Journal of Future Artificial Intelligence and Technologies

E-ISSN: 3048-3719

Research Article

Mapping Unseen Connections: Graph Clustering to Expose User Interaction Patterns

Haroon Ahmad [1], Muhammad Sajid [1], Faheem Mazhar [2,*], and Muhammad Fuzail [2]

[1] Department of Computer Science, Air University, Islamabad, 44230, Capital, Pakistan; e-mail : haroon@aumc.edu.pk; msajid@aumc.edu.pk

[2] Computer Science Department, NFC Institute of Engineering and Technology, 59030, Multan, Punjab, Pakistan; e-mail : faheemmazharbalouch@gmail.com; fuzail@nfciet.edu.pk

* Corresponding Author : Faheem Mazhar

Abstract: Expanding extensive and intricate social networks has required sophisticated community detection techniques. This study presents an innovative hybrid methodology that utilizes node space similarity and local knowledge to enhance community identification. Node space similarity is defined by integrating eigenvector centrality (EC), which quantifies a node's influence inside the network, with proximity metrics, such as closeness, to evaluate the connectivity between nodes. This enables us to identify cohorts of individuals with analogous influence and connectivity. We use local knowledge by concentrating on these pivotal nodes' direct connections and attributes, allowing the technique to broaden community discovery (CD) outward effectively. Our five-phase methodology, grounded in an iterative seed expansion algorithm, commences with identifying highly central nodes and progressively develops communities by integrating nodes exhibiting high similarity and local connectivity. The method incorporates graph statistical inference and embedding features to improve accuracy and capture extensive network patterns. This integrated approach facilitates the precise and effective identification of communities within extensive social networks, exceeding the constraints of conventional techniques. This research attained a modularity of 95.05% on the DBLP dataset and 94.50% on the Amazon dataset. This study achieved a Normalized Mutual Information (NMI) of 91.80% on the DBLP dataset, 92.50% on the Amazon dataset, and 90.43% on the football dataset, demonstrating superior performance relative to previous methodologies. The findings indicate that the hybrid method outperforms other recognized methods in large-scale graphs, showcasing notable robustness and efficiency.

Keywords: Closeness; Clustering; Communities; Eigenvector Centrality; Social Network; Strong and weak entities.

Received: February, 26th 2025
Revised: March, 18th 2025
Accepted: March, 22nd 2025
Published: March, 24th 2025
Curr. Ver.: March, 24th 2025

Copyright: © 2025 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY SA) license (https://creativecommons.org/licenses/by-sa/4.0/)

1. Introduction

The demand for social data is escalating with the proliferation of internet activities. Individuals increasingly use social media platforms to share opinions on products, policies, and social issues[1]. These platforms empower a wide range of users—from heads of state to small business owners—to disseminate information and access collective knowledge unrestricted by geographic or demographic constraints[2]. Social media facilitates the formation of virtual communities, transforming how people interact and share information. Prominent platforms such as Instagram, LinkedIn, Facebook, and Twitter now serve as significant arenas for engagement and analysis. Consequently, researchers are challenged with the extensive volume of data produced by user interactions, necessitating scalable and intelligent analytical methods[3].

Social Network Analysis (SNA) enables the exploration of social phenomena by analyzing the structure of user interactions within these networks[4], [5]. In such contexts, a social network (SN) is often modeled using graph theory, where nodes represent users and edges

represent relationships or interactions [6], [7]. Graph-based modeling captures the complexity of user behavior, preferences, and community dynamics[8], and is essential for analyzing phenomena in e-learning, healthcare, and recommender systems[9]–[15].

Community detection (CD) plays a pivotal role in understanding social networks' underlying structure and functionality. It identifies groups of users with similar interaction patterns, thereby revealing how information flows and how influence is distributed across the network[16]. Figure 1 illustrates a typical graph structure showing interconnections between multiple communities. CD approaches are typically based on topological features, enriched node, and edge information, or a combination [17]. However, selecting an appropriate community structure remains complex due to network size and temporal dynamics. Some methods can process large graphs quickly but may sacrifice the quality of detected communities[18].

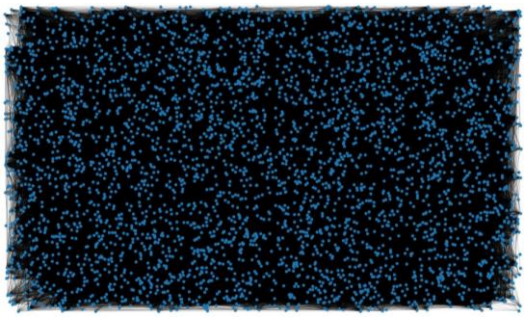

Figure 1. The graph illustrated with Communities.

High modularity indicates a strong community structure where nodes are densely connected and functionally distinct[19]. Figure 2 visualizes the outcome of applying our proposed methodology. It begins with representing the input network via an adjacency matrix and identifying prominent nodes based on their interaction density and connection frequency[20]. This approach utilizes graph embeddings to map high-dimensional graph structures into low-dimensional vector spaces for more efficient community detection[13].

Centrality metrics, including eigenvector centrality (EC) and proximity (closeness) centrality, quantify node importance and facilitate the identification of core members within communities[21]. Graph embedding techniques such as DeepWalk and node2vec preserve structural and proximity information, allowing nodes with similar vector representations to be clustered together [22], [23]. However, current modularity-based methods face challenges such as resolution limits, which prevent detecting small but meaningful communities or forming overly broad and indistinct clusters. These limitations reduce their ability to capture nuanced structures in large networks[22].

We propose a hybrid model that integrates modularity-based global network structure, centrality metrics, and graph embeddings to address these shortcomings for a more comprehensive community detection framework[4]. By leveraging EC, we identify influential nodes often residing at the center of otherwise undetected small communities[24]. Proximity centrality further improves precision by locating nodes strategically positioned within clusters, thereby helping establish more accurate community boundaries and avoiding excessive generalization. Together, these techniques provide a multi-level understanding of user interactions and influence dynamics in social networks.

Graph embedding is especially effective in transforming network data into a lower-dimensional vector space while preserving structural integrity[25]. In this form, nodes are represented uniformly, facilitating clustering and classification. The proposed method comprises three key phases. First, we construct the embedding space and identify highly influential nodes based on centrality measures. Second, we apply the Jaccard coefficient (JC) to cluster nodes most similar to these influential nodes within the embedding space, forming the initial community structure. Finally, weak communities excluded in the earlier phases are merged with stronger ones to form robust, well-defined clusters.

This study is motivated by identifying key individuals or influencers within social networks. Users with high EC may serve as brand ambassadors or knowledge disseminators, while those with high proximity centrality may act as bridges between distinct groups, facilitating cross-community communication. Recognizing such roles supports strategic objectives

like promoting collaboration or improving information dissemination across networks. By combining clustering with EC and proximity centrality, we aim to better understand how influence and interaction patterns shape social structures.

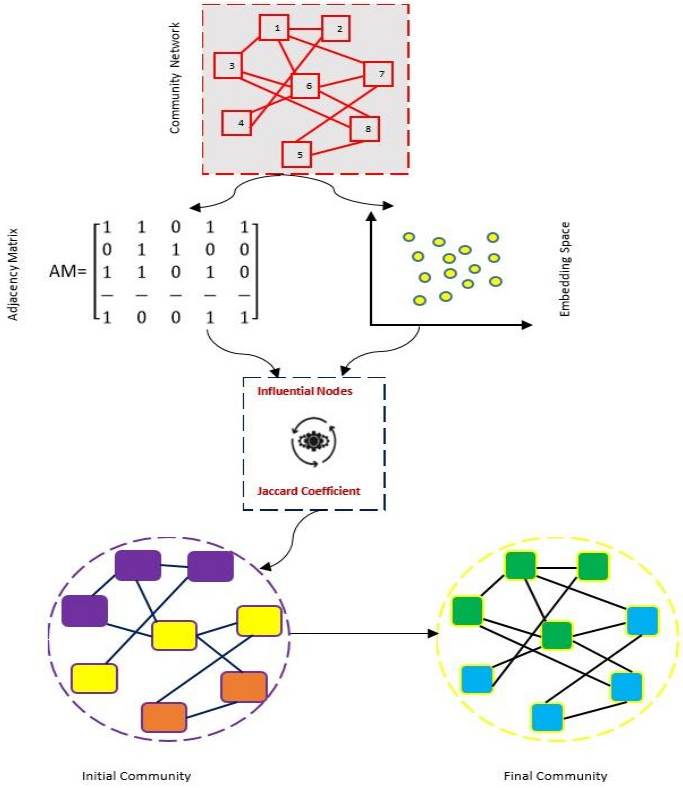

Figure 2. Proposed hybrid model ideology

Despite abundant user interaction data on online platforms, accurately identifying communities remains challenging due to the complex nature of user behavior and information diffusion. Our research seeks to answer the following questions:

- What is the level of connectivity among users within designated user societies?
- Who are the principal contributors or influencers in each community?
- Can a hybrid methodology integrating EC and proximity centrality improve the precision and comprehension of SN recognition?
- What insights can graph segmentation offer regarding user behavior and interaction routines?

The main contributions of this research are as follows:

- We assess community cohesiveness using edge weight, adaptability, and the mean grouping ratio.
- We analyze how central individuals influence user experiences, knowledge dissemination, and inter-community connections.
- We evaluate the hybrid method against traditional clustering algorithms (e.g., Louvain, Girvan-Newman), demonstrating improvements in modularity and clarity.
- We uncover hidden behavioral patterns using graph grouping techniques, highlighting the effectiveness of combining centrality and embedding strategies.

The structure of this paper is as follows. Section 2 presents related work. Section 3 explains the proposed hybrid model. Section 4 reports experimental results using real-world datasets. Finally, Section 5 concludes with insights and directions for future research.

2. Literature Review

2.1. Theoretical Foundations

This section provides a concise overview of the proposed hybrid model and illustrates an example of the utilized graph measurements. The architectural design of the proposed

solution is illustrated in Figure 2. The proposed paradigm has five stages. Initially, we establish an embedding space in which vector representations of the nodes are located. Through degree centrality assessments, we uncover nodes with a notable ability to establish communities and exert considerable influence over others.

Furthermore, we employ the JC similarity within the embedding space to cluster nodes that closely resemble prominent nodes within the same community, establishing an initial community structure. Moreover, the communities are categorized into either weak or strong classifications. Moreover, the less resilient groups initially excluded from the community structure generated during the s phase are integrated with the more robust communities. Finally, the ultimate communities are identified and prioritized based on modularity and NMI criteria.

2.2 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 SN 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 $C_i \epsilon D$ while having weak connections with users in different communities $C_j \neq i \epsilon D$. Furthermore, our objective is to identify significant actors or leaders inside each community $O_i \epsilon D$ to improve our comprehension of the internal organization of each community.

2.3. Significance of Nodes

2.3.1. Adjacency Matrix

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

$$a_i \begin{cases} 1, & \text{if } e_{u_i,u_j} \in E \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

2.3.2. Degree of Node

The degree of a graph $G = (N, E)$ is the number of edges that connect a node. Equation (2) provides the algorithm for computing the degree of a collection of neighbors of a node, represented by (u):

$$\text{Deg}_G(u_i) = |\rho_G(u_i)| = \left|\{u_j \in N | a_{i,j} = 1\}\right| \tag{2}$$

where $a_{i,j} = 1$ denotes the presence of an edge between u_i and u_j, and $|\rho_G(u_i)|$ represents the cardinality of the node's neighborhood. Formally, the degree of node u_i in G with an adjacency matrix $A = [a_i, j]$ is given in Equation (3).

$$\text{Deg}_G(u_i) = \sum_{j=1}^{n} a_{i,j} \tag{3}$$

Users engaging in a higher volume of interactions than their counterparts may wield more significant influence and enjoy enhanced access to information. Individuals possessing the highest educational attainment within the network are seen as active nodes, or hubs, proficient in disseminating knowledge within a certain region of the graph. In community detection, it is crucial to concentrate on these nodes, as they are generally the most significant and possess a high likelihood of creating communities.

2.3.3. Node Degree Centrality

The relative significance of a vertex inside the network is quantified by a metric referred to as degree centrality. To enable comparison, normalizing the degree value specified in the

Equation (4) is often beneficial. The degree centrality of node $u_i \in N$ is represented as $DC_G(\text{ui})$ whenever there is an adjacency matrix $AG = [a_i, x_n$.

$$\mathrm{DC}_G(u_i) = \frac{\mathrm{Deg}_G(u_i)}{n-1} = \frac{1}{n-1}\sum_{j=1}^{n} a_{i,j} \tag{4}$$

EC, betweenness centrality, and proximity centrality are among them. Each statistic signifies a unique focal point of attention. The centrality measures are thoroughly elucidated in Table 1.

Table 1. The time complexity of different centrality measures.

Approach	Centrality Measure	Time Complexity
Freeman et al. [26]	Degree Neighbors based Centrality	$0(n)$
Freeman et al. [27]	Closeness diameter-based Centrality	$0(n.log(n) + n.m$
Borgatti et al.[28]	Eigenvector values-based Centrality	$0(n^3)$
Borgatti et al.[28]	Betweenness flow-based Centrality	$0(n^3)$

2.3.4. Closeness Centrality

According to [29], 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.

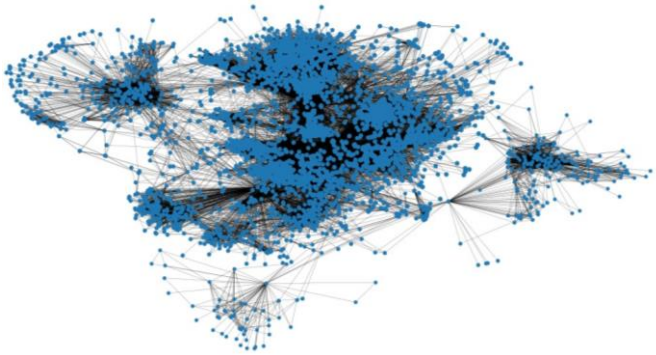

Figure 3. Community with Closeness Centrality

Equation (5) provides the estimated normalized CC_e of node j.

$$\mathrm{CC}_e[j] = \frac{N_o - 1}{\sum_{k=1}^{N_o} d_G(j,k)} \tag{5}$$

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 $\mathrm{CC}_e[j]$ 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 (6), from each given node to every other node

$$dl[k] = \sum_{j=1}^{N_o} dl_G(k,j) \tag{6}$$

2.3.5. Eigenvector Centrality (EC)

EC [30], as shown in Figure 4, evaluates a node's importance based not only on the number of its direct connections but also on the influence of its neighbors. Unlike degree centrality, which simply counts adjacent nodes (Equation (7)), EC assigns greater weight to connections with influential nodes. It reflects how well a node is connected to other highly

connected components in the network. Nodes with high EC scores typically maintain broad and deep connections across the network. EC corresponds to the dominant eigenvector of the network's adjacency matrix. SCAN++ leverages the property that real-world graphs, like web networks, often exhibit high clustering coefficients [31]. Node density is measured using the clustering coefficient[32], which increases when a node and its neighbors form a near-complete graph or clique, as illustrated in Figure 5. In real-world networks, nodes and their two-hop neighbors often share a significant portion of their connections [33], a characteristic utilized by SCAN++ to simplify density calculations by pruning shared nodes.

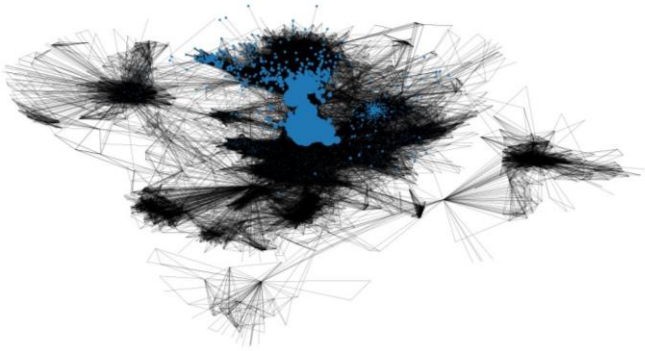

Figure 4. Community with Eigenvector Centrality

$$Av = \lambda v \tag{7}$$

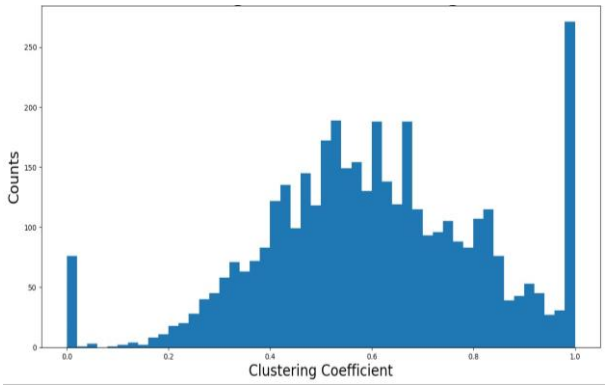

Figure 5. Clustering coefficient histogram

An eigenvector of a square matrix A is a non-zero vector v such that the product of A and v yields a scalar multiple of v. The symbol generally denotes the constant multiple λ. The eigenvalue λ is associated with the vector v in matrix A. Individuals with elevated EC occupy leadership roles within the network. They are frequently prominent individuals with an extensive association network with other distinguished figures. Consequently, they often function as prominent thinking leaders. Conversely, individuals with high EC may not consistently fulfill responsibilities characterized by high betweenness and closeness. The time complexity of integrating proximity centrality and EC for community detection can be reduced by mitigating the computational bottlenecks associated with each metric. We diminished the temporal complexity of proximity centrality by applying the random sampling technique. Consider using a random sampling strategy rather than calculating closeness centrality for each user. This necessitates fewer computations and yields a satisfactory estimate of average proximity centrality inside the network. To diminish the temporal complexity of EC measurements, we utilized iterative approaches. It asserts that iterative techniques like the Power Method can compute EC. Although these methods may require additional iterations to achieve convergence, they frequently outperform the explicit computation of the eigenvectors of the adjacency matrix in terms of speed.

2.4. An Index of Community Detection Methodologies

Community detection has been the subject of extensive research [18], and various algorithms[34] exist for this purpose. The methods can be classified into the following categories: 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[35].

2.5. Modularity-Driven Group Recognition

According to Equation (8), the Girvan-Newman algorithm [36] employed modularity, a recognized standard metric, to detect communities within the network. Subsequently, modularity served as the foundation for the development of further algorithms. These algorithms produce robust comparative results and are widely utilized across various fields, including product recommendations and research group identification [37]. The modularity metric is modified and linked to the spanning tree for community detection [38].

$$Q_m = \frac{1}{2n} \sum_{in,jn} \left[A_{injn} - \frac{Km_{in} * km_{jn}}{2n} \right] \delta(C_{in}, C_{jn}) \tag{8}$$

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 (9), the Kronecker delta is (C_{in}, C_{jn}). It equals 1 in the case when $c1$ equals $c2$ and 0 otherwise.

$$\delta(C_{in}, C_{jn}) = \begin{cases} 1, \text{if } in \text{ and } jn \text{ in similar community} \\ 0, \quad \text{else} \end{cases} \tag{9}$$

A density-based technique is an alternative method for CD[39]; however, this method requires the algorithm to accept the resolution parameter as input. By recognizing and addressing its deficiencies, the society attains more cohesion[40]. Nonetheless, the application of this technique results in inferior modularity and NMI performance measures for certain networks compared to alternative algorithms.

2.6. Evaluating Contemporary Community Detection Methods

Various methodologies have been employed historically to tackle the community detection issue. Several methodologies can identify communities, including networks, modularity, mathematical models, and evolutionary computing. Instances of these models encompass fuzzy logic[41], matrix factorization [42], and statistical methods[43]. Clans [44], local communities[45], and network embedding [46] exemplify applications of the network approach in research. The modularity technique optimizes community quality, as indicated by Louvain [47], Leiden [48], [49], Girvan Newman [36], and Greedy modularity[50]. Evolutionary computing techniques employ abstract principles from biological evolutionary theory to create optimization algorithms or methodologies. This method combines the tenets of biological evolution with computational technologies, including particle swarm optimization[51] and genetic algorithms[52]. Nonetheless, several strategies utilized to achieve this maximum modularity yield suboptimal results. Moreover, several methods generate groups of either substantial or negligible size that may lack practical significance. Some algorithms demonstrate reduced adaptability to alterations in the network, especially those involving the addition or removal of edges or nodes. The results differ when diverse methods are utilized to examine a network for community identification. Each technique produces unique modularity and community results[53].

2.7. Challenges in Community Detection: Surpassing Local Optima

Multiple factors may contribute to the emergence of local optima in community detection. Due to a resolution restriction, modularity-based community recognition algorithms may overlook small communities [54]. The generalized modularity density technique may discern communities of diverse sizes and configurations by assessing node density within the network[55], [56]. Modularity utilizing Z-scores, which standardizes the modularity score, is a further method that may detect communities of differing sizes [57], [58]. A significant concern is the inadequate community infrastructure [59]. Researchers have proposed many

techniques, like concealed community identification and poor supervision, to address this issue [60], [61]. Hidden communities denote clandestine or obscure groupings that are challenging to locate through traditional community detection methods. An alternative method for recognizing community structure is weak supervision, employing the node2vec technique[62] to detect communities of diverse sizes and shapes. Communities with a low degree of embedding present challenges in identification, as noted by [63], [64] in their research on node2vec.

2.8 Hybrid Approach for Community Detection Utilizing Enhanced Modularity

Table 2. Review of community detection algorithms.

Approach	Main Highlights	Parameter	Algorithm	Time Complexity
Hafez et al. [65]	Expectation Maximization (EM) is a statistical model of the interactions among participants in a social network	Directed Acyclic Graphs, EM	Bayesian network statistical model	$O(m.k)$
Srinivas et al. [66]	Simultaneously, the community structure and the influential nodes linked to each community will be determined.	Intra-community distance, Inter cluster distance	Mathematical programming model	$O(d_m + k_m^o)$
Cheng et al. [67]	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 Similarity-Based Local Algorithm (NSA)	$O(nlog(n))$
You et al. [68]	Identification of central nodes, the spread of labels, and combining of communities	Local and global information	Optimization algorithms	$0(n^3 + (nlog(n) + 0(n^3)$
Kasoro et al. [69]	Identify the complete set of communities computed by the clique percolation algorithm.	EC method	Clique percolation algorithm (CPM)	NA
Tahir et al. [70]	MCD (Mutual Community Detection) refers to analyzing 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. [71]	Convert an intricate network into a streamlined network, specifically a weighted tree (or forest).	Leading and following degrees	Tree-based Community Detection algorithm	$O(2n.m + n)$

A summary is shown in Table 2, with an account of their principal contributions. Most techniques for community detection utilize modularity and similarity metrics independently[72], [73]. The proposed hybrid method utilizes the modularity of network measurements, such as proximity and EC, to improve the ultimate community structure. Moreover, our approach surpasses prior methods regarding collaborative results, NMI metrics, and node categorization. It also exhibits exceptional modularity.

3. Proposed Ideology

We delineate the essential terminology of our proposed hybrid model and, thereafter, provide an in-depth explanation of each step of the model. Figure 6 illustrates the flow diagram for the Proposed Hybrid paradigm. The proposed methodology consists of the following primary 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.

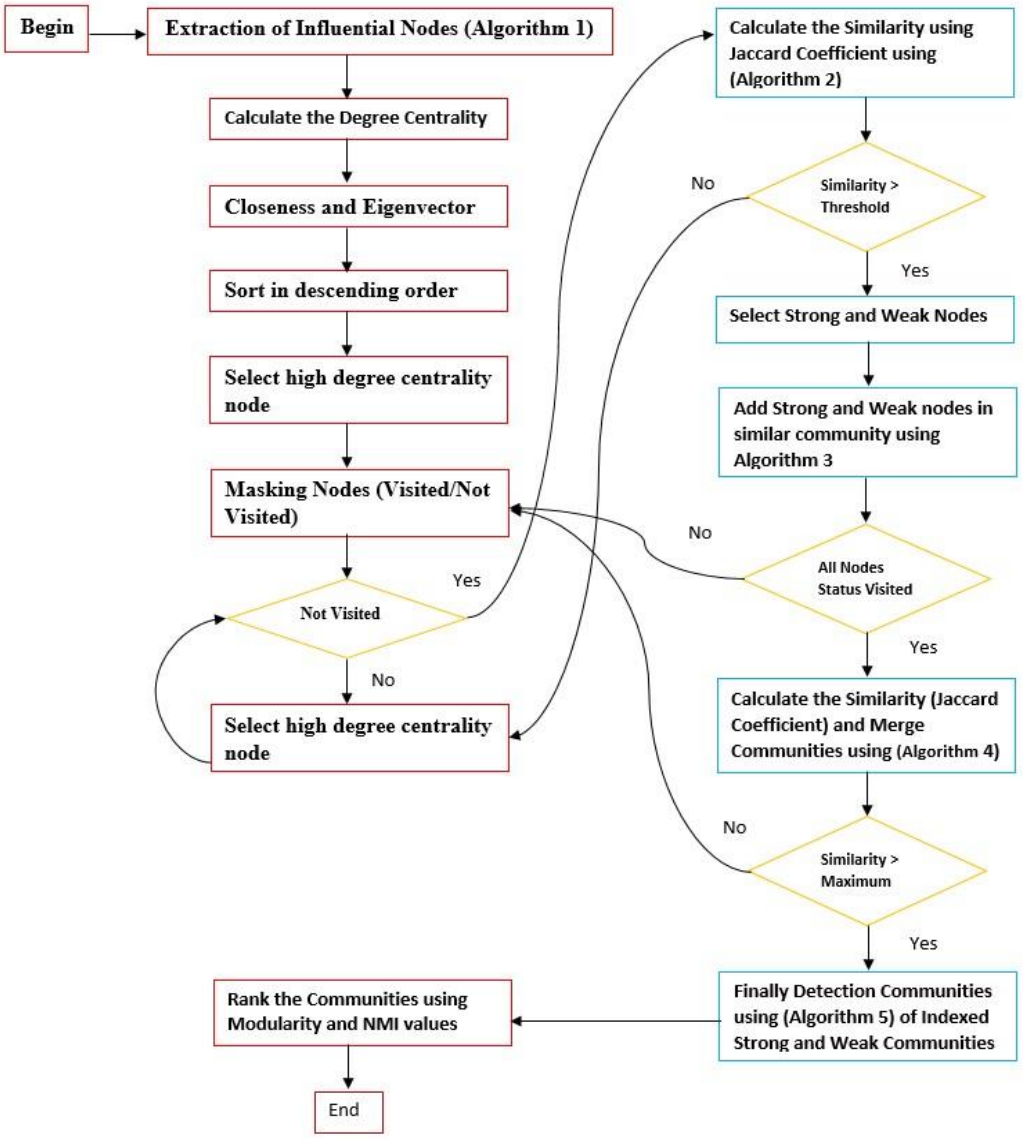

Figure 6. Flow diagram for the selection of the community.

The degree of a node i, represented as d_i, is the count of edges linked to node i, computed as $d_i = \sum A_{ij}$. The degree matrix of the graph G is defined as $D = diag(d_1, d_2, \ldots, d_n)$, where D is a diagonal matrix containing the degrees of each node on its diagonal. The Laplacian matrix L of the G is defined as $L = D - A$. The diagonal elements of L correspond to the degrees of the nodes, $L_{ii} = di$, whereas the off-diagonal elements are represented as $L_{ij} = -A_{ij}$. Given that A is symmetric for an undirected graph, L is likewise symmetric, and the sum of each row in L is zero. The L is recognized as positive semi-definite, indicating that all its eigenvalues, collectively termed the Laplacian spectrum, are nonnegative. We represent the eigenvalues in descending order as in Equation (10).

$$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n \tag{10}$$

The least eigenvalue, λ_n, is consistently zero, with its associated eigenvector being proportional to the vector of ones, $q_n = 1_n$. Conventionally, we normalize the eigenvectors

$q_1, \ldots, q_n$ to create an orthogonal set: each eigenvector possesses unit length $(q_i^T q_i = 1)$, and they are mutually orthogonal $q_i^T q_j = 0$ for $i \neq j$. We present specific notations for clarity. Let q^2 denote the square of the vector q. For the sake of notational simplicity, we define $q_0 = q_n = \sqrt{\frac{1}{n}}$, so that ${q_0}^2 = \frac{1}{n}$.

3.1. Nodes with considerable influence

Identifying influential individuals within a network is crucial, as they play a key role in disseminating information across communities. We begin the community detection process using Algorithm 1, which selects influential nodes based on degree centrality, proximity centrality, and EC. To reduce computational complexity and preserve structural relationships, we use the Laplacian Eigenmap (LE) for dimensionality reduction. This technique projects nodes from a high-dimensional space into a lower-dimensional embedding space. The embedding dimension, denoted by d, determines how much structural information is retained. A smaller value of d results in a more compact representation, enabling efficient detection of influential nodes, while a larger d offers higher fidelity but increases computational cost. Thus, we select an optimal, smaller value of d to balance compression and accuracy. Nodes are first sorted by degree centrality in descending order. The LE method is then applied to construct the embedding space, where each node is represented as a vector. All nodes are initially marked as Not Visited, indicating they have not yet been assigned to any community. This step sets the foundation for the subsequent community detection phases.

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 (5), and (7).
2: Sort the nodes in the graph in descending order.
3: $V_{in} \Leftarrow$ influential nodes
4: Marked the nodes that are not visited.
5: $Status(V_i \epsilon V_{in})$ =Not Visited
6: To obtain embedding space, use the LE method.
7: Embed = Laplacian Eigenmap (G, d) returns influential nodes and embedding space

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.

3.2. Preliminary Community Identification

The similarity is computed using Algorithm 2 within the embedding space generated by the Laplacian Eigenvectors technique, employing the JC metric. Nodes exhibiting significant similarity to the chosen core should be categorized together. To do this, we allocate the core community to all connected nodes whose similarity value surpasses a certain threshold, S. Upon establishing the initial community and designating its members as "Visited," we advance to the next unvisited node with the highest centrality rating.

In summary, the algorithm 2 comprises a sequence of meticulously organized steps:

1. Select the node with the maximum centrality and a state of Not-visited.
2. Calculate the JC 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.

To establish the preliminary community structure, we initially select the node exhibiting the highest degree centrality as the core and compute its similarity with the remaining nodes in the graph. The similarity is calculated in the embedding space generated by the LE

algorithm utilizing the JC measure using Equations (11) – (13). The objective is to cluster nodes that exhibit significant similarity with the chosen core. We allocate all analogous nodes to the core with a similarity value over a specified threshold k to the core's community to accomplish this. Upon the initial community's completion, we designate its members' status as "Visited" and proceed to the next unvisited node exhibiting the highest centrality. We implement the identical procedure on the second designated core and persist until every node in the graph has been designated as "visited." The preliminary community structure is now established.

Algorithm 2. Identify the 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

$P \Leftarrow \{\}$
while $i \leq n$ **do**
  if Status($V_{in}[i]$==Not Visited) **then**
    $Co_i \Leftarrow v_{in}$
    $Cu.\text{append} \Leftarrow Co$
  end if
  while $j = i + 1 \leq n$ **do**
    if Status($V_{in}[j]$==Not Visited) **then**
      $Similarity = JC$(embed(co), embed($V_{in}[j]$))
      if $Similarity > S$ **then**
        $Cu.\text{append} \Leftarrow v_{in}[j]$
        Status $V_{in}[j] \Leftarrow$ Visited
      end if
    end if
  end while
end while
$P.\text{append}(Cu)$
$Co = \{\}$
$PC(initial) \Leftarrow Merge(P)$

3.3. Identification of Robust and Fragile Communities

Weak communities are smaller than external ones; they may consist of singleton communities or exhibit reduced interactions among members. Therefore, certain acquired communities must be merged to achieve the ideal community structure for the graph, as delineated 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 $C_i \epsilon P$ **do**
  if length$\left(C_i < \text{average}(V_i)\right)$or $C_i \nexists$ any 3-clique **then**
    $C_{weak}.\text{append}(C_i)$
  else
    $C_{strong}.\text{append}(C_i)$
    $PC_{final}.\text{append}\left(C_{strong}\right)$
  end if
end while

3.4. Consolidation of Communities

The predominant techniques for optimizing community structure have been articulated in the literature and focus on maximizing or decreasing a specific target function. The subsequent phase, detailed in Algorithm 4, involves identifying deficient communities in the initial community structure and amalgamating them with robust communities.

Algorithm 4. Finally Merging Strong and Weak Communities

Require: $PC(merged) = C_1, C_2, \ldots, C_n$ merged Communities
Ensure: $PC(final)$
while $i \epsilon C_{strong}$ **do**
  $maximum_{similarity} \Leftarrow -2$
  $index_{strong} \Leftarrow 0$
  $index_{weak} \Leftarrow 0$
  while $j \epsilon C_{weak}$ **do** Calculate the JC similarity of each Strong and Weak community using Equations 11, 12, and 13.
    if $similarity(i,j) > maximum_{similarity}$ **then**
      $maximum_{similarity} = similarity(i,j)$
      $index_{strong} \Leftarrow i$
      $index_{weak} \Leftarrow j$
    end if
  end while
end while

To minimize the temporal complexity as much as feasible, we calculated the JC similarity [74] 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 JC similarity between each weak community's nodes and each strong community's core that can be calculated using Equation (11).

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

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 (12), $|N_u|$ and $|N_v|$ represent the cardinality of the sets N_u and N_v, respectively.

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

Suppose $|\bar{N}_u|$, $|\bar{N}_v|$, are the cardinality of the set of items un-co-rated by users u and v respectively. Hence, $|\bar{N}_u| = |N_u| - |N_u \cap N_v|$ and $|\bar{N}_v| = |N_v| - |N_u \cap N_v|$. As a result, the JC similarity can be written as in Equation (13).

$$\begin{aligned} \mathrm{sim}(u,v)^{Jaccard} &= \frac{|N_u \cap N_v|}{(|\bar{N}_u| + |N_u \cap N_v|) + (|\bar{N}_v| + |N_u \cap N_v|) - |N_u \cap N_v|} \\ &= \frac{|N_u \cap N_v|}{|\bar{N}_u| + |\bar{N}_v| + |N_u \cap N_v|} \end{aligned} \tag{13}$$

Modularity is the primary criterion to consider in discussions of community detection. The modularity of a network quantifies its capacity for division into distinct groups[75]. Optimization frameworks employ modularity to identify community networks. This concerns the discrepancy between the actual and expected amounts of edges. The notation used in Equation (14) represents modularity Q.

$$Q = \sum_i e_{ii} - a_i^2 \tag{14}$$

Two communities should be merged if there are more connections between them than 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 (15).

$$l_{ij} = \left|(v_i, v_j) : v_i \in C_i \text{ and } v_j \in C_j\right| \tag{15}$$

We use Equation (16) to determine if, among all the communities in the community setting, the community C_j and C_i should merge.

$$S_{ij} = \frac{l_{ij}}{dc_i dc_j} \tag{16}$$

Algorithm 5. Detecting Community

Require: Final set of communities

Ensure: List of C_{weak}, C_{strong}

$Com \Leftarrow 0$
While for every node u_i in V_j **do**
  Retrieve node based on similarity of u_i as v_j
  if $C_{ui}, C_{vj} \nexists$ **then**
    Create C_{ui} as $C_{ui} = \{u_i, v_j\}$
    $Com \Leftarrow Com \cup \{C_{ui}\}$
  else if $C_{ui} \exists$ and $v_j \nexists$ in any Com_i **then**
    $C_{ui} \Leftarrow C_{ui} \cup \{v_j\}$
  else if $C_{vj} \exists$ and $u_i \nexists$ in any Com_i **then**
    $C_{vj} \Leftarrow C_{vj} \cup \{u_i\}$
  end if
  repeat
end while
Compute modularity Q by using Equations (8), (15), and (16).
Select Com_i and Com_j such that $St_{ij} = max[St_{mn}: Com_n, Com_n] \epsilon Com$ using Equation (16)
Compute Modularity Qm_j for $Com - Com_i, Com_j \cup Com_i \cup Com_j$
if $Qm_j > Qm$ **then**
  $Com_i \Leftarrow Com_i \cup Com_j$
  $Com \Leftarrow Com - \{Com_i, Com_j\}$
  $Com_i \Leftarrow Com \cup Com_i$
end if till modularity does not show any improvement

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.

3.5. Performance Evaluation Metrics

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

3.5.1. Modularity

The preliminary metric is well-acknowledged in the literature. This method contrasts the real connections within a community against the probability of encountering those links in a randomly generated network. The efficacy of a network is maximized when there is a high concentration of connections within communities and a low concentration of connections between communities. The division with the highest modularity score is considered the most optimal in this context. Equation (17) delineates the modularity of division D given a graph G as Equation (17).

$$Q(D) = \sum_{i=1}^{|D|} (e_{ii} - a_i^2) \tag{17}$$

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 normalized mutually (NMI), normalized to a number between 0 and 1, determines the amount of information about two variables. The NMI is calculated using Equation (18), 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 $\log P_{UV}(i,j)P_U(i)P_V(j)$. Values approaching 1 imply a robust connection between two variables, while values approaching 0 signify a feeble one.

$$NMI(U,V) = \frac{2\sum_{i=1}^{R}\sum_{j=1}^{C} P_{UV}(i,j)\log\frac{P_{UV}(i,j)}{P_U(i)P_V(j)}}{-\sum_{i=1}^{R} P_U(i)\log P_U(i) - \sum_{i=1}^{R} P_v(i)\log P_v(i)} \tag{18}$$

3.5.2. Clustering Coefficient

We now give the primary observation utilized in both network average and global grouping coefficient estimators. Equations (19)-(21), shows the initial equality is a consequence of the law of total expectation. The second equality is valid because there are 2 equal probability combinations $x_k - 1, v_i, x_k + 1$, of which only 2 form a triangle v_j,v_i,v_k or a reverse triangle v_k,v_i,v_j. Observe that in a triangle or a reverse triangle, $v_j$is connected to $v_k(A_j, k = 1)$. The third equality is established using algebraic manipulation.

$$\mathbb{E}[\phi_k f(x_k)] = \sum_{i=1}^{n} p_i\, \mathbb{E}[\phi_k f(x_k) | x_k = i] \tag{19}$$

$$= \sum_{i=1}^{n} \frac{d_i}{D}\frac{2l_i}{d_i^2} f(v_i) \tag{20}$$

$$= \sum_{i=1}^{n} \frac{1}{D}\frac{2l_i}{d_i} f(v_i) \tag{21}$$

4. Experiments and Results

This section delineates the datasets and evaluates the performance of the most advanced community detection algorithms. The primary objective of this axis is to conduct an experimental investigation to determine the feasibility of our idea. We achieve this by evaluating the model's efficacy on both simulated and actual networks. We employ industry-standard metrics, such as modularity and NMI, as performance indicators.

4.1. Experimental Setup

Table 3 shows the hardware requirements for the proposed model. The code is written in Python, and the other techniques were executed using the Python igraph package[76]. The network module in Python[77] is utilized to enhance the visualization of the detected communities.

Table 3. Experimental Settings.

Configuration	Parameters
CPU	Intel(R) Xeon(R) Gold 6154
GPU	NVidia GeForce RTX 3090
Operating System	Windows 10
Environment	CUDA 11.1
Development Platform	Jupiter Notebook
Python Library	PyTorch
Video Memory	32GB
System Drive	1TB

Table 4 delineates hyperparameter setups for graph clustering, specifying essential parameters and their standard values, which will be fine-tuned for experimental evaluation.

Table 4. Experimental Settings.

Hyperparameter	Values / Ranges
Number of Clusters (k)	Elbow Method
Graph Representation	Adjacency Matrix
Embedding Dimension	16, 32, 64, 128
Clustering Algorithm	k-means, Louvain
Distance Metric	Jaccard
GNN Layers	4
Learning Rate	1×10^{-3}
Epochs	100
Batch Size	64
Regularization (λ)	1×10^{-4}
Dropout Rate	0.5
Edge Weighting Strategy	TF-IDF
Walk Length	20
Number of Walks (Node2Vec)	10
P, Q Parameters	P: 0.30, Q: 0.33
Evaluation Metric	Modularity, NMI

As a note edge weights are computed using TF-IDF similarity based on user profile attributes (e.g., shared keywords or descriptions). This strategy enriches the structural representation of the graph with semantic context.

4.2. Real World Datasets

Table 5 presents a summary of the six real-world data sets used.

Table 5. Real-world datasets.

Ref	Network	No of Edges(m)	No of Nodes(n)	No of Classes(c)
[78]	Karate	78	34	2
[79]	Dolphins	159	62	2
[50]	Football	613	115	12
[80]	Amazon	925872	334863	75149
[80]	DBLP	1049866	317080	13477
[81]	Ego-Facebook	4039	88234	13

1. The network of Zachary's Karate Club is examined in the study by [78]. Zachary established a concrete network by leveraging the social relationships among the 34 members of a karate group. Due to a political dispute between the club's leadership and the instructor, the network has been divided into two portions. For this investigation, we employ the fundamental iteration of this network.
2. Between 1994 and 2001, a SN known as The Dolphin's Network [79] documented those 62 bottlenose dolphins seen in New Zealand consistently established associations with each other. Two groups exist inside the network.
3. As per [50], the College Football Network presents the 2000 college football schedule, where teams are denoted as vertices and matches between two teams are indicated by edges. Among the 115 vertices in the network, twelve constitute coalitions.
4. A co-authorship network called the Digital Bibliography and Library Project (DBLP) [80] connects authors who have collaborated on at least one paper.
5. An individual's SN is represented by the Ego-Facebook Network Data Set, which comprises groups from Facebook networks (also known as friends lists). With Facebook users as vertices and various relationships between 10 ego-networks as edges, it contains 4039 nodes and 88234 edges, each with 193 ground-truth circles [81].

4.3. Experimental Results

To enhance the proposed algorithm's performance, several refinements are applied. The process begins with the identification of influential nodes using the Eigenmaps method. These nodes are embedded into a lower-dimensional space, where the value of d directly impacts community recognition and model effectiveness [40], [42]. The dimension d is determined based on the structure of the target network. In the next step, nodes are clustered based on their vector similarity in the embedding space to form an initial community structure. A node is assigned to a core community if its similarity to a central node exceeds 0.8, a threshold consistently used across all tested networks [44]. Using Algorithm 5, we evaluated the proposed model on six real-world datasets with known community structures. The performance was assessed using modularity and NMI metrics. Results are shown in Tables 6 and 7 and further visualized in Figures 7 to 9. Each network's outcome was reported individually, and ground truth comparisons were highlighted using distinct community colorings.

Table 6. Modularity values obtained in real-life datasets with ground truth.

Network	Louvain [47]	Spinglass [82]	Walktrap [83]	FLPA[20]	Girvan Newman [50]	Infomap [84]	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

Table 7. NMI values obtained in real-life datasets with ground truth.

Network	Louvain [47]	Spinglass [82]	Walktrap [83]	FLPA[20]	Girvan Newman [50]	Infomap [84]	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

In the case of the Karate Club network (Figure 7), although the proposed method yielded more than the two actual communities, it achieved higher modularity and NMI than traditional methods[34], [36]. Similarly, for the Dolphins network (Figure 8), the method identified six communities (versus four in the ground truth), yet outperformed Louvain [47]and Infomap [84] in both NMI and modularity. Our approach consistently delivered better results in modularity and NMI, even when the number of discovered communities differed from the ground truth.

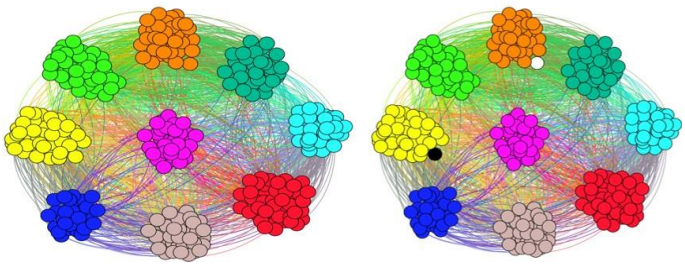

Figure 7. Karate Club Community structure (**Left**) Ground Truth (Right) Community Structure detected by Hybrid Model

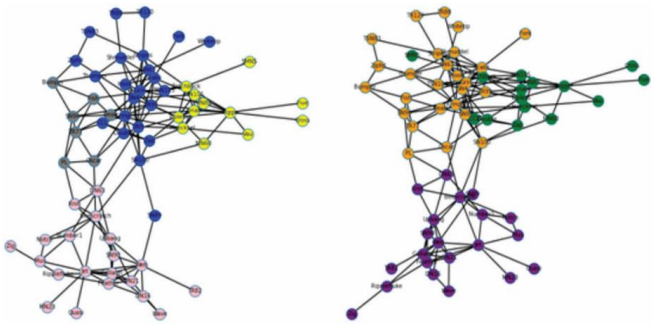

Figure 8. The hybrid model identified the dolphin community structure (left) versus the ground truth (right) community structure.

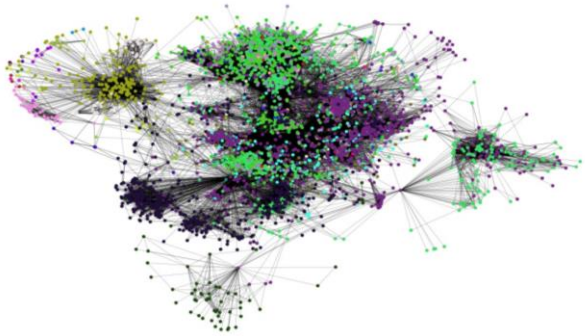

Figure 9. Clustering of different Communities.

As summarized in Table 8, our hybrid model demonstrated higher accuracy and stability than other techniques, such as Louvain [47], FLPA [20], and Spinglass [82], due to its deterministic nature and robustness in dense graphs. The resulting community structures, illustrated in Figure 9, reflect the hybrid model's superior ability to detect cohesive clusters. In particular, the method quickly and reliably identifies communities closely aligned with real-world groupings, consistent with findings in [17].

Table 8. Final ranking for the compared algorithms in terms of NMI on six Datasets.

Network	Louvain [47]	Spinglass [82]	Walktrap [83]	FLPA[20]	Girvan Newman [50]	Infomap [84]	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

The experiment incorporates several baseline methods, including classical community detection algorithms and established clustering techniques, to demonstrate the advantages of the proposed model. To assess its effectiveness, the model is compared against existing embedding-based baselines and graph embedding techniques for community detection (CD), as referenced in [1]–[3]. Community structures are derived from the low-dimensional representations generated by Algorithm 5. The evaluation uses modularity (Q) and Normalized Mutual Information (NMI) metrics, following the standards set in[6], [8], [85]. The proposed method achieves the highest NMI scores across all six real-world datasets and the highest modularity scores in five out of six cases. While the modularity value (Q) for the Ego-Facebook and Dolphins datasets shows slight deviations from optimal, the method still reaches an NMI value of 1, indicating perfect alignment with ground truth communities. Notably, on the DBLP dataset, the proposed model outperforms prior approaches [4], [5], achieving the top

NMI score of 0.91. The Amazon dataset also exhibits strong performance, with both high Q and an NMI close to 1, reflecting strong correspondence with real-world community structures. Overall, the DBLP results are the most favorable, showcasing the logical consistency and effectiveness of the proposed strategy over alternative algorithms, particularly those that yielded the lowest modularity scores.

4.4. Community Visualization and Evaluation Results

In Figure 10, the karate club network was analyzed using the suggested model, yielding three communities, although the ground truth had only two communities. However, the modularity and NMI surpass those of the other algorithms. Figure 11 illustrates the football social network, revealing nine identified communities, but twelve communities existed. The modularity and NMI for the communities identified by the proposed model are superior to those of existing algorithms. Although the quantity of identified communities varies from the ground truth, our model demonstrates superior modularity and NMI.

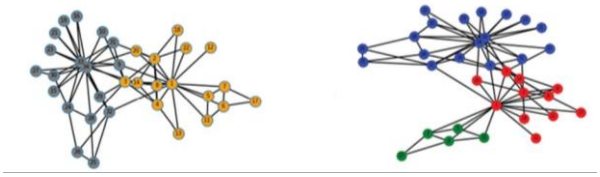

Figure 10. (**left**) Ground Truth Karate Club Community;(**right**) Karate Club Community detected by the proposed method

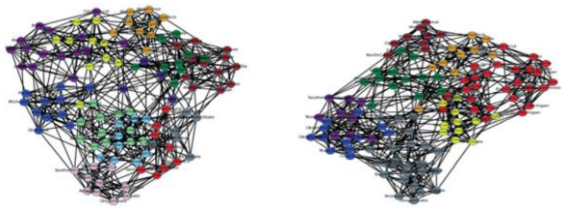

Figure 11. (**left**) Ground truth football network community;(**right**) Football network community detected by the proposed method

4.5. Temporal Complexity

The greedy method requires $O(n+m)$ for each iteration. CN iterations were required for this method to function. Consequently, the overall time complexity of this algorithm was $O(m+n)(n)$. 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 (22).

$$T(n) = (m \cdot d_w^2) + O(n^2) + O(ns.nw) \tag{22}$$

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

4.6. Parameter Settings and Calculation of Similarity (S) and Distance (D)

Graph clustering divides a network into groups where nodes are more connected internally than those outside the cluster. User interaction patterns help identify communities, interests, or anomalies. This process uses similarity (S) and distance (D) metrics—particularly the JC—to measure the strength of node relationships. High S values indicate strong interactions, while high D values imply weak or distant connections. The proposed model applies Algorithm 1 to identify influential nodes and Algorithm 2 to form initial community structures using JC similarity. Algorithms 3 and 4 classify and merge strong and weak communities. Equations 7–16 define key metrics used throughout the model. To optimize performance and reduce computational complexity, JC similarity is computed specifically between core nodes of strong communities and members of weak ones. Hyperparameter tuning is performed to enhance detection quality further.

4.7. Discussion

Table 9 presents the results of experiments comparing centrality measures—degree, eigenvector, betweenness, closeness (proximity), and clustering coefficient—based on their NMI scores across six datasets. NMI (Normalized Mutual Information) is used to evaluate how well the detected communities align with the ground truth, making it a reliable metric for clustering performance. The results indicate that degree centrality (NMI: 0.96–0.99) and EC (0.94–0.97) are highly effective, showing that nodes with many or influential connections strongly contribute to accurate community detection. Similarly, betweenness and closeness centrality (0.99 and 0.98) support the role of bridge nodes in shaping interaction patterns. However, the clustering coefficient produces inconsistent outcomes (−0.38 to 0.99), suggesting that local density does not always translate to meaningful clusters.

Table 9. Comparison of different centrality measures with Eigenvector and Closeness Centrality for Football real world dataset.

	Degree	Eigenvector	Betweenness	Closeness	Clustering coefficient
Degree	0.96	-	-	-	-
Eigenvector	0.99	0.94	-	-	-
betweenness	0.98	0.97	0.99	0.95	-
Closeness	0.93	0.74	0.94	0.98	-
Clustering coefficient	-0.38	-0.34	-0.91	0.06	0.99

These findings in Table 9 emphasize the importance of selecting suitable centrality measures, especially in domains like social network analysis, recommender systems, or fraud detection. Degree and betweenness are reliable indicators for identifying key influencers or hubs. Table 10 further explores the interrelations among degree, betweenness, and clustering coefficient, revealing strong correlations (0.85–0.86) between degree and betweenness but negative correlations between clustering coefficient and both metrics. Without eigenvector and proximity centrality, the model loses insight into globally influential nodes—those connected not just frequently but also to other important nodes. EC is essential in identifying central figures in complex or hierarchical networks, such as top influencers or decision-makers.

Table 10. Comparison of different centrality measures without Eigenvector and Closeness Centrality for Football real world dataset.

	Degree	Betweenness	Clustering coefficient
Degree	0.86	0.78	0.86
betweenness	0.85	0.77	0.87
Clustering coefficient	-0.47	-0.42	-0.98

4.8. Performance Comparison and Limitations of the Proposed Model

This study evaluates the proposed model against existing methods using key metrics such as modularity, NMI, degree, betweenness, and clustering coefficient, as summarized in Table 11.

Table 11. Comparison of modularity and NMI for different approaches for utilized Football real world dataset.

Ref	Modularity	NMI	Degree	Betweenness	Clustering Coefficient
Palacio [56]	91	90	81	82	80
Yuan [58]	92	89	82	74	85
Farrokh [61]	86	89	81	83	84
Werner [64]	85	88	83	89	85
Cao [72]	89	87	84	78	74
Proposed	95.05	92.50	91.62	90.65	86

The proposed model achieves the highest modularity (95.05%) and NMI (92.50%), indicating more effective community structure detection and stronger alignment with ground truth than prior methods. It also shows superior performance in identifying hub (degree) and bridge (betweenness) nodes, which are essential for network integrity. While its clustering coefficient (86%) is slightly behind some models, it remains competitive. These results confirm the model's robustness and accuracy across various network analysis dimensions.

Despite its strengths, the model has limitations. EC may overemphasize dominant hubs, potentially overlooking smaller but meaningful communities. Its reliance on eigenvector and proximity computations also increases complexity, especially in large-scale graphs. Additionally, nodes with fewer connections may be underrated despite their potential importance within the network. These trade-offs highlight the need for careful tuning and possible enhancements in future iterations.

5. Conclusions

The rise of social networks has reshaped how individuals interact, forming virtual communities based on shared interests and behaviors. Detecting these communities is essential for understanding complex social structures. In response to real-world networks' increasing size and complexity, this study introduces a novel five-phase hybrid model that integrates node space similarity, EC, closeness centrality, and graph embeddings. Unlike traditional approaches, the proposed method combines global and local network features to enhance community detection accuracy. It begins by identifying influential nodes and expands outward to form cohesive clusters. Experimental results on real-world datasets demonstrate the model's superior performance, achieving modularity scores of 95.05% (DBLP) and 94.50% (Amazon), and NMI scores of 91.80% (DBLP), 92.50% (Amazon), and 90.43% (Football)—surpassing existing methods.

5.1 Future work, Development Plans

To further improve our model and its applicability, future research will focus on:

- In the future, we will improve clustering techniques to augment the precision and clarity of identified groups and user interactions.
- This study will implement dynamic graph clustering that incorporates user interactions over time.
- This study will utilize hybrid models that combine standard clustering methods, such as k-means clustering, with deep learning-based embeddings, such as Graph Neural Networks, to enhance the quality and performance of identified communities.
- This study will enhance our community detection model by adjusting various parameter values and employing reinforcement learning to fine-tune hyper-parameters, improving cluster identification.

5.2 Limitations

The caliber of the user interaction data may influence the calculated centrality metrics. Inaccurate or insufficient data may result in erroneous conclusions. Online platforms may limit access to user interaction data due to privacy constraints. When gathering and employing this data type, it is essential to contemplate ethical implications. It can take a lot of computing power for large user interaction networks to figure out proximity centrality and EC. These constraints may restrict the scalability of the method for practical applications. Careful feature engineering may be needed to ensure that centrality ratings significantly improve the clustering process when added to more user data. User interaction networks are fluid, and communities evolve. The selected communities may not accurately correspond to the latest network architecture.

Author Contributions: H.A. M.S.; methodology, H.A. M.S. and F.M.; software, H.A. and F.M.; validation, F.M. M.S.; formal analysis, M.S. and H.A.; investigation, F.M.; resources, M.S. and F.M; data curation, F.M and M.S.; writing—original draft preparation, H.A.; writing – review and editing, M.S. and F.M. funding acquisition, M.S. All authors have read and agreed to the published version of the manuscript

Funding: No specific funding was received for this research study

Data Availability Statement: All data supporting this study are openly available within the paper and online as open source, without restrictions

Conflicts of Interest: The authors have declared that no competing interests exist

References

[1] J. O. Bolorunduro and Z. Zou, "Community Detection On Multi-layer Graph using Intra-layer and Inter-layer Linkage Graphs (CDMIILG)," *Expert Syst. Appl.*, vol. 238, p. 121713, Mar. 2024, doi: 10.1016/j.eswa.2023.121713.

[2] C. Yang, J. Liu, Y. Yan, and C. Shi, "FairSIN: Achieving Fairness in Graph Neural Networks through Sensitive Information Neutralization," *Proc. AAAI Conf. Artif. Intell.*, vol. 38, no. 8, pp. 9241–9249, Mar. 2024, doi: 10.1609/aaai.v38i8.28776.

[3] L. Gadár and J. Abonyi, "Finding multifaceted communities in multiplex networks," *Sci. Rep.*, vol. 14, no. 1, p. 14521, Jun. 2024, doi: 10.1038/s41598-024-65049-6.

[4] A. P. Christensen, L. E. Garrido, K. Guerra-Peña, and H. Golino, "Comparing community detection algorithms in psychometric networks: A Monte Carlo simulation," *Behav. Res. Methods*, vol. 56, no. 3, pp. 1485–1505, Jun. 2023, doi: 10.3758/s13428-023-02106-4.

[5] R. Cavoretto, A. De Rossi, S. Lancellotti, and F. Romaniello, "Node-bound communities for partition of unity interpolation on graphs," *Appl. Math. Comput.*, vol. 467, p. 128502, Apr. 2024, doi: 10.1016/j.amc.2023.128502.

[6] Y. Xia, Q. Xu, J. Fang, R. Tang, and P. Du, "Bipartite graph-based community-to-community matching in local energy market considering socially networked prosumers," *Appl. Energy*, vol. 353, p. 122245, Jan. 2024, doi: 10.1016/j.apenergy.2023.122245.

[7] B. Baruah, M. P. Dutta, S. Banerjee, and D. K. Bhattacharyya, "A novel density based community detection algorithm and its application in detecting potential biomarkers of ESCC," *J. Comput. Sci.*, vol. 81, p. 102344, Sep. 2024, doi: 10.1016/j.jocs.2024.102344.

[8] J. Zheng, S. Wang, D. Li, and B. Zhang, "Personalized recommendation based on hierarchical interest overlapping community," *Inf. Sci. (Ny).*, vol. 479, pp. 55–75, Apr. 2019, doi: 10.1016/j.ins.2018.11.054.

[9] L. S. Yilmaz and A. J. Walhout, "Metabolic network modeling with model organisms," *Curr. Opin. Chem. Biol.*, vol. 36, pp. 32–39, Feb. 2017, doi: 10.1016/j.cbpa.2016.12.025.

[10] M. E. J. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Phys. Rev. E*, vol. 69, no. 2, p. 026113, Feb. 2004, doi: 10.1103/PhysRevE.69.026113.

[11] S. Ahajjam, M. El Haddad, and H. Badir, "A new scalable leader-community detection approach for community detection in social networks," *Soc. Networks*, vol. 54, pp. 41–49, Jul. 2018, doi: 10.1016/j.socnet.2017.11.004.

[12] M. Azaouzi, D. Rhouma, and L. Ben Romdhane, "Community detection in large-scale social networks: state-of-the-art and future directions," *Soc. Netw. Anal. Min.*, vol. 9, no. 1, p. 23, Dec. 2019, doi: 10.1007/s13278-019-0566-x.

[13] M. Rostami, V. Farrahi, S. Ahmadian, S. Mohammad Jafar Jalali, and M. Oussalah, "A novel healthy and time-aware food recommender system using attributed community detection," *Expert Syst. Appl.*, vol. 221, p. 119719, Jul. 2023, doi: 10.1016/j.eswa.2023.119719.

[14] M. Rostami, M. Oussalah, K. Berahmand, and V. Farrahi, "Community Detection Algorithms in Healthcare Applications: A Systematic Review," *IEEE Access*, vol. 11, pp. 30247–30272, 2023, doi: 10.1109/ACCESS.2023.3260652.

[15] M. Rostami, U. Muhammad, S. Forouzandeh, K. Berahmand, V. Farrahi, and M. Oussalah, "An effective explainable food recommendation using deep image clustering and community detection," *Intell. Syst. with Appl.*, vol. 16, p. 200157, Nov. 2022, doi: 10.1016/j.iswa.2022.200157.

[16] N. Alotaibi and D. Rhouma, "A review on community structures detection in time evolving social networks," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 34, no. 8, pp. 5646–5662, Sep. 2022, doi: 10.1016/j.jksuci.2021.08.016.

[17] M. Xu, "Understanding Graph Embedding Methods and Their Applications," *SIAM Rev.*, vol. 63, no. 4, pp. 825–853, Jan. 2021, doi: 10.1137/20M1386062.

[18] P. Kumar, R. Jain, S. Chaudhary, and S. Kumar, "Solving Community Detection in Social Networks: A comprehensive study," in *2021 5th International Conference on Computing Methodologies and Communication (ICCMC)*, Apr. 2021, pp. 239–345. doi: 10.1109/ICCMC51019.2021.9418412.

[19] R. Bhattacharya, N. K. Nagwani, and S. Tripathi, "A community detection model using node embedding approach and graph convolutional network with clustering technique," *Decis. Anal. J.*, vol. 9, p. 100362, Dec. 2023, doi: 10.1016/j.dajour.2023.100362.

[20] V. A. Traag and L. Šubelj, "Large network community detection by fast label propagation," *Sci. Rep.*, vol. 13, no. 1, p. 2701, Feb. 2023, doi: 10.1038/s41598-023-29610-z.

[21] M. Rabiei, M. Fartash, and S. Nazari, "A hybrid optimization approach for graph embedding: leveraging Node2Vec and grey wolf optimization," *J. Supercomput.*, vol. 81, no. 4, p. 514, Feb. 2025, doi: 10.1007/s11227-025-07022-2.

[22] X. Wu *et al.*, "Graph reconstruction and attraction method for community detection," *Appl. Intell.*, vol. 55, no. 5, p. 357, Apr. 2025, doi: 10.1007/s10489-024-05858-4.

[23] A. Ullah and Y. Meng, "Finding influential nodes via graph embedding and hybrid centrality in complex networks," *Chaos, Solitons & Fractals*, vol. 194, p. 116151, May 2025, doi: 10.1016/j.chaos.2025.116151.

[24] Iman Hasan Abed and S. Bahadori, "Using Density Criterion and Increasing Modularity to Detect Communities in Complex Networks," *InfoTech Spectr. Iraqi J. Data Sci.*, vol. 2, no. 1, pp. 1–15, Jan. 2025, doi: 10.51173/ijds.v2i1.12.

[25] G. Chen and S. Zhou, "A novel overlapping community detection strategy based on Core-Bridge seeds," *Int. J. Mach. Learn. Cybern.*, vol. 15, no. 6, pp. 2131–2147, Jun. 2024, doi: 10.1007/s13042-023-02020-3.

[26] L. C. Freeman, "A Set of Measures of Centrality Based on Betweenness," *Sociometry*, vol. 40, no. 1, p. 35, Mar. 1977, doi: 10.2307/3033543.

[27] L. C. Freeman, "Centrality in social networks conceptual clarification," *Soc. Networks*, vol. 1, no. 3, pp. 215–239, Jan. 1978, doi: 10.1016/0378-8733(78)90021-7.

[28] S. P. Borgatti and M. G. Everett, "A Graph-theoretic perspective on centrality," *Soc. Networks*, vol. 28, no. 4, pp. 466–484, Oct. 2006, doi: 10.1016/j.socnet.2005.11.005.

[29] A. Bavelas, "Communication Patterns in Task-Oriented Groups," *J. Acoust. Soc. Am.*, vol. 22, no. 6, pp. 725–730, Nov. 1950, doi: 10.1121/1.1906679.

[30] A. Bihari and M. K. Pandia, "Eigenvector centrality and its application in research professionals' relationship network," in *2015 International Conference on Futuristic Trends on Computational Analysis and Knowledge Management (ABLAZE)*, Feb. 2015, pp. 510–514. doi: 10.1109/ABLAZE.2015.7154915.

[31] H. Shiokawa, Y. Fujiwara, and M. Onizuka, "SCAN++: efficient algorithm for finding clusters, hubs and outliers on large-scale graphs," *Proc. VLDB Endow.*, vol. 8, no. 11, pp. 1178–1189, Jul. 2015, doi: 10.14778/2809974.2809980.

[32] H. Gmati, A. Mouakher, A. Gonzalez-Pardo, and D. Camacho, "A new algorithm for communities detection in social networks with node attributes," *J. Ambient Intell. Humaniz. Comput.*, vol. 15, no. 2, pp. 1779–1791, Feb. 2024, doi: 10.1007/s12652-018-1108-5.

[33] L. Guerreiro, F. N. Silva, and D. R. Amancio, "Identifying the perceived local properties of networks reconstructed from biased random walks," *PLoS One*, vol. 19, no. 1, p. e0296088, Jan. 2024, doi: 10.1371/journal.pone.0296088.

[34] M. El-Moussaoui, T. Agouti, A. Tikniouine, and M. El Adnani, "A comprehensive literature review on community detection: Approaches and applications," *Procedia Comput. Sci.*, vol. 151, pp. 295–302, 2019, doi: 10.1016/j.procs.2019.04.042.

[35] J. Mothe, K. Mkhitaryan, and M. Haroutunian, "Community detection: Comparison of state of the art algorithms," in *2017 Computer Science and Information Technologies (CSIT)*, Sep. 2017, pp. 125–129. doi: 10.1109/CSITechnol.2017.8312155.

[36] M. E. J. Newman, "Fast algorithm for detecting community structure in networks," *Phys. Rev. E*, vol. 69, no. 6, p. 066133, Jun. 2004, doi: 10.1103/PhysRevE.69.066133.

[37] S. R. Chintalapudi and M. H. M. K. Prasad, "Finding research groups using modularity based community detection algorithm," in *2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)*, 2016, pp. 65–69. [Online]. Available: https://ieeexplore.ieee.org/document/7724229

[38] R. K. Behera, S. K. Rath, and M. Jena, "Spanning Tree Based Community Detection Using Min-Max Modularity," *Procedia Comput. Sci.*, vol. 93, pp. 1070–1076, 2016, doi: 10.1016/j.procs.2016.07.311.

[39] S. G. Krishnan, S. Karthika, and S. Bose, "Detection of communities in dynamic social networks," in *2016 International Conference on Recent Trends in Information Technology (ICRTIT)*, Apr. 2016, pp. 1–6. doi: 10.1109/ICRTIT.2016.7569567.

[40] T. Jiang, C. Zhang, H. Zhu, and G. Deng, "Energy-efficient scheduling for a job shop using grey wolf optimization algorithm with double-searching mode," *Math. Probl. Eng.*, vol. 2018, no. 1, p. 8574892, 2018.

[41] I. Gutiérrez, D. Gómez, J. Castro, and R. Espínola, "A New Community Detection Problem Based on Bipolar Fuzzy Measures," in *Computational Intelligence and Mathematics for Tackling Complex Problems 2*, Springer, 2022, pp. 91–99. doi: 10.1007/978-3-030-88817-6_11.

[42] M. Huang, Q. Jiang, Q. Qu, and A. Rasool, "An Overlapping Community Detection Approach in Ego-Splitting Networks Using Symmetric Nonnegative Matrix Factorization," *Symmetry (Basel).*, vol. 13, no. 5, p. 869, May 2021, doi: 10.3390/sym13050869.

[43] J. Zhu *et al.*, "A no self-edge stochastic block model and a heuristic algorithm for balanced anti-community detection in networks," *Inf. Sci. (Ny).*, vol. 518, pp. 95–112, May 2020, doi: 10.1016/j.ins.2020.01.005.

[44] Y. Yin, Y. Zhao, H. Li, and X. Dong, "Multi-objective evolutionary clustering for large-scale dynamic community detection," *Inf. Sci. (Ny).*, vol. 549, pp. 269–287, Mar. 2021, doi: 10.1016/j.ins.2020.11.025.

[45] M. A. Tabarzad and A. Hamzeh, "A heuristic local community detection method (HLCD)," *Appl. Intell.*, vol. 46, no. 1, pp. 62–78, Jan. 2017, doi: 10.1007/s10489-016-0824-9.

[46] X. Zhang, C. Wang, Y. Su, L. Pan, and H.-F. Zhang, "A Fast Overlapping Community Detection Algorithm Based on Weak Cliques for Large-Scale Networks," *IEEE Trans. Comput. Soc. Syst.*, vol. 4, no. 4, pp. 218–230, Dec. 2017, doi: 10.1109/TCSS.2017.2749282.

[47] X. Zhou, K. Yang, Y. Xie, C. Yang, and T. Huang, "A novel modularity-based discrete state transition algorithm for community detection in networks," *Neurocomputing*, vol. 334, pp. 89–99, Mar. 2019, doi: 10.1016/j.neucom.2019.01.009.

[48] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *J. Stat. Mech. Theory Exp.*, vol. 2008, no. 10, p. P10008, Oct. 2008, doi: 10.1088/1742-5468/2008/10/P10008.

[49] V. A. Traag, L. Waltman, and N. J. Van Eck, "From Louvain to Leiden: guaranteeing well-connected communities," *Sci. Rep.*, vol. 9, no. 1, pp. 1–12, 2019.

[50] M. Girvan and M. E. J. Newman, "Community structure in social and biological networks," *Proc. Natl. Acad. Sci.*, vol. 99, no. 12, pp. 7821–7826, Jun. 2002, doi: 10.1073/pnas.122653799.

[51] A. K. Ghoshal, N. Das, S. Bhattacharjee, and G. Chakraborty, "A Fast Parallel Genetic Algorithm Based Approach for Community Detection in Large Networks," in *2019 11th International Conference on Communication Systems & Networks (COMSNETS)*, Jan. 2019, pp. 95–101. doi: 10.1109/COMSNETS.2019.8711127.

[52] B. Alhijawi and A. Awajan, "Genetic algorithms: theory, genetic operators, solutions, and applications," *Evol. Intell.*, vol. 17, no. 3, pp. 1245–1256, Jun. 2024, doi: 10.1007/s12065-023-00822-6.

[53] X. Zeng, W. Wang, C. Chen, and G. G. Yen, "A Consensus Community-Based Particle Swarm Optimization for Dynamic Community Detection," *IEEE Trans. Cybern.*, vol. 50, no. 6, pp. 2502–2513, Jun. 2020, doi: 10.1109/TCYB.2019.2938895.

[54] H. C. Rustamaji *et al.*, "A network analysis to identify lung cancer comorbid diseases," *Appl. Netw. Sci.*, vol. 7, no. 1, p. 30, Dec. 2022, doi: 10.1007/s41109-022-00466-y.

[55] S. Fortunato and M. Barthélemy, "Resolution limit in community detection," *Proc. Natl. Acad. Sci.*, vol. 104, no. 1, pp. 36–41, Jan. 2007, doi: 10.1073/pnas.0605965104.

[56] J.-O. Palacio-Niño and F. Berzal, "Enhancing Community Detection in Networks: A Comparative Analysis of Local Metrics and Hierarchical Algorithms," *arXiv*. Aug. 16, 2024. [Online]. Available: http://arxiv.org/abs/2408.09072

[57] J. Guo, P. Singh, and K. E. Bassler, "Resolution limit revisited: community detection using generalized modularity density," *J. Phys. Complex.*, vol. 4, no. 2, p. 025001, Jun. 2023, doi: 10.1088/2632-072X/acc4a4.

[58] L. Yuan, X. Zhang, Y. Ke, Z. Lu, X. Li, and C. Liu, "Temporal Community Detection and Analysis with Network Embeddings," *Mathematics*, vol. 13, no. 5, p. 698, Feb. 2025, doi: 10.3390/math13050698.

[59] A. Miyauchi and Y. Kawase, "Z-Score-Based Modularity for Community Detection in Networks," *PLoS One*, vol. 11, no. 1, p. e0147805, Jan. 2016, doi: 10.1371/journal.pone.0147805.

[60] S. Fortunato and D. Hric, "Community detection in networks: A user guide," *Phys. Rep.*, vol. 659, pp. 1–44, Nov. 2016, doi: 10.1016/j.physrep.2016.09.002.

[61] Y. Farrokhpour Dizaj, S. Lotfi, and J. Hajipour, "A multi-objective social-based algorithm for community detection," *Multimed. Tools Appl.*, pp. 1–30, Jan. 2025, doi: 10.1007/s11042-024-20555-7.

[62] K. He, Y. Li, S. Soundarajan, and J. E. Hopcroft, "Hidden community detection in social networks," *Inf. Sci. (Ny).*, vol. 425, pp. 92–106, Jan. 2018, doi: 10.1016/j.ins.2017.10.019.

[63] S. Chattopadhyay and D. Ganguly, "Node2vec with weak supervision on community structures," *Pattern Recognit. Lett.*, vol. 150, pp. 147–154, Oct. 2021, doi: 10.1016/j.patrec.2021.06.024.

[64] M. A. Werner, J. de Ron, E. I. Fried, and D. J. Robinaugh, "Iterated community detection in psychological networks.," *Psychol. Methods*, Mar. 2025, doi: 10.1037/met0000744.

[65] A. Ibrahem Hafez, A. ella Hassanien, A. A. Fahmy, and M. F. Tolba, "Community detection in social networks by using Bayesian network and Expectation Maximization technique," in *13th International Conference on Hybrid Intelligent Systems (HIS 2013)*, Dec. 2013, pp. 209–214. doi: 10.1109/HIS.2013.6920484.

[66] S. Srinivas and C. Rajendran, "Community detection and influential node identification in complex networks using mathematical programming," *Expert Syst. Appl.*, vol. 135, pp. 296–312, Nov. 2019, doi: 10.1016/j.eswa.2019.05.059.

[67] J. Cheng *et al.*, "Neighbor Similarity Based Agglomerative Method for Community Detection in Networks," *Complexity*, vol. 2019, no. 1, p. 8292485, Jan. 2019, doi: 10.1155/2019/8292485.

[68] X. You, Y. Ma, and Z. Liu, "A three-stage algorithm on community detection in social networks," *Knowledge-Based Syst.*, vol. 187, p. 104822, Jan. 2020, doi: 10.1016/j.knosys.2019.06.030.

[69] N. Kasoro, S. Kasereka, E. Mayogha, H. T. Vinh, and J. Kinganga, "PercoMCV: A hybrid approach of community detection in social networks," *Procedia Comput. Sci.*, vol. 151, pp. 45–52, 2019, doi: 10.1016/j.procs.2019.04.010.

[70] N. Tahir, A. Hassan, M. Asif, and S. 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)*, Mar. 2019, pp. 160–165. doi: 10.1109/C-CODE.2019.8680980.

[71] L. Bai, J. Liang, H. Du, and Y. Guo, "A novel community detection algorithm based on simplification of complex networks," *Knowledge-Based Syst.*, vol. 143, pp. 58–64, 2018.

[72] J. Cao *et al.*, "A unified framework of semi-supervised community detection integrating network topology and node content," *Inf. Sci. (Ny).*, vol. 686, p. 121349, 2025.

[73] G. K. Orman, V. Labatut, and H. Cherifi, "Qualitative Comparison of Community Detection Algorithms," in *Digital Information and Communication Technology and Its Applications: International Conference, DICTAP 2011, Dijon, France, June 21-23, 2011, Proceedings, Part II*, 2011, pp. 265–279. doi: 10.1007/978-3-642-22027-2_23.

[74] M. R. Hamedani and S.-W. Kim, "JacSim: An accurate and efficient link-based similarity measure in graphs," *Inf. Sci. (Ny).*, vol. 414, pp. 203–224, Nov. 2017, doi: 10.1016/j.ins.2017.06.005.

[75] M. E. J. Newman, "Modularity and community structure in networks," *Proc. Natl. Acad. Sci.*, vol. 103, no. 23, pp. 8577–8582, Jun. 2006, doi: 10.1073/pnas.0601602103.

[76] G. Csardi and T. Nepusz, "The igraph software package for complex network research." pp. 1–9, 2006.

[77] A. Hagberg, P. J. Swart, and D. A. Schult, "Exploring network structure, dynamics, and function using NetworkX," 2008.

[78] K. E. Avrachenkov, A. Y. Kondratev, and V. V Mazalov, "Cooperative game theory approaches for network partitioning," in *Computing and Combinatorics: 23rd International Conference, COCOON 2017, Hong Kong, China, August 3-5, 2017, Proceedings 23*, 2017, pp. 591–602.

[79] D. Lusseau, K. Schneider, O. J. Boisseau, P. Haase, E. Slooten, and S. M. Dawson, "The bottlenose dolphin community of Doubtful Sound features a large proportion of long-lasting associations," *Behav. Ecol. Sociobiol.*, vol. 54, no. 4, pp. 396–405, Sep. 2003, doi: 10.1007/s00265-003-0651-y.

[80] J. Yang and J. Leskovec, "Defining and evaluating network communities based on ground-truth," *Knowl. Inf. Syst.*, vol. 42, no. 1, pp. 181–213, Jan. 2015, doi: 10.1007/s10115-013-0693-z.

[81] L. Jure, "SNAP Datasets: Stanford large network dataset collection," *Stanford Large Network Dataset Collection*, 2014. https://snap.stanford.edu/data/

[82] E. Eaton and R. Mansbach, "A spin-glass model for semi-supervised community detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2012, vol. 26, no. 1, pp. 900–906.

[83] N. R. Smith, P. N. Zivich, L. M. Frerichs, J. Moody, and A. E. Aiello, "A Guide for Choosing Community Detection Algorithms in Social Network Studies: The Question Alignment Approach," *Am. J. Prev. Med.*, vol. 59, no. 4, pp. 597–605, Oct. 2020, doi: 10.1016/j.amepre.2020.04.015.

[84] J. Zeng and H. Yu, "A Distributed Infomap Algorithm for Scalable and High-Quality Community Detection," in *Proceedings of the 47th International Conference on Parallel Processing*, Aug. 2018, pp. 1–11. doi: 10.1145/3225058.3225137.

[85] S. Fortunato, "Community detection in graphs," *Phys. Rep.*, vol. 486, no. 3–5, pp. 75–174, Feb. 2010, doi: 10.1016/j.physrep.2009.11.002.