represents the adjacency matrix between vertices in and jn. n indicates how many edges there are in the graph. C_{in} indicates the class that is associated with node i. As stated in Equation 2, the Kronecker delta is (C_{in}, C_{jn}). It equals 1 in the case when c1 equals c2 and 0 otherwise.

$$\delta\left(C_{in}, C_{jn}\right) = \begin{cases} 1 & if\ in\ and\ jn\ are\ in\ similar\ community \\ 0 & else \end{cases} \tag{2}$$

A density-based method is another approach to CD [29]; however, in this method, the algorithm receives the resolution parameter as an input. By identifying and resolving its weaknesses, the community becomes more cohesive [30]. However, while employing this strategy, the modularity and NMI performance metrics are worse for particular networks when compared to other algorithms.

2.3. Comparing Current Community Detection Techniques

Numerous approaches have been used in the past to address the community detection problem. Communities can be found using a variety of techniques, such as networks, modularity, mathematical models, and evolutionary computing. Examples of these models include fuzzy [31] logic, matrix factorization [32], and statistics [33]. Clans [34], local communities [35], and network embedding [36] are a few instances of how the network approach can be used to study. According to Louvain [37], Leiden [38, 39], Girvan Newman [26], and Greedy modularity [40], the modularity technique maximizes community quality. Evolutionary computational strategies utilize abstract concepts from biological evolutionary theory to develop optimization algorithms or methodologies. This approach integrates the principles of biological evolution with computer technologies such as particle swarm optimization [41] and genetic algorithms [42]. Nevertheless, several of the methods employed to get this utmost modularity result in sub-optimal outcomes. Furthermore, several algorithms produce groups of either significant or insignificant size that may lack practical relevance. Certain algorithms exhibit lower adaptability to network changes compared to others, particularly those that involve the addition or removal of edges or nodes. The outcomes vary when various techniques are employed to analyze a network to identify communities. Each technique yields distinct modularity and community outcomes [43].

Table 1: Review of Community Detection Algorithms

Approach	Main Highlights	Parameters	Algorithm	Time Complexity
Hafez et al. [44]	Expectation Maximization (EM), Statistical model of the interactions among participants in a social network	Directed Acyclic Graphs, EM estimates	Bayesian network statistical model	$O\ (m.k)$

Srinivas et al. [45]	Simultaneously determine the community structure and the influential nodes linked to each community.	Intra-community distance, Intercluster distance	Mathematical programming model	$O\ (d_m + k_m^o)$
Cheng et al. [46]	Invoke the FPC() and PCM() functions to execute the two stages.	The minimum and maximum number of nodes, average degree, and maximum degree.	The Node SimilarityBased Local Algorithm (NSA)	$O\ (n \log(n))$
You et al. [47]	Identification of central nodes, the spread of labels, and combining of communities	Local and the global information	Optimization algorithms	$O(n^3 + (n \log(n)) + O(n^3)$
Kasoro et al. [48]	Identify the complete set of communities that are computed by the clique percolation algorithm.	Eigenvector Centrality method	Clique percolation algorithm (CPM)	NA
Tahir et al. [49]	MCD (Mutual Community Detection) refers to the analysis of mutual connectivity inside various networks, such as U.S. airline firms and the Zachary Karate Club.	inter-connected nodes	clustering coefficient approach	NA
Bai et al. [50]	Convert an intricate network into a streamlined network, specifically a weighted tree (or forest).	Leading and following degrees	Tree-based Community Detection algorithm	$O(2n.m + n)$

2.4. Difficulties in Community Detection: Going Beyond Local Optima

There are various reasons why local optima for community detection may emerge. Because of a resolution limit, modularity-based community detection algorithms may miss small communities [51]. The technique of generalized modularity density can identify communities of various sizes and shapes by evaluating the node density within the network [52]. Modularity based on Z-scores, which standardizes the modularity score, is a further technique that can identify communities of varying sizes [53]. Another notable issue is the insufficient community infrastructure [54]. Various methods, such as disguised community identification and weak supervision, have been suggested by researchers to tackle this problem [55]. Hidden communities refer to clandestine or obscure groups that are difficult to identify using conventional community detection techniques. Another approach to community structure recognition is weak supervision, which uses the node2vec method [56] to identify communities with varying sizes and forms. Communities that have a low level of embedding also provide difficulties in identification, as stated by [57] in their study on node2vec.

2.5. Hybrid Method for Community Detection using Enhanced Modularity

A summary is provided in Table 1, along with a description of their primary contributions. Most algorithms to detect communities also use modularity and similarity measurements separately [58]. Here, the suggested hybrid approach makes use of the modularity of the network metrics, including proximity and eigenvector centrality, to enhance the final community structure. Furthermore, our method outperforms previous algorithms in terms of collaborative outcomes, NMI values, and node classification. It also demonstrates excellent modularity.

3. Prelude and Denotation

This section gives a brief description of the hybrid model that is suggested and shows an example of the graph measurements that are employed. The suggested approach's architecture design is shown in Figure 2. There are five stages to the suggested model. First, we construct an embedding space where vector representations of the nodes are found. Using degree centrality measurements, we identify nodes that have a remarkable capacity to form communities and exert significant influence over others. In addition, we use the Jaccard coefficient similarity in the embedding space to cluster nodes that are very similar to well-known nodes in the same community, so creating an initial community structure. In addition, the communities are classified into groupings that are either weak or powerful. Furthermore, the less resilient communities that were initially left out from the initial community structure produced during the s phase are merged with the more robust communities. Lastly, the final communities are detected and ranked according to modularity and NMI values.

3.1. Problem Denotation

This study depicts a community of people as an unweighted and undirected graph $G_r = (N_o,E_d)$, which is composed of a set of m_e edges $E_d \subset N_o x N_o$, where $E_d = u_i, u_j \epsilon \frac{N_o}{2}$. The nodes in the social network are their users, and the edges represent their ties or interactions with each other. In this case, our objective is to divide graph G into a collection of separate communities, ensuring that each user u_i in the neighborhood N_o is distinct inside a community. The primary goal is to identify a community arrangement in which users exhibit strong connections with other users within the same community $Ci \epsilon D$ while having weak connections with users in different communities $Cj \neq i \epsilon D$. Furthermore, our objective is to identify significant actors or leaders inside each community $Oi \epsilon D$ to improve our comprehension of the internal organization of each community.

3.2. Significance of Nodes

3.2.1. Adjacency Matrix

The matrix representing the adjacency A of graph G $G = (N,D)$ is a nxn matrix, where $N = u_1,u_2,....,u_n$ and $E = E_{ui}|(u_i,u_j)\epsilon N$. $AG = [a_i,j]1 \leq i,j \leq n$ as shown in Equation 3.

$$a_{i,j} = \begin{cases} 1 & if\ e_{u_i,u_j} \in E \\ 0 & otherwise \end{cases} \tag{3}$$

3.2.2. Degree of Node

The degree of a graph $G = (N,E)$ is the number of edges that connect a node. Equation 4 provides the algorithm for computing the degree of a collection of neighbors of a node, represented by (u):

$$Deg_G(u_i) = |\rho_G(u_i)| = \left|\{u_j \epsilon\ N \mid a_{i,j} = 1\}\right| \tag{4}$$

where $a_{i,j} = 1$ denotes the presence of an edge between u_i and u_j, and —$PG(u_i)$ is the cardinality of the collection of neighbors. Formally speaking, the degree of node $u_i \epsilon N$ with $AG = a_{i,}xn$ is given in Equation 5:

$$Deg_G(u_i) = \sum_{j=1}^{n} a_{ij} \tag{5}$$

Users who participate in a greater number of interactions than their peers may possess more influence and have more convenient access to information. Individuals with the highest level of education in the network are seen as active nodes, or hubs, capable of disseminating knowledge within a certain area of the graph. When it comes to community detection, it is essential to focus on these nodes as they are typically the most important and have a high probability of forming communities.

Table 2: Time complexity of different centrality measures

Approach	Centrality Measure	Time Complexity
Freeman et al. [59]	Degree Neighbors based Centrality	O(n)
Freeman et al. [60]	Closeness diameter based Centrality	O(n.log(n) + n.m)
Borgatti et al. [61]	Eigenvector values based Centrality	$O(n^3)$
Borgatti et al. [61]	Betweenness flow based Centrality	$O(n^3)$

3.2.3. Degree Centrality of Node

A vertex's relative importance inside the network is expressed using a simple metric known as degree centrality. To facilitate comparison, it is frequently advantageous to normalize the degree value mentioned in Equation 6. The degree centrality of node $u_i \epsilon N$ is represented as $DC_G(u_i)$ whenever there is an adjacency matrix $AG = [a_{i,}xn]$.

$$DC_G(u_i) = \frac{Deg_G(u_i)}{n-1} = \frac{1}{n-1}\sum_{j=1}^{n} a_{ij} \tag{6}$$

Eigenvector centrality, betweenness centrality, and proximity centrality are a few of them. Each statistic represents a distinct interest point. These centrality metrics have detailed explanations in Table 2.

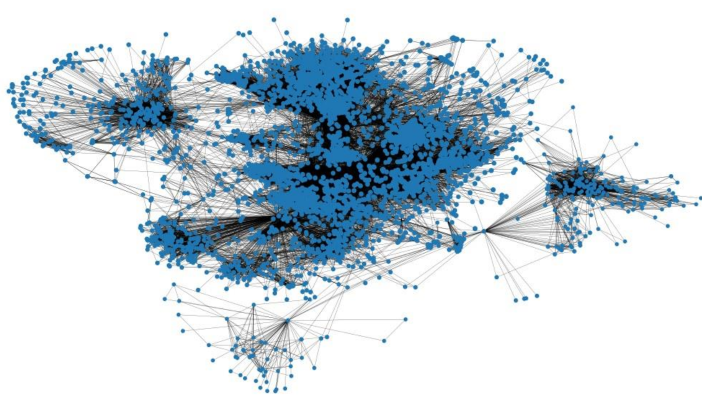

Figure 3: Community with Closeness Centrality

3.2.4. Closeness Centrality

According to [62], the closeness centrality CC_e of a node in a network is calculated by taking the reciprocal of the total length of the shortest paths that connect the node to all other nodes. This calculation may be seen in Figure 3. Equation 7 provides the estimated normalized CC_e of node j.

$$CC_e[j] = \frac{N_0 - 1}{\sum_{k=1}^{N_o} d_G(j,k)} \tag{7}$$

The value of $|V|$ is equal to N. The closeness centrality values of the nodes are often denoted as the CC_e vector when the $CC_e[j]$ values are organized into a vector of length N. Importantly, the normalized CC_e of (1) adheres to the fundamental principle of centrality, wherein a greater CC_e value signifies greater significance. However, for the sake of making things easier, we take into account the combined length of all the shortest routes, as given by Equation 8, from each given node to every other node:

$$dl[k] = \sum_{j=1}^{N_o} dl_G(k,j) \tag{8}$$

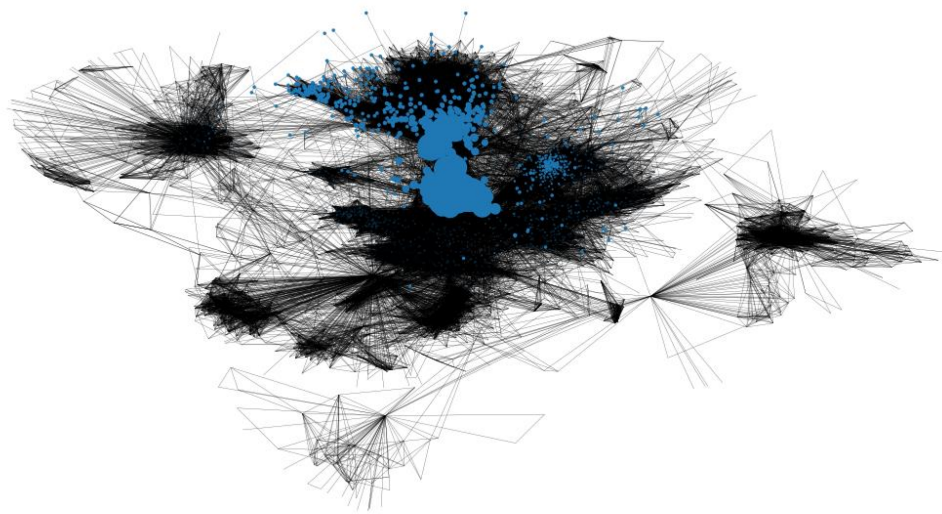

Figure 4: Community with Eigenvector Centrality

3.2.5. Eigenvector Centrality

In eigenvector centrality [63], as shown in Figure 4, the neighboring node's importance is considered in addition to the total number of neighboring nodes, whereas, in degree centrality, a node's degree centrality is simply computed by counting all of the nodes that are connected, as provided in Equation 9. In eigenvector centrality, not all connections are created equal. One's impact is usually larger in relationships with prominent people than with less influential people. Apart from its connections, the connected node's score (eigenvector centrality) is important in eigenvector centrality. Eigenvector centrality is computed by assessing a person's level of connectivity to the network's most strongly related segments. Individuals with high eigenvector analysis scores are highly connected, with many of those connections reaching to the network's conclusion. Eigenvector domination of the adjacency matrix is known as eigenvector centrality. A variant of eigenvector centrality, created by [64], is Google's PageRank. SCAN++ is predicated on the observation that real-world graphs, like web graphs, have high clustering coefficient scores [65]. Node density is determined by a node's clustering coefficient [66]. A node's clustering coefficient score rises when it and its nearby nodes get closer to a full graph, also known as a clique as shown in Figure 5. That is, it is predicted that a node and its two-hop-away node, particularly in real world graphs [67], will share a significant portion of their neighborhoods. This feature is the basis for SCAN++'s pruning of the density evaluation for shared nodes between a node and its two-hop-away node.

$$Av = \lambda v \tag{9}$$

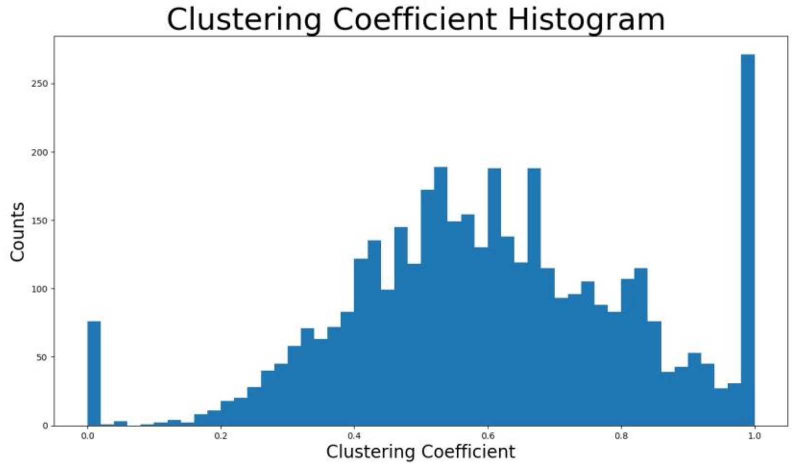

Figure 5: Clustering Coefficient

An eigenvector of a square matrix A is a non-zero vector v that, when multiplied by A, results in a constant multiple of v. This constant multiple is typically represented by the symbol λ. The eigenvalue λ corresponds to the vector v in matrix A. The individuals who possess high eigenvector centrality are the ones who have leadership positions inside the network. They are often well-known individuals with a wide network of connections to other notable figures. They frequently serve as significant thought leaders as a result. On the other hand, high betweenness and high closeness roles may not always be able to be played by people with high Eigenvector centrality. The time complexity of combining proximity centrality and eigenvector centrality for community detection can be minimized by addressing the computational bottlenecks related to each metric. We reduced the temporal complexity of the closeness centrality using the random sample technique. Instead of calculating closeness centrality for every user, think about using a random sampling technique. This requires fewer calculations and provides a good approximation of average proximity centrality inside the network. To reduce the temporal complexity for eigenvector centrality measures, we employed iterative techniques. It claims that iterative methods like the Power Method can be used to calculate eigenvector centrality. These methods may take more iterations to converge, but they are often faster than explicitly computing the eigenvectors of the adjacency matrix.

4. Proposed Ideology

We define the key terms of our proposed hybrid model and then go into great detail to explain each of the model's phases. The flow diagram for the Proposed Hybrid model is shown in Figure 6. The proposed method is composed of the following main steps:

1. Performing the extraction of influential nodes and the generation of an embedding space.
2. Determining the initial configuration of the community.
3. Choosing strong and weak communities.
4. Community final merging.
5. Community detection and ranking based on modularity and NMI values.

4.1. Nodes with Significant Influence

It enables the identification of people who are highly relevant for a range of vocations because of their ability to disseminate knowledge and information within the network rapidly. We employ algorithm 1 to extract the most influential nodes and initiate the community detection process, taking into account the degree, proximity, and eigenvector centrality measurements. The centers of the communities are these powerful nodes. Then, after placing each node in the network in descending order based on their degree centrality value, we utilize the LE technique to create the embedding space. The representation of nodes as vectors in the embedding space makes it easier to analyze the network's structure and interactions. After the formation of the embedding space, each node in the network is assigned the label Not visited indicating that they have not yet been assigned to a community. The following is a summary of the primary steps in community detection:

1. After determining the level of centrality of each node, place the nodes in decreasing order.
2. Create the embedding space using the Laplacian Eigenvectors technique.
3. Give each node a “No visited” flag.

Algorithm 1: Selection of Influential Nodes

Require: Influential Nodes and Embedding Space

Ensure: G(V,E), Dimension: d

1. To extract influential nodes, calculate the DC, using Closeness and Eigenvector centralities using Equations 7, and 9.
2. Sort the nodes in the graph in descending order.
3. Vin ⇐ influential nodes
4. Marked the nodes that are not visited.
5. Status (Vi∊Vin) =Not Visited
6. To obtain embedding space, use the LE method.
7. Embed = Laplacian Eigenmap (G, d) returns influential nodes and embedding space.

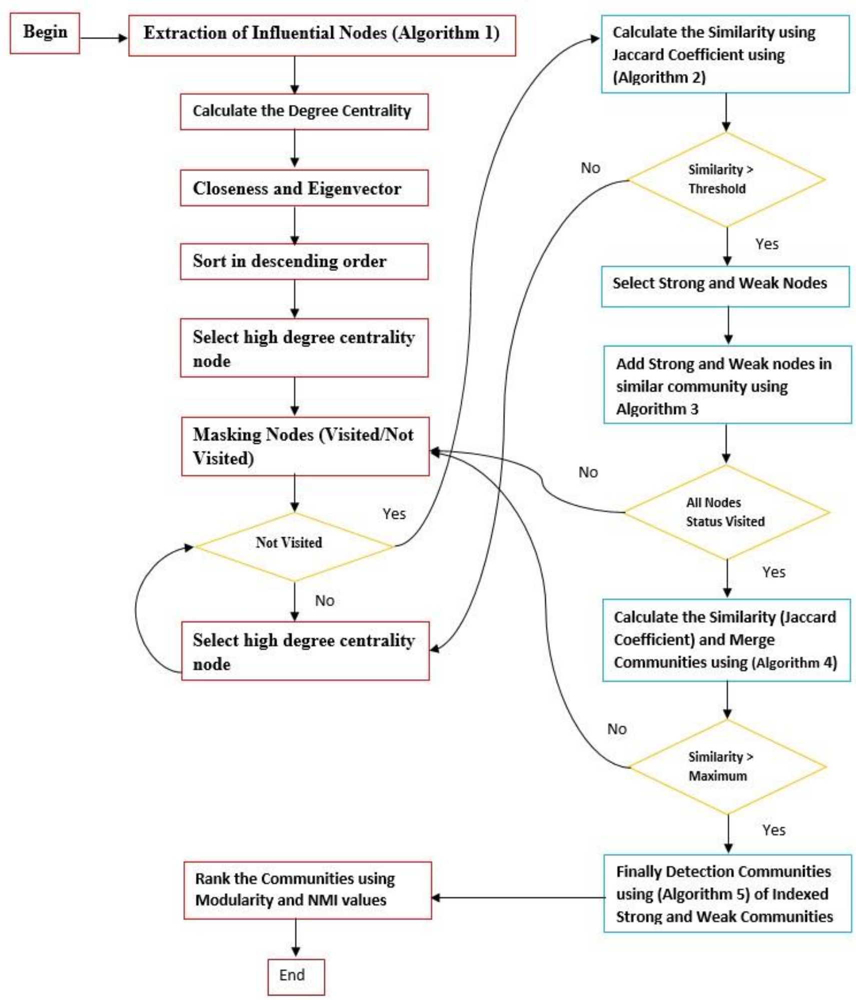

Figure 6: Flow diagram for the selection of community

4.2. Initial Community Identification

The similarity is calculated using algorithm 2 in the embedding space created by the Laplacian Eigenvectors method using the Jaccard Coefficient metric. Nodes that show a a preset threshold, S. Once the initial community has been built and the status of its members has been changed to "Visited," we proceed to the next node that has not been visited yet and has the highest level of centrality. To summarize, the algorithm 2 consists of a series of well-structured steps:

1. Select the node with the maximum centrality and a state of Not-visited.
2. Calculate the Jaccard Coefficient of Similarity between the most influential node and the remaining nodes that have the status of not-visited.
3. Combine the most important node in the community with additional nodes that are comparable to it, and then designate them as "Visited."
4. Continue doing the identical procedures for the subsequent significant vertex that has not been visited until all the vertices in the graph have been marked as visited.

Algorithm 2: Identify initial community structure

Require: $PC_{(initial)} = C_1, C_2, \dots C_n$ initial community structure

Ensure: $G(V, E)$, influential nodes and embedding space, Threshold: S

```
P ⇐ {}
while i ≤ n do
   if Status (V_in[i] ==Not Visited) then
      Co_i ⇐ v_in
      Cu.append ⇐ Co
   end if
   while j = i + 1 ≤ n do
      if Status (Vin[j] ==Not Visited) then
         Similarity = JC (embed(co), embed (Vin[j]))
         if Similarity > S then
            Cu.append ⇐ v_in[j]
            Status V_in[j] ⇐ Visited
         end if
      end if
   end while
end while
P.append (Cu)
Co = {}
PC (initial) ⇐ Merge(P)
```

4.3. Selection of Strong and Weak Communities

Weak communities are smaller than those found outside of them, they may be singleton communities, for instance, or have fewer interactions among members. Consequently, some of the acquired communities need to be joined to get the optimal community structure for the graph, as provided by Algorithm 3.

Algorithm 3: Weak and Strong Communities

Require: Selection of Weak and Strong Communities

Ensure: $PC(Final) = C_1, C_2, \dots, C_n$ Select Strong and Weak communities among P

```
while Ci ϵ P do
   if length (Ci < Average (Vi) or Ci ∄ any 3-clique then
```

3. C_{Weak}.append(C_i)
4. **else**
5. C_{Strong}.append(C_i)
6. PC_{Final}.append(C_{Strong})
7. **end if**
8. **end while**

4.4. Final Merging of Communities

The most popular methods for community structure optimization have been proposed in the literature and are centered around maximizing or minimizing a certain objective function. The next stage, which is described in Algorithm 4, is to identify which communities were weak in the initial community structure and merge them with strong communities. To minimize the temporal complexity as much as feasible, we calculated the Jaccard Coefficient similarity [68] between the cores of strong communities and the members of weak communities. The initial stage in merging weak and strong communities is to find the Jaccard Coefficient similarity between each weak community's nodes and each strong community's core that can be calculated using Equation 10.

Algorithm 4: Finally Merging Strong and Weak Communities

Require: *PC(Merged)* = $C_1, C_2, ..., C_n$ Merged communities

Ensure: *PC(Final)*

1. **while** i ϵ C_{Strong} **do**
2. $\text{Maximum}_{\text{Similarity}} \Leftarrow -2$
3. $\text{Index}_{\text{Strong}} \Leftarrow 0$
4. $\text{Index}_{\text{Weak}} \Leftarrow 0$
5. **while** j ϵ C_{Weak} **do** Calculate the Jaccard Coefficient similarity of each Strong and Weak community using Equations 10, 11, and 12.
6. **if** Similarity(i,j) > $\text{Maximum}_{\text{Similarity}}$ **then**
7. $\text{Maximum}_{\text{Similarity}}$ = Similarity (i, j)
8. $\text{Index}_{\text{Strong}} \Leftarrow i$
9. $\text{Index}_{\text{Weak}} \Leftarrow j$
10. **end if**
11. **end while**
12. **end while**
13. $i \Leftarrow \text{Index}_{\text{Weak}} \cup \text{Index}_{\text{Strong}}$

$$sim\ (u, v)^{Jaccard} = \frac{|N_u \cap N_v|}{|N_u \cup N_v|} \tag{10}$$

N_u and N_v represent the collection of things that users u and v, respectively, have rated. The addition rule theorem is applied in this case to form $|N_u \cap N_v| = |N_u| + |N_v| - |N_u \cap N_v|$, since N_u and N_v are not mutually exclusive. On the other hand, according to Equation 11, $|Nu|$ and $|Nv|$ represent the cardinality of the sets N_u and N_v, respectively.

$$sim\ (u, v)^{Jaccard} = \frac{|N_u \cap N_v|}{|N_u| + |N_v| - |N_u \cap N_v|} \tag{11}$$

Suppose $|\overline{N_u}|$, $|\overline{N_v}|$ are the cardinality of the set of items un-co-rated by users u and v respectively. Hence, $|\overline{N_u}| = |N_u| - |N_u \cap N_v|$ and $|\overline{N_v}| = |N_v| - |N_u \cap N_v|$. As a result, the Jaccard similarity can be written as in Equation 12.

$$sim\ (u,v)^{Jaccard} = \frac{|N_u \cap N_v|}{(|\overline{N_u}|+|N_u \cap N_v|)+(|\overline{N_v}|+|N_u \cap N_v|)-|N_u \cap N_v|} = \frac{|N_u \cap N_v|}{|\overline{N_u}|+|\overline{N_v}|+|N_u \cap N_v|} \tag{12}$$

Modularity is the main parameter to take into account when talking about community detection. A network's ability to be separated into groups is measured by its modularity [69]. Optimization structures utilize modularity to detect community networks. This pertains to the disparity between the real and anticipated quantities of edges. The notation employed in Equation 13 denotes modularity Q:

$$Q = \sum_i (e_{ii} - a_i^2) \tag{13}$$

Two communities should be merged if there are a greater number of connections between them compared to other groupings as can be extracted using algorithm 5. The variable l_{ij} is defined as the count of inter-community linkages between C_i and C_j, as stated in Equation 14.

$$l_{ij} = \left|(v_i,\ v_j): v_i \in\ C_i\ and\ v_j \in\ C_j\right| \tag{14}$$

We use Equation 15 to determine if, among all the communities in the community setting, the community C_j and C_i should merge.

$$S_{ij} = \frac{l_{ij}}{dc_i dc_j} \tag{15}$$

Let Qm_j represent the community set's modularity before merging. If $Qm_j > Qm$, merge Com_i and Com_j into a single community to update the community structure. This process should be continued until there is no more room for improvement in modularity; at that time, the resulting community structure will have the highest feasible modularity. Inter- and intra-community edges are used to visually portray the identified communities to improve understanding of the relationships between nodes and communities.

Algorithm 5: Detecting Community

Require: Final set of communities

Ensure: List of C_{Weak}, C_{Strong}

1. Com ⇐ 0
2. **while** for every node u_i in V_j **do**
3. Retrieve node based on similarity of ui, as vj
4. **if** C_{ui},C_{vj} ∄ **then**
5. Create C_{ui} as $C_{ui} = \{u_i,v_j\}$
6. Com ⇐ Com ∪ {Cui}
7. **else if** C_{ui} ∃ and v_j ∄ in any Com_i **then**
8. $C_{ui} \Leftarrow C_{ui} \cup \{v_j\}$
9. **else if** C_{vj} ∃ and u_i ∄ in any Com_i **then**
10. $C_{vj} \Leftarrow C_{vj} \cup \{u_i\}$
11. **end if**
12. Repeat
13. **end while**
14. Compute modularity Q by using Equations 1, 14, and 15.
15. Select Com_i and Com_j such that $St_{ij} = \max[St_{mn} : Co_m, Co_n] \in$ Com using Equation 15.
16. Compute Modularity Q_{mj} for Com - Com_i, $Com_j \cup Com_i \cup Com_j$.
17. **if** $Q_{mj} > Q_m$ **then**
18. $Com_i \Leftarrow Com_i \cup Com_j$
19. Com = Com − {Comi,Comj}
20. Com_i = Com ∪ Com_i
21. **end if** till modularity does not show any improvement.

5. Evaluation

Modularity can be utilized to evaluate the results of multiple algorithms and identify the optimal method through CD.

5.1. Modularity

The initial measure is widely recognized in the literature. This method compares the actual connections within a community with the likelihood of finding those connections in a randomly generated network. The utility of a network is highest when there is a high density of links within communities and a low density of links between communities. The division that has the highest modularity score is regarded as the most optimal one in this scenario. Equation 16 provides the modularity of division D for a graph G in the following manner:

$$Q(D) = \sum_{i=1}^{|D|}(e_{ii} - a_i^2) \tag{16}$$

The likelihood of an intra-community link in the community C_i is denoted by e_{ij}, while the likelihood of a relationship with at least one extremity is indicated by a_i. The information that is normalized mutually (NMI) Normalized mutual information (NMI), normalized to a number between 0 and 1, is used to determine the amount of information about two variables. The NMI is calculated using Equation 17, which involves taking the logarithm of the ratio between the joint probability of communities U and V and the product of the probabilities of each community, denoted as $logPUV\ (i,j)\ PU(i)PV\ (j)$. Values approaching 1 imply a robust connection between two variables, while values approaching 0 signify a feeble one.

$$NMI(U,V) = \frac{2\sum_{i=1}^{R}\sum_{j=1}^{C} P_{UV}(i,j)\log\frac{P_{UV}(i,j)}{P_U(i)P_V(j)}}{-\sum_{i=1}^{R} P_U(i)\log P_U(i) - \sum_{i=1}^{R} P_V(i)\log P_V(i)} \tag{17}$$

6. Experiments and Results

This section presents the datasets and the algorithm's performance evaluation of the most sophisticated community detection methods. This axis's main goal is to carry out an experimental analysis to see whether our plan is feasible. We accomplish this by testing the model's performance on both simulated and real-world networks. As performance measures, we use industry-standard measurements like Modularity and NMI.

6.1. Experimental Setup

An Intel(R) Core (TM) i7 with 8 GB RAM and a 2.30 GHz processor was used to perform the suggested algorithm. While the code is written in Python, the remaining techniques were implemented using the Python igraph [70] package. To further visualize the identified communities, the network [71] module in Python is employed. Table 3 provides an overview of the six real-world datasets that we used using the recommended technique.

Table 3: Real world datasets

Approach	Network	m	n	C
[72]	Karate	78	34	2
[73]	Dolphins	259	62	2
[40]	Football	613	115	12
[74]	Amazon	925872	334863	75149
[74]	DBLP	1049866	317080	13477
[75]	Ego-Facebook	4039	388234	13

6.2. Real World Datasets

1. The network of Zachary's Karate Club is discussed in the paper by [72]. Zachary created a tangible network by utilizing the social connections among the 34 individuals in a karate group. Due to a political argument between the club's administration and instructor, the network has been separated into two sections. For this study, we utilize the most basic iteration of this network.
2. According to a social network called The Dolphin's Network [73], 62 bottle-nose dolphins that were sighted in New Zealand between 1994 and 2001 regularly formed associations with one another. Within the network, there are two groups.
3. According to [40], the College Football Network displays the 2000 college football schedule, with teams represented by vertices and games between two teams represented by edges. Of the 115 vertices in the network, twelve are coalitions.
4. The data for the Amazon product co-purchasing network was obtained by systematically browsing the Amazon website, where vertices stand in for the items [74]. If items I and J are routinely bought together, an undirected edge forms between them.
5. A co-authorship network called the Digital Bibliography and Library Project (DBLP) [74] connects authors who have collaborated on at least one paper.
6. An individual's social network is represented by the Ego-Facebook Network Data Set, which is made up of groups from Facebook networks (also known as friends lists). With Facebook users as vertices and various types of relationships between 10 ego-networks as edges, it contains 4039 nodes and 88234 edges, each of which has 193 ground-truth circles [75].

6.3. Evaluation and Discussion

A few particular adjustments are needed for the suggested algorithm to function better. Using the Eigenmaps approach, we first extract the most influential nodes. The obtained vectors in the embedding space are employed in the second and third rounds of the proposed method to construct and improve the initial community structure [30, 32]. Therefore, the value of d directly affects how communities are identified. As such, it has a major impact on the performance of the proposed model. The actual network structure will modify this value to determine the appropriate dimension d. The second step of the proposed method is to cluster the node-representation vectors based on their similarity. The goal is to establish an initial community structure by clustering nodes that are similar to each other, which can then be further improved in the third phase. A node is said to belong to the community of an influential node if there exists a substantial degree of similarity between them. During the trial phase, a node is deemed to be a member of the core community if the similarity between the two nodes surpasses 0.8. It is crucial to remember that this value was utilized by every network that was analyzed [34]. We have used Algorithm 5 on the six real-world networks with ground truth community structure. The found community structures were assessed and measured for both modularity and NMI using Algorithm 5 and state-of-the-art methods. Tables 4 and 5 present the findings of the assessment metrics, listing and contrasting them with the most sophisticated community detection methods.

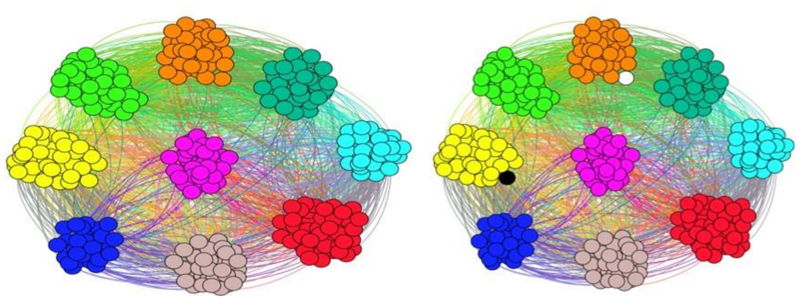

Figure 7: Karate Club Community structure (Left) Ground Truth (Right) Community Structure detected by Hybrid Model

The findings of Algorithm 5 for every network on the list were analyzed and presented individually. The ground truth and the Algorithm 5 discovered communities' visualization are displayed in the figures for each data collection. The identified communities are highlighted with different colors. Here, we illustrate the intermediate steps for applying algorithm 5 to derive the final community structure from the karate club network's preliminary community structure.

Table 4 Modularity values obtained in real-life datasets with ground truth

Networks	Louvain [37]	Spinglass [76]	Walktrap [77]	FLPA [22]	Girvan Newman [40]	Infomap [78]	Proposed
Karate	0.52	0.55	0.53	0.57	0.50	0.52	0.61
Dolphins	0.41	0.43	0.46	0.44	0.49	0.47	0.53
Football	0.71	0.53	0.62	0.51	0.54	0.63	0.55
Amazon	0.62	0.63	0.65	0.60	0.66	0.62	0.78
DBLP	0.50	0.62	0.72	0.59	0.61	0.76	0.86
Ego-Facebook	0.50	0.62	0.42	0.64	0.48	0.46	0.67

At each phase, two communities are selected, and those communities are then amalgamated, based on the number of edges both within and between communities. The communities will merge once again if the modularity keeps getting better. The karate club network in Figure 7 was processed using algorithm 5, producing different communities in the result, compared to two in the ground truth. However, the modularity and NMI are higher than with the other methods [24, 26]. Figure 8 shows the resulting dolphin social network, which has six communities instead of the ground truth's four communities. The NMI of the found communities is the greatest among the different algorithms. Furthermore, modularity is comparable to Louvain [37] and Infomap [78] algorithms and is higher. The communities with the highest modularity and NMI have been discovered, among other approaches. Our method using algorithms from 1 to 5 produces better NMI and modularity when compared to the ground truth, even while the number of recognized communities varies. We use the proposed algorithm along with related methods [20, 23]to extract communities from the six real-world networks presented in Table 3 once the experimental setup and data are collected. The suggested algorithm successfully identifies the Karate network's communities with unique membership features, displayed in Figure 7.

Table 5: NMI values obtained in real-life datasets with ground truth

Networks	Louvain [37]	Spinglass [76]	Walktrap [77]	FLPA [22]	Girvan Newman [40]	Infomap [78]	Proposed
Karate	0.62	0.65	0.68	0.60	0.65	0.69	0.74
Dolphins	0.42	0.59	0.59	0.42	0.56	0.59	0.65
Football	0.74	0.70	0.75	0.71	0.73	0.76	0.80
Amazon	0.60	0.64	0.65	0.50	0.70	0.57	0.82
DBLP	0.80	0.83	0.85	0.81	0.84	0.86	0.91
Ego-Facebook	0.60	0.52	0.58	0.54	0.68	0.66	0.72

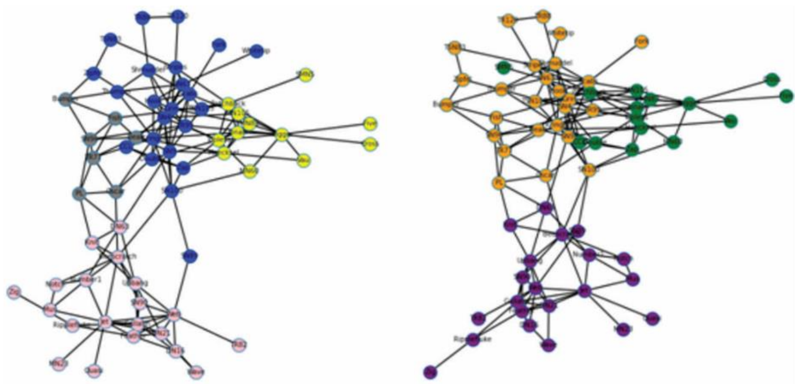

Figure 8: Dolphin Community structure (Left) Ground Truth (Right) Community Structure detected by Hybrid Model

Table 6: Final ranking for the compared algorithms in terms of NMI on six Datasets

Networks	Louvain [37]	Spinglass [76]	Walktrap [77]	FLPA [22]	Girvan Newman [40]	Infomap [78]	Proposed
Karate	6	8	5	11	10	8	1
Dolphins	3	12	8	4	7	5	1
Football	7	3	2	8	5	6	1
Amazon	3	5	7	4	6	4	2
DBLP	10	11	6	5	5	7	1
Ego-Facebook	11	13	8	4	7	9	5

This is further corroborated by the NMI metrics in Table 6, demonstrating how the suggested algorithm performs better than alternative methods in obtaining higher values. This figure 7 demonstrates that the suggested method works better for the Karate network, despite having a lower value than the Walktrap [77], FLPA [22], Louvain [37], and Spinglass [76] algorithms. Regardless of the size of the dataset, our method computes a community structure that is more congruent with the ground truth than opposing approaches. As Table 6 makes clear, our idea has further important advantages. The stability of our proposed hybrid model algorithm surpasses that of other algorithms, including Louvain [37], FLPA [22], and Spinglass [76]. This is true as our concept does not rely on a haphazard process.

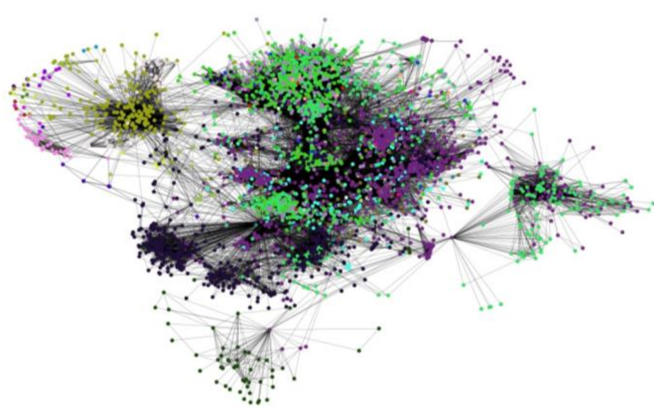

Figure 9: Clustering of different Communities

Additionally, our suggested hybrid model works well in dense graphs, providing significant NMI values, which is seen to be an essential benefit for social network community finding. Figure 9 displays the community structure that the hybrid model identified. To summarize, as discussed in [19] the quick identification of a community structure that closely mimics the actual structure. Therefore, our approach performs exceptionally well in locating significant communities inside social networks.

To demonstrate how much better the recommended method is, the experiment selects the baseline approaches, which include several traditional community detection techniques and well-known clustering techniques. To assess the performance of the Proposed technique, it will be compared to existing embedding-based baselines with existing graph embedding baseline approaches [1–3] for CD, we fit networks into them. After that, we extract community divisions from the lowdimensional vector space we have learned using the Algorithm 5. Using the data from Tables 4 and 5, we compare the results with our proposed technique and six existing community detection methods on the network of real-world ground truth datasets. We used the modularity value (Q) and the NMI value in the evaluation procedure following [6, 8, 10]. The strategy achieves all six of the highest NMI values and all five of the highest Q values in the real-world community datasets. Even if there is a tiny discrepancy between the Q values produced from the Ego-Facebook dataset and the Q values in the Dolphins dataset, the largest NMI value of 1 is attained, demonstrating that it is exactly the correct community for the actual categorization. In the DBLP dataset, the recommended approach performs better than the other methods [4, 5]and yields more accurate information, obtaining the greatest NMI value of 0.91. The Amazon dataset has higher Q and NMI values, and the NMI is closer to 1, which is more consistent with data from genuine communities. With the greatest outcomes and the greatest rationality and effectiveness compared to the other algorithms, which generated the least modularity, was DBLP.

The empirical results give compelling proof that the suggested hybrid algorithm performs well at finding meaningful communities across a wide range of real-world networks. The approach got the best modularity (Q = 0.86) and NMI (NMI = 0.91) on the DBLP dataset, beating the best methods like Louvain, Infomap, and Walktrap. On networks like Karate and Dolphins, the model also consistently came in first in modularity and NMI, which means it did a superior job of grouping nodes by structure. The results show that combining eigenvector and proximity centrality with embedding methods makes it easier to find important nodes and community boundaries. The higher modularity shows that the algorithm is good at building communities that are dense on the inside and sparse on the outside. The high NMI scores show that the results match the ground truth. So, the suggested model shows that it can be used in a wide range of situations, is accurate, and is useful in real life with complicated graph topologies.

6.4. Time Complexity

The greedy method requires $0(n+m)$ for each iteration. CN iterations were required for this method to function. Consequently, the overall time complexity of this algorithm was $0(m+n)(n)$. Since there are more edges than nodes in the graph, $0(mn)$ has the highest level of complexity. In the end, the temporal complexity of the method can be represented by Equation 18 as follows:

$$T(N) = (m.d_w^2) + O(n^2) + O(ns.nw) \tag{18}$$

which is almost equivalent to O ($m.d^2$).

7. Conclusion

Easy communication between individuals on a single platform has been made possible by the growth of the web and the development of SNS. A graph containing nodes and edges linking the nodes can be used to depict a social network. The edges show how these entities interact with one another, whereas the nodes represent the individuals or entities. Individuals with comparable choices, tastes, and preferences who frequently connect on social media platforms create virtual groups or communities. It entails recognizing cohesive groups with related entities and setting cohesive groups apart from other groupings. Numerous approaches have been put out for community detection, each taking a distinct angle on the issue. Large-scale graph handling community detection techniques, however, are now required since complex and huge networks are emerging across multiple sectors. This research suggests a new method to detect social networks of people depending on community knowledge and embedding spaces of similar nodes. To improve CD in social networks, we integrate eigenvector centrality and closeness measurements. Comprehensive tests on real-world networks show our proposal's effectiveness. The experimental findings demonstrate how robust and efficient the suggested method is, and how well it performs in large-scale graphs when compared to other well-known algorithms. The complexity of the algorithm stays quadratic about the number of vertices and linear about the number of iterations. This algorithm is still not very good at solving big data challenges. However, there is still research to see if changing the data format will bring the complexity down to almost a linear level. It is also possible to develop better disassembly methods for communities and nodes, which would require fewer iterations to achieve increased modularity. This method can be applied to real-world problems to detect communities in the bio-informatics industry, which entails assembling the major proteins involved in cancer and searching for functional connections within the resulting communities.

7.1. Limitations

The quality of the user interaction data may affect the computed centrality measures. Inaccurate or lacking data may lead to incorrect inferences. Online sites may restrict access to user interaction data due to privacy limitations. It is imperative to consider ethical considerations when collecting and utilizing this type of data. Computing proximity centrality and eigenvector centrality can be computationally costly for very large user interaction networks. This might limit how large the method can be scaled for real-world applications. When paired with other user data, careful feature engineering can be required to guarantee that centrality ratings make a substantial contribution to the clustering process. User interaction networks are dynamic, and communities are subject to change over time. It's possible that the communities that have been identified don't precisely match the most recent configuration of the network.

Funding Statement: This research did not receive any specific grant from any funding agency.

Conflicts of Interest: There are no financial, personal, or professional conflicts of interest.

Data Availability: This study utilizes six real-world datasets, which were obtained from publicly accessible repositories.

References

[1] Bolorunduro, Janet Oluwasola, and Zhaonian Zou. "Community detection on multi-layer graph using intra-layer and inter-layer linkage graphs (cdmiilg)." *Expert Systems with Applications* 238 (2024): 121713.

[2] Yang, Cheng, Jixi Liu, Yunhe Yan, and Chuan Shi. "Fairsin: Achieving fairness in graph neural networks through sensitive information neutralization." In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 8, pp. 9241-9249. 2024.

[3] Gadár, László, and János Abonyi. "Finding multifaceted communities in multiplex networks." *Scientific Reports* 14, no. 1 (2024): 14521.

[4] Cavoretto, Roberto, Alessandra De Rossi, Sandro Lancellotti, and Federico Romaniello. "Node-bound communities for partition of unity interpolation on graphs." *Applied Mathematics and Computation* 467 (2024): 128502.

[5] Christensen, Alexander P., Luis Eduardo Garrido, Kiero Guerra-Peña, and Hudson Golino. "Comparing community detection algorithms in psychometric networks: A Monte Carlo simulation." *Behavior Research Methods* 56, no. 3 (2024): 1485-1505.

[6] Xia, Yuanxing, Qingshan Xu, Jicheng Fang, Rongchuan Tang, and Pengwei Du. "Bipartite graph-based community-to-community matching in local energy market considering socially networked prosumers." *Applied energy* 353 (2024): 122245.

[7] Baruah, Bikash, Manash P. Dutta, Subhasish Banerjee, and Dhruba K. Bhattacharyya. "A novel density based community detection algorithm and its application in detecting potential biomarkers of ESCC." *Journal of Computational Science* 81 (2024): 102344.

[8] Zheng, Jianxing, Suge Wang, Deyu Li, and Bofeng Zhang. "Personalized recommendation based on hierarchical interest overlapping community." *Information Sciences* 479 (2019): 55-75.

[9] Adraoui, Meriem, Asmaâ Retbi, Mohammed Khalidi Idrissi, and Samir Bennani. "Maximal cliques based method for detecting and evaluating learning communities in social networks." *Future Generation Computer Systems* 126 (2022): 1-14.

[10] Fortunato, Santo. "Community detection in graphs." *Physics reports* 486, no. 3-5 (2010): 75-174.

[11] Yilmaz, L. Safak, and Albertha JM Walhout. "Metabolic network modeling with model organisms." *Current opinion in chemical biology* 36 (2017): 32-39.

[12] Newman, Mark EJ, and Michelle Girvan. "Finding and evaluating community structure in networks." *Physical review E* 69, no. 2 (2004): 026113.

[13] Ahajjam, Sara, Mohamed El Haddad, and Hassan Badir. "A new scalable leader-community detection approach for community detection in social networks." *Social Networks* 54 (2018): 41-49.

[14] Azaouzi, Mehdi, Delel Rhouma, and Lotfi Ben Romdhane. "Community detection in large-scale social networks: state-of-the-art and future directions." *Social Network Analysis and Mining* 9, no. 1 (2019): 23.

[15] Rostami, M., Farrahi, V., Ahmadian, S., Jalali, S.M.J. and Oussalah, M., 2023. A novel healthy and time-aware food recommender system using attributed community detection. *Expert Systems with Applications*, *221*, p.119719.

[16] Rostami, Mehrdad, Mourad Oussalah, Kamal Berahmand, and Vahid Farrahi. "Community detection algorithms in healthcare applications: a systematic review." *IEEE Access* 11 (2023): 30247-30272.

[17] Rostami, Mehrdad, Usman Muhammad, Saman Forouzandeh, Kamal Berahmand, Vahid Farrahi, and Mourad Oussalah. "An effective explainable food recommendation using deep image clustering and community detection." *Intelligent Systems with Applications* 16 (2022): 200157.

[18] Alotaibi, Norah, and Delel Rhouma. "A review on community structures detection in time evolving social networks." *Journal of King Saud University-Computer and Information Sciences* 34, no. 8 (2022): 5646-5662.

[19] Xu, Mengjia. "Understanding graph embedding methods and their applications." *SIAM Review* 63, no. 4 (2021): 825-853.

[20] Kumar, Prashant, Raghav Jain, Shivam Chaudhary, and Sanjay Kumar. "Solving community detection in social networks: A comprehensive study." In *2021 5th International Conference on Computing Methodologies and Communication (ICCMC)*, pp. 239-345. IEEE, 2021.

[21] Bhattacharya, Riju, Naresh Kumar Nagwani, and Sarsij Tripathi. "A community detection model using node embedding approach and graph convolutional network with clustering technique." *Decision Analytics Journal* 9 (2023): 100362.

[22] Traag, Vincent A., and Lovro Šubelj. "Large network community detection by fast label propagation." *Scientific Reports* 13, no. 1 (2023): 2701.

[23] Chen, Gaolin, and Shuming Zhou. "A novel overlapping community detection strategy based on Core-Bridge seeds." *International journal of machine learning and cybernetics* 15, no. 6 (2024): 2131-2147.

[24] Mohamed, El-Moussaoui, Tarik Agouti, Abdessadek Tikniouine, and Mohamed El Adnani. "A comprehensive literature review on community detection: Approaches and applications." *Procedia Computer Science* 151 (2019): 295-302.

[25] Mothe, Josiane, Karen Mkhitaryan, and Mariam Haroutunian. "Community detection: Comparison of state of the art algorithms." In *2017 Computer Science and Information Technologies (CSIT)*, pp. 125-129. IEEE, 2017.

[26] Newman, Mark EJ. "Fast algorithm for detecting community structure in networks." *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics* 69, no. 6 (2004): 066133.

[27] Chintalapudi, S. Rao, and MHM Krishna Prasad. "Finding research groups using modularity based community detection algorithm." In *2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)*, pp. 65-69. IEEE, 2016.

[28] Behera, Ranjan Kumar, S. K. Rath, and Monalisa Jena. "Spanning tree based community detection using min-max modularity." *Procedia Computer Science* 93 (2016): 1070-1076.

[29] Krishnan, S. Gokula, S. Karthika, and S. Bose. "Detection of communities in dynamic social networks." In *2016 International Conference on Recent Trends in Information Technology (ICRTIT)*, pp. 1-6. IEEE, 2016.

[30] Jiang, Tianhua, Chao Zhang, Huiqi Zhu, and Guanlong Deng. "Energy-efficient scheduling for a job shop using grey wolf optimization algorithm with double-searching mode." *Mathematical Problems in Engineering* 2018, no. 1 (2018): 8574892.

[31] Gutiérrez, Inmaculada, Daniel Gómez, Javier Castro, and Rosa Espínola. "A new community detection problem based on bipolar fuzzy measures." In *Computational Intelligence and Mathematics for Tackling Complex Problems 2*, pp. 91-99. Cham: Springer International Publishing, 2022.

[32] Huang, Mingqing, Qingshan Jiang, Qiang Qu, and Abdur Rasool. "An overlapping community detection approach in ego-splitting networks using symmetric nonnegative matrix factorization." *Symmetry* 13, no. 5 (2021): 869.

[33] Zhu, Jiajing, Yongguo Liu, Hao Wu, Zhi Chen, Yun Zhang, Shangming Yang, Changhong Yang, Wen Yang, and Xindong Wu. "A no self-edge stochastic block model and a heuristic algorithm for balanced anti-community detection in networks." *Information Sciences* 518 (2020): 95-112.

[34] Yin, Ying, Yuhai Zhao, He Li, and Xiangjun Dong. "Multi-objective evolutionary clustering for large-scale dynamic community detection." *Information Sciences* 549 (2021): 269-287.

[35] Tabarzad, Mohammad Ali, and Ali Hamzeh. "A heuristic local community detection method (HLCD)." *Applied Intelligence* 46, no. 1 (2017): 62-78.

[36] Zhang, Xingyi, Congtao Wang, Yansen Su, Linqiang Pan, and Hai-Feng Zhang. "A fast overlapping community detection algorithm based on weak cliques for large-scale networks." *IEEE Transactions on Computational Social Systems* 4, no. 4 (2017): 218-230.

[37] Zhou, Xiaojun, Ke Yang, Yongfang Xie, Chunhua Yang, and Tingwen Huang. "A novel modularity-based discrete state transition algorithm for community detection in networks." *Neurocomputing* 334 (2019): 89-99.

[38] Blondel, Vincent D., Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. "Fast unfolding of communities in large networks." *Journal of statistical mechanics: theory and experiment* 2008, no. 10 (2008): P10008.

[39] Traag, Vincent A., Ludo Waltman, and Nees Jan Van Eck. "From Louvain to Leiden: guaranteeing well-connected communities." *Scientific reports* 9, no. 1 (2019): 1-12.

[40] Girvan, Michelle, and Mark EJ Newman. "Community structure in social and biological networks." *Proceedings of the national academy of sciences* 99, no. 12 (2002): 7821-7826.

[41] Ghoshal, Arnab Kumar, Nabanita Das, Subhasis Bhattacharjee, and Goutam Chakraborty. "A Fast Parallel Genetic Algorithm Based Approach for Community Detection in Large Networks." In *COMSNETS*, pp. 95-101. 2019.

[42] Alhijawi, Bushra, and Arafat Awajan. "Genetic algorithms: Theory, genetic operators, solutions, and applications." *Evolutionary Intelligence* 17, no. 3 (2024): 1245-1256.

[43] Zeng, Xiangxiang, Wen Wang, Cong Chen, and Gary G. Yen. "A consensus community-based particle swarm optimization for dynamic community detection." *IEEE transactions on cybernetics* 50, no. 6 (2019): 2502-2513.

[44] Hafez, Ahmed Ibrahem, Aboul ella Hassanien, Aly A. Fahmy, and Mohamed Fahmy Tolba. "Community detection in social networks by using Bayesian network and Expectation Maximization technique." In *13th International Conference on Hybrid Intelligent Systems (HIS 2013)*, pp. 209-214. IEEE, 2013.

[45] Srinivas, Sharan, and Chandrasekharan Rajendran. "Community detection and influential node identification in complex networks using mathematical programming." *Expert Systems with Applications* 135 (2019): 296-312.

[46] Cheng, Jianjun, Xing Su, Haijuan Yang, Longjie Li, Jingming Zhang, Shiyan Zhao, and Xiaoyun Chen. "Neighbor similarity based agglomerative method for community detection in networks." *Complexity* 2019, no. 1 (2019): 8292485.

[47] You, Xuemei, Yinghong Ma, and Zhiyuan Liu. "A three-stage algorithm on community detection in social networks." *Knowledge-Based Systems* 187 (2020): 104822.

[48] Kasoro, Nathanaël, Selain Kasereka, Elie Mayogha, Ho Tuong Vinh, and Joël Kinganga. "PercoMCV: A hybrid approach of community detection in social networks." *Procedia Computer Science* 151 (2019): 45-52.

[49] Tahir, Noman, Ali Hassan, Muhammad Asif, and Shahbaz Ahmad. "MCD: mutually connected community detection using clustering coefficient approach in social networks." In *2019 2nd International Conference on Communication, Computing and Digital systems (C-CODE)*, pp. 160-165. IEEE, 2019.

[50] Bai, Liang, Jiye Liang, Hangyuan Du, and Yike Guo. "A novel community detection algorithm based on simplification of complex networks." *Knowledge-Based Systems* 143 (2018): 58-64.

[51] Rustamaji, Heru C., Yustina S. Suharini, Angga A. Permana, Wisnu A. Kusuma, Sri Nurdiati, Irmanida Batubara, and Taufik Djatna. "A network analysis to identify lung cancer comorbid diseases." *Applied Network Science* 7, no. 1 (2022): 30.

[52] Fortunato, Santo, and Marc Barthelemy. "Resolution limit in community detection." *Proceedings of the national academy of sciences* 104, no. 1 (2007): 36-41.

[53] Guo, Jiahao, Pramesh Singh, and Kevin E. Bassler. "Resolution limit revisited: community detection using generalized modularity density." *Journal of Physics: Complexity* 4, no. 2 (2023): 025001.

[54] Miyauchi, Atsushi, and Yasushi Kawase. "Z-score-based modularity for community detection in networks." *PloS one* 11, no. 1 (2016): e0147805.

[55] Fortunato, Santo, and Darko Hric. "Community detection in networks: A user guide." *Physics reports* 659 (2016): 1-44.

[56] He, Kun, Yingru Li, Sucheta Soundarajan, and John E. Hopcroft. "Hidden community detection in social networks." *Information Sciences* 425 (2018): 92-106.

[57] Chattopadhyay, Swarup, and Debasis Ganguly. "Node2vec with weak supervision on community structures." *Pattern Recognition Letters* 150 (2021): 147-154.

[58] Orman, Günce Keziban, Vincent Labatut, and Hocine Cherifi. "Qualitative comparison of community detection algorithms." In *International conference on digital information and communication technology and its applications*, pp. 265-279. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011.

[59] Freeman, Linton C. "A set of measures of centrality based on betweenness." *Sociometry* (1977): 35-41.

[60] Freeman, Linton C. "Centrality in social networks conceptual clarification." *Social networks* 1, no. 3 (1978): 215-239.

[61] Borgatti, Stephen P., and Martin G. Everett. "A graph-theoretic perspective on centrality." *Social networks* 28, no. 4 (2006): 466-484.

[62] Bavelas, Alex. "Communication patterns in task-oriented groups." *Journal of the acoustical society of America* (1950).

[63] Spizzirri, Leo. "Justification and application of eigenvector centrality." *Algebra in Geography: Eigenvectors of Network* (2011).

[64] Ding, De-wu, and Xiao-qing He. "Notice of Retraction: Application of eigenvector centrality in metabolic networks." In *2010 2nd International Conference on Computer Engineering and Technology*, vol. 1, pp. V1-89. IEEE, 2010.

[65] Shiokawa, Hiroaki, Yasuhiro Fujiwara, and Makoto Onizuka. "Scan++ efficient algorithm for finding clusters, hubs and outliers on large-scale graphs." *Proceedings of the VLDB Endowment* 8, no. 11 (2015): 1178-1189.

[66] Gmati, Haifa, Amira Mouakher, Antonio Gonzalez-Pardo, and David Camacho. "A new algorithm for communities detection in social networks with node attributes." *Journal of Ambient Intelligence and Humanized Computing* 15, no. 2 (2024): 1779-1791.

[67] Guerreiro, Lucas, Filipi Nascimento Silva, and Diego Raphael Amancio. "Identifying the perceived local properties of networks reconstructed from biased random walks." *Plos one* 19, no. 1 (2024): e0296088.

[68] Hamedani, Masoud Reyhani, and Sang-Wook Kim. "JacSim: An accurate and efficient link-based similarity measure in graphs." *Information Sciences* 414 (2017): 203-224.

[69] Newman, Mark EJ. "Modularity and community structure in networks." *Proceedings of the national academy of sciences* 103, no. 23 (2006): 8577-8582.

[70] Csardi, Gabor, and Tamas Nepusz. "The igraph software." *Complex syst* 1695 (2006): 1-9.

[71] Hagberg, Aric, Pieter J. Swart, and Daniel A. Schult. *Exploring network structure, dynamics, and function using NetworkX*. No. LA-UR-08-05495; LA-UR-08-5495. Los Alamos National Laboratory (LANL), Los Alamos, NM (United States), 2008.

[72] Avrachenkov, Konstantin E., Aleksei Yu Kondratev, and Vladimir V. Mazalov. "Cooperative game theory approaches for network partitioning." In *International Computing and Combinatorics Conference*, pp. 591-602. Cham: Springer International Publishing, 2017.

[73] Lusseau, David, Karsten Schneider, Oliver J. Boisseau, Patti Haase, Elisabeth Slooten, and Steve M. Dawson. "The bottlenose dolphin community of doubtful sound features a large proportion of long-lasting associations: can geographic isolation explain this unique trait?." *Behavioral ecology and sociobiology* 54, no. 4 (2003): 396-405.

[74] Yang, Jaewon, and Jure Leskovec. "Defining and evaluating network communities based on ground-truth." In *Proceedings of the ACM SIGKDD workshop on mining data semantics*, pp. 1-8. 2012.

[75] Jure, Leskovec. "Snap datasets: Stanford large network dataset collection." *Retrieved December 2021 from http://snap. stanford. edu/data* (2014).

[76] Eaton, Eric, and Rachael Mansbach. "A spin-glass model for semi-supervised community detection." In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 26, no. 1, pp. 900-906. 2012.

[77] Smith, Natalie R., Paul N. Zivich, Leah M. Frerichs, James Moody, and Allison E. Aiello. "A guide for choosing community detection algorithms in social network studies: The question alignment approach." *American journal of preventive medicine* 59, no. 4 (2020): 597-605.

[78] Zeng, Jianping, and Hongfeng Yu. "A distributed infomap algorithm for scalable and high-quality community detection." In *Proceedings of the 47th International Conference on Parallel Processing*, pp. 1-11. 2018.