An Iterated Heuristic Community Detection Algorithm for Social Networks Based on Centrality and Similarity Measures

Ali Chenaoui
LME Laboratory, Faculty of Technology, Hassiba Benbouali University, Chlef, Algeria | Department of Informatics, Faculty of Exact Sciences and Informatics, Hassiba Benbouali University, Chlef, Algeria
a.chenaoui@univ-chlef.dz (corresponding author)

Mohammed Amin Tahraoui
LIA Laboratory, Department of Informatics, Faculty of Exact Sciences and Informatics, Hassiba Benbouali University, Chlef, Algeria
m.tahraoui@univ-chlef.dz

Received: 4 March 2025 | Revised: 29 April 2025 | Accepted: 8 May 2025

 DOI: https://doi.org/10.48084/etasr.10791

ABSTRACT

Community detection is a crucial analytical approach to comprehend the structure and functionality of intricate networks, which are frequently conceptualized as graphical representations. This problem is extremely difficult and has not yet been satisfactorily solved. It is believed that neighbors may strongly surround a community's central or leader node, and that the centers of two communities may be far apart. Furthermore, it is postulated that the similarity between nodes within the same community is greater than that observed between nodes belonging to different communities. Thus, it is evident that local and global structural information is important in community detection. This paper presents an iterative heuristic algorithm for community detection in social networks using centrality and similarity measures called HCCS, which consists of four main steps: parameter initialization, leader nodes' selection, communities' formation, and post-processing. The main contributions are as follows: first, both local and global information about the network is considered, and then a heuristic formula for measuring the similarity between two nodes is redefined. The leader nodes are identified based on two criteria: degree and distance. During the community formation phase, the degree of similarity between the nodes is quantified using the heuristic formula. The nodes are then assigned to the same community once they reach maximum similarity. The postprocessing phase entails the integration of two communities when the modularity increment is positive, reaching its maximum when the two communities are merged. Experiments on real networks demonstrate the effectiveness of the proposed approach.

Keywords-community detection; social network; centrality; similarity measures; heuristic; modularity

I. INTRODUCTION

The study of social networks has become a prominent field in data science, driven by the exponential growth of social networking platforms. Although social interactions between individuals have been studied for decades [1], the advent of modern communication technologies, such as mobile phones and the Internet, has introduced novel modes of interaction. These interactions can now be tracked on a large scale, involving millions of individuals, offering substantial opportunities for social science research. A social network community is a group of individuals connected by a common set of friends or shared interests and activities.

Detecting communities in social networks means dividing social actors (nodes in the social graph) with certain social links (edges between nodes) into highly related groups (subgraphs of nodes with strong interactions between them and little interaction with the other subgraphs). The community detection problem can be defined as an optimization problem that seeks to identify a data analysis technique capable of uncovering the hidden structure of a large-scale networked dataset into disjoint and compact clusters. However, the number and size of the subgroups are unknown [2]. Furthermore, finding communities is a computationally challenging task, as this problem has been proven to be a non-deterministic polynomial-time hard (NP-hard) problem [3].

Community detection has evolved significantly, with methods ranging from traditional approaches to modern machine learning techniques. Early methods, such as modularity optimization, include the Louvain algorithm [4], Fast Greedy [5], and GA-OMA [6], which iteratively merge communities to maximize modularity but suffer from resolution limits [7]. Hierarchical methods, such as the Girvan-Newman algorithm [2], use divisive strategies to partition networks based on edge betweenness centrality, while agglomerative methods, such as Clauset-Newman-Moore [5], build communities by merging nodes. Spectral methods, including spectral clustering [3], leverage the eigenvalues and eigenvectors of the network's Laplacian matrix [8], offering strong performance but at high computational cost. Label propagation and local methods, such as the Label Propagation Algorithm (LPA) [9], WLPA [10], and Walktrap [11], rely on local node interactions and are efficient but can produce unstable results. Information-theoretic methods, such as Infomap [12], minimize the description length of a random walker's trajectory, making them effective for large, sparse networks. Recent advances in machine learning have introduced node embedding techniques, such as Node2Vec [13], Graph Neural Networks (GNNs) [14], and CLA-Chaos [15], which learn low-dimensional representations of nodes for clustering. These methods excel at capturing complex relationships but may lose structural information. Overlapping and dynamic community detection methods, such as the Clique Percolation Method (CPM) [16], Temporal Network Embedding [17], and CDHEA [18], address the limitations of static approaches by detecting overlapping communities or modeling evolving structures.

Existing community detection methods often struggle to balance local and global network information, leading to suboptimal community structures. Although efficient, modularity-based methods can suffer from resolution limits and may fail to detect small communities [7]. Although powerful, spectral methods can be computationally intensive for large networks. Node embedding, while capturing complex node relationships, can lose structural information crucial for community detection.

Community detection algorithms can be classified according to the type of network information they use. Local methods, such as LPA algorithms, rely on local node interactions to form communities [9]. Global methods, such as spectral clustering, consider the entire network structure to identify community patterns [3]. Hybrid methods combine both local and global information to balance granularity and scalability [19-20]. Many case studies highlight the effectiveness of using both centrality and similarity measures to detect communities in social networks [21-23]. These studies have demonstrated that both measures can provide valuable insights into social dynamics and organizational structure.

Most complex networks exhibit a free-scale structure, indicating that a small fraction of nodes possess a high degree, while the majority have limited connections, resulting in a low degree. Examination of intricate networks has demonstrated that within each community, a node assumes a central role [24]. This individual is tasked with sharing information and engaging potential new members of the community. Central or leader nodes are determined by applying centrality measures. Centrality serves as a metric to pinpoint significant nodes within a network and has been extensively studied within the domain of network analysis [25-26].

Similarity measures, in addition to centrality measures, are used to determine similarity or relatedness between nodes in a network. Similarity measures facilitate various tasks, including link prediction, recommendation systems, and community detection. When considering the similarity between nodes and central nodes, detecting communities within a social network is possible.

This paper proposes a novel iterated Heuristic Community Detection algorithm using Centrality and Similarity measures in social networks (HCCS). The study makes two primary contributions: first, it considers both local and global network information to improve community detection accuracy, and second, it proposes a novel heuristic formula for measuring the similarity between two nodes.

II. BACKGROUND

One way to represent a social network is as a graph that consists of vertices (representing individuals in the network) and edges (representing relationships between individuals). This section introduces fundamental concepts of graph theory and social network analysis, specifically centrality measures, similarity measures, and modularity.

All graphs considered here are undirected and unweighted, denoted by $G = (V, E)$, where V and E are the vertex and edge sets, respectively. The number of nodes and edges in the network is represented by $n = |V|$ and $m = |E|$, respectively.

The network structure is presented as an adjacency matrix $A = (a_{ij})_{n\times n}$, where a_{ij} equals 1 if there is an edge between nodes i and j; otherwise, a_{ij} equals 0. A vertex u is said to be adjacent (or neighbouring) to another vertex v if $u_v \in E$ (there is an edge between u and v). The neighbourhood of a vertex u, denoted $N(u)$, is the set of vertices adjacent to a vertex u and defined by $N(u) = \{v \mid uv \in E \text{ or } vu \in E\}$. The degree of a vertex u is the number of vertices adjacent to u and is given by $d(u) = |N(u)|$. The distance between two vertices, u and v, denoted by $dis(u, v)$, is defined as the shortest path/chain length between u and v. For any undefined terms, refer to [27].

In social network analysis, centrality measures play a crucial role in assessing the importance and influence of nodes within a network [28]. Different centrality measures, such as centrality degree [29], closeness centrality [30], betweenness centrality [31], and eigenvector centrality [32], provide different perspectives on node centrality. These centrality measures help identify key players, critical connectors, and influential nodes in social networks, supporting applications such as community discovery, information diffusion modelling, and targeted marketing strategies.

Degree centrality measures the number of edges connected to a node. Nodes with a higher degree are considered more central. Degree centrality is defined as:

$$Degree\ Centrality(v) = \frac{Number\ of\ edges\ connected\ to\ node\ v}{Total\ number\ of\ nodes - 1} \quad (1)$$

There are several commonly used approaches for calculating the centrality in a graph. Every method captures distinct features of node significance and can be implemented based on the specific characteristics and demands of the network being examined. In social network analysis, different similarity measures are crucial to assessing relatedness and similarity among nodes in a network. Standard similarity measures comprise Common neighbors, Jaccard similarity [33], and cosine Similarity [34]. These measures offer varied viewpoints on node similarity, including shared connections, overlapping neighborhoods, feature-based similarity, and structural proximity.

Common neighbors similarity measures the similarity between two nodes based on the number of neighbors they share and is calculated as follows:

$$Common\ Neighbors(u,v) = \frac{|N(u)\cap N(v)|}{|N(u)|+|N(v)|-|N(u)\cap N(v)|} \quad (2)$$

where $|N(u)|$ and $|N(v)|$ represent the number of neighbours of nodes u and v, respectively, and $|N(u) \cap N(v)|$ is the number of common neighbours of nodes u and v. Jaccard similarity compares the size of the common neighbours set to the size of the union of neighbours for two nodes, defined as:

$$Jaccard\ Similarity(u,v) = \frac{|N(u)\cap N(v)|}{|N(u)\cup N(v)|} \quad (3)$$

where $|N(u) \cap N(v)|$ is the number of common neighbours of nodes u and v, and $|N(u)\cup N(v)|$ is the number of distinct neighbours of nodes u and v. Cosine similarity measures the cosine of the angle between two vectors representing the neighborhoods of two nodes, defined as:

$$Cosine\ Similarity(u,v) = \frac{\sum_w a_{uw}\cdot a_{vw}}{\sqrt{\sum_w a_{uw}^2}\cdot\sqrt{\sum_w a_{vw}^2}} \quad (4)$$

where a_{uw} and a_{vw} are entries in the adjacency matrix, representing the presence (1) or absence (0) of edges between nodes u and w and between nodes v and w, respectively.

Similarity measures can also be employed to discover communities in a social network. Nodes exhibiting high similarity measurements are likelier to belong to the same community.

The evaluation of community detection algorithms employs a range of metrics, with modularity (Q) and Normalised Mutual Information (*NMI*) being two of the most widely utilized. These measures help assess the quality of detected communities compared to a ground truth or known structure.

In [35], network modularity was introduced to quantify the partitioning of a network into distinct, densely connected subgroups or communities. A high modularity score indicates that a network has a clear, highly modular community structure. The detailed expression of the modularity Q is:

$$Q = \frac{1}{2m}\sum_{ij}\left(a_{ij} - \frac{d(i)d(j)}{2m}\right)\delta(c_i,c_j) \quad (5)$$

where c_i and c_j represent the communities to which nodes i and j belong, respectively, and $\delta(c_i,c_j)$ is the Kronecker delta function, which equals 1 if nodes i and j are in the same community (module) and 0 otherwise.

NMI is a metric employed to assess the degree of similarity between a detected partition (U) and a ground truth partition (V) [36]. It is founded upon Mutual Information (MI), quantifying the extent of shared information between the two partitions. The mutual information $I(U,V)$ is calculated as follows:

$$I(U,V) = \sum_{u\in U}\sum_{v\in V} p(u,v)\log\left(\frac{p(u,v)}{p(u)p(v)}\right) \quad (6)$$

where $p(u,v)$ is the joint probability distribution of nodes in communities u and v, and $p(u)$ and $p(v)$ are the marginal distributions for communities U and V. NMI normalizes this mutual information by the square root of the product of the entropies of both partitions:

$$NMI(U,V) = \frac{I(U,V)}{\sqrt{H(U)H(V)}} \quad (7)$$

where $H(U)$ and $H(V)$ are the entropies of partitions U and V, respectively. *NMI* ranges from 0 to 1, with 1 indicating complete concordance.

III. PROPOSED APPROACH: HCCS ALGORITHM

This section describes the proposed Heuristic algorithm for the Community detection problem based on Centrality and Similarity (HCCS).

```
Algorithm 1: HCCS Framework
Input an undirected graph G = (V,E)
Output C*: Final assignment of nodes to
communities
k, d, MaxIter : Integers
θ: Float
Parameter-Initialisation(G, k, d, MaxIter,
θ)
C* ← ∅
for j = 1 to MaxIter do
  LN_i ← Leader-Nodes-Selection(G, k, d)
  PreC_i ← Communities-Formation(G, LN_i)
  C_i ← Post-Processing(G, PreC_i)
  if Q(C_i) > Q(C*) then
    C* ← C_i
  end if
end for
return C*
```

The algorithm consists of four steps: Parameter initialization, Leader nodes' selection, Communities' formation, and post-processing. First, it initializes some parameters and computes the similarity matrix, noted SM, based on the new similarity measure. After that, a new vertex ranking mechanism is proposed to select a set of nodes as the initial core of the communities. Progressively, the remaining suitable nodes are incorporated during the community formation phase to expand the initial communities. Finally, these communities gradually merge until the modularity Q does not increase. The last three modules are repeated until a stopping criterion is met, and the solution with the highest modularity is returned. To explain the proposed algorithm step-by-step, a small network with 14 vertices is presented in Figure 1.

A. Parameter Initialization

Algorithm 2 outlines the parameter initialization process. Key parameters include the degree threshold k and the distance threshold d, which are employed during the leader node selection phase. The threshold k is typically assigned a value close to the network's average degree to ensure leader nodes exhibit sufficient connectivity. At the same time, d is set to approximately one-fourth of the network diameter. This spatial constraint maintains separation between leaders, mitigating overlap and fostering diverse community seed placement across the network.

Subsequently, the similarity matrix, designated *SM*, is calculated. This matrix functions as a fundamental input during the community formation phase. In this context, an innovative similarity metric is introduced to evaluate the resemblance between two nodes, u and v, as delineated below:

$$NoD(u,v) = \frac{\theta \times |CN(u,v)|}{|N(u) \cup N(v)|} + \frac{(1-\theta) \times 2|E[CN(u,v)]|}{|CN(u,v)|(|CN(u,v)-1|)} \quad (8)$$

where $CN(u,v)$ represents the set of common neighbours of vertices u and v, and $E[CN(u,v)]$ represents the set of edges connecting common neighbours of u and v.

The similarity measure *NoD* evaluates node relationships through two complementary components. The first, a neighborhood overlap term, quantifies the proportion of shared neighbors relative to all neighbors of the two nodes, akin to a Jaccard index. This reflects the likelihood of nodes belonging to the same community, as shared neighbors often signal shared membership. The second component, a common neighbor density term, measures the density of connections among shared neighbors by comparing the number of actual edges with the total possible edges between them. This emphasizes structural cohesion, where tightly interconnected neighbors reinforce community strength. These components are balanced by a weighting parameter θ. In this work, θ=0.5 ensures equal emphasis on overlap and cohesion, aligning with the principle that robust communities require both shared membership and internal connectivity.

```
Algorithm 2: Parameter-Initialization (G, d, k, MaxIter, θ )
set parameters k, d, MaxIter, and θ
for each edge u, v ∈ E do
  calculate SM[u, v] according to (8)
end for
```

B. Leader Nodes Selection

This phase identifies leader nodes that act as initial seeds for communities, guided by two principles observed in real-world networks:

- Centrality-Driven Community Anchors: Communities often form around highly connected nodes (degree ≥ k), as these hubs naturally attract and bind members (e.g., influencers in social networks).
- Spatial Dispersion of Leaders: Central nodes of distinct communities tend to be non-adjacent. Enforcing a minimum distance d between leaders prevents overlapping communities and ensures diverse seed placement.

The algorithm begins by identifying candidate leaders with degrees ≥ k (lines 3-7). These candidates are iteratively pruned to ensure mutual distances ≥ d (lines 8-17). This process ensures leaders are both central and spatially distinct, forming diverse community seeds.

```
Algorithm 3: Leader-Nodes-Selection (G, d, k)
X ← Ø
S ← Ø
for each vertex u_i of V
  if d(u_i)≥ k then
    S ← S ∪ {u_i}
  end if
end for
while S ≠ Ø do
  Select a random vertex x from S
  X ← X ∪ {x}
  S ← S \ {x}
  for each vertex v_i of S
    if dis(x, v_i) < d  then
      S ← S \ {v_i}
    end if
  end for
end while
return X
```

In this example, as shown in Figure 1(a), the initial candidate leader nodes set with k = 3 is S = {1, 2, 3, 4, 5, 6, 9, 10, 12, 13}. When using the distance d = 2, certain nodes will be removed from S. Several solutions are possible. Figure 1(b) shows an example of the final detected leader nodes set X = {2, 5, 6, 13} with d = 2.

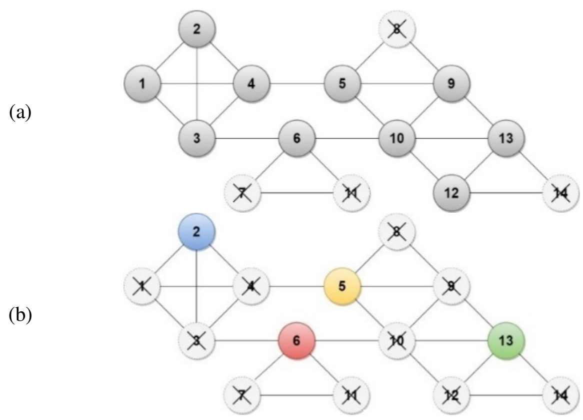


Fig. 1. The Leader node selection phase with k = 3 and d = 2: (a) S = {1,2,3,4,5,6,9,10,12,13}, X = { }; (b) S = { }, X = {2,5,6,13}.

C. Communities' Formation

This phase start from the leader nodes LN={x_1, x_2, …, x_k} returned by Algorithm 3 to create a set of intermediate communities C'. First, C' is initialized as follows: C' =

$\{C'_1 = \{x_1\}, C'_2 = \{x_2\}, \ldots, C'_k = \{x_k\}\}$. Then, each unassigned node v_j is assigned in the remaining nodes set RN to its corresponding intermediate community C'_m, based on the assumption that the nodes in the same community exhibit the highest similarities. Using the similarity matrix SM, the pair (x, y) of nodes with the highest similarity is grouped in the same community. Algorithm 4 is repeated from lines 5 to 14 until all remaining nodes RN are assigned to C'. Finally, the set of intermediate pre-communities C' is returned.

```
Algorithm 4: Communities-Formation (G, LN)
1. Let LN = {x1, x2, . . . , xk}
2. RN ← V \ LN
3. Let C' = {C'1 = {x1}, C'2 = {x2}, ..., C'k = {xk}}
4. while RN ≠ Ø do
5.   maxsim ← 0
6.   for each pair (ui, vj) ∈ C'×RN where
     ui, vj ∈ E do
7.     if SM[ui, vj] > maxsim then
8.       (x, y) ← (ui, vj)
9.       maxsim ← SM[ui, vj]
10.    end if
11.  end for
12.  m = communityof(x)
13.  C'm ← C'm ∪ {y}
14.  RN ← RN \ {y}
15.end while
16. return C'
```

Let the initial set of communities in this example (Figure 1) be $C' = \{C'_1, C'_2, C'_3, C'_4\}$, where C'_1={5}, C'_2={13}, C'_3={2}, C'_4={6}. So, the set of remaining nodes, denoted RN, is {1, 3, 4, 7, 8, 9, 10, 11, 12}.

The similarity matrix in Table I clearly shows that nodes 1 and 2 have the greatest similarity among the first four maximal similarities, so node 1 is assigned to the community of node 2, as shown in Figure 2, i.e., $C' = \{C'_1 = \{5\}, C'_2 = \{13\}, C'_3 = \{2,1\}, C'_4 = \{6\}\}$. This process is repeated until all nodes in RN are assigned to C'.

TABLE I. SIMILARITY MATRIX SM OF THE EXAMPLE GRAPH PRESENTED IN FIGURE 1 WITH $\theta = 0.5$

nodes	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1		1	0.83	0.83										
2	1		0.83	0.83										
3	0.83	0.83		0.75										
4	0.83	0.83	0.75											
5								0.17	0.25	0.08				
6							0.17				0.17			
7						0.17					0.50			
8					0.17				0.17					
9					0.25			0.17		0.20			0.10	
10					0.08				0.20			0.10	0.20	
11						0.17	0.50							
12										0.10			0.33	0.25
13									0.10	0.20		0.33		0.17
14												0.25	0.17	

It is important to note that this step may reveal situations where two pairs have the same highest similarity. For example, Node 10 exhibits the greatest similarity with Nodes 9 and 13, which are part of different communities. As a result, Node 10 is arbitrarily assigned to Node 9's community. Thus, the final pre-communities set is $C' = \{C'_1 = \{5, 9, 10, 8\}, C'_2 = \{13, 12, 14\}, C'_3 = \{2, 1, 3, 4\}, C'_4 = \{6, 7, 11\}\}$, as shown in Figure 2.

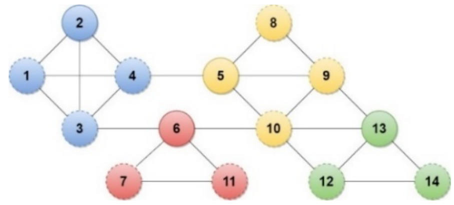


Fig. 2. The result of the communities' formation phase.

D. Post-Processing

As explained in Section II, community detection quality is commonly measured by modularity. Higher modularity Q values indicate higher-quality community structures. In the post-processing phase, the aim is to enhance the intermediate communities C' generated by Algorithm 4 by utilising modularity as an evaluation index for gauging network community structure quality.

Intermediate communities are incrementally merged until no further increases in modularity occur. First, the communities set C is initialized by C' and the initial modularity Mc=$Q(C)$. is calculated. Two communities C_i and C_j of C' are merged into one community $C_i \cup C_j$ when the modularity Mc' result of the merging becomes the largest and satisfies $Mc' > Mc$, where:

$$Mc' = \max_{\forall C_i, C_j \in C} Q\left(\left(C \backslash \{C_i, C_j\}\right) \cup \{C_i \cup C_j\}\right) \quad (7)$$

As shown in Algorithm 5, after updating C by merging the two communities' C_i and C_j into one community, the merging process is repeated until modularity is no longer increased. The optimal solution is formed, and the final set of communities C is returned.

```
Algorithm 5: Post-Processing (G, C')
Let C' = {C'1, C'2, ..., C'k}
nc ← k ; Merge ← true
while Merge do
  Mc ← Q(C')
  Merge ← false
  for each community C'i of C' do
    for each community C'j of C' do
      C←C'
      Let C = {C1, C2, ... , Cnc}
      C ← (C\{Ci, Cj}) ∪ {Ci ∪ Cj}
      Mc' ← Q(C)
      if Mc' > Mc then
      (x1, x2) ← (i, j);
        Mc ←Mc';
        Merge ← true;
```

```
        end if
      end for
    end for
  if Merge then
    C ← C'
    Merge C_{x1} and C_{x2} into one community
    C_{x1} ∪ C_{x2}
      nc--
      C' ← C
  end if
end while
return C
```

In this example, four communities $C = \{C_1, C_2, C_3, C_4\}$ have been detected. The maximum increase in modularity occurs when communities C_1 and C_2 are merged. Therefore, C_1 and C_2 are merged to create a single community $C_1 \cup C_2$. In the next step, there are no further mergers that increase modularity, and the following final community set is obtained: $C_1 = \{5, 8, 9, 10, 12, 13, 14\}$, $C_2 = \{1, 2, 3, 4\}$, and $C_3 = \{6, 7, 11\}$ as shown in Figure 3.

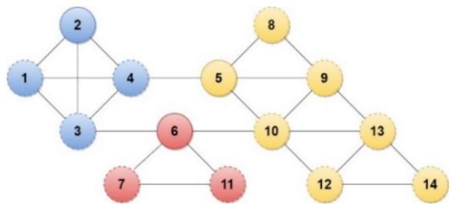


Fig. 3. Final community set.

IV. EXPERIMENTAL RESULTS

HCCS was compared against six established algorithms representing fundamental methodological paradigms, Louvain (modularity optimization), Fast Greedy (hierarchical agglomeration), Infomap (information-theoretic), LPA (label propagation), Eigenvector (spectral clustering), and Node2Vec (node embedding), to ensure a thorough evaluation. These baselines provide a comprehensive basis for assessing performance across multiple distinct methods. To validate the real-world applicability of HCCS, experiments were carried out on networks with and without ground-truth communities. These datasets span social, biological, and technological domains, enabling rigorous accuracy, scalability, and generalizability analysis.

A. *Real Networks with Ground Truth*

As shown in Table II, four ground truth networks with undirected and unweighted links were used to evaluate the proposed algorithm's efficiency and accuracy and compare it with the six previously mentioned algorithms. Table III shows the *NMI* and modularity values in the ground truth networks for HCCS and the other six algorithms. The proposed HCCS algorithm shows near-optimal performance regarding *NMI* and modularity on the four networks. The HCCS algorithm performs better than the other methods in the Karate and Football networks and maintains a competitive position in the other networks.

TABLE II. REAL NETWORKS WITH GROUND TRUTH.

Networks	$\|V\|$	$\|E\|$	**True communities**	**Description**
Dolphins	62	159	2	Dolphin social network [37]
Football	115	613	12	Network of American football games [2]
Karate	34	78	2	Zachary's social network of a karate club [38]
Polbooks	105	441	3	Network of books about US politics [39]

TABLE III. COMPARISON OF PERFORMANCE ON THE GROUND TRUTH NETWORKS.

		Dolphins	**Football**	**Karate**	**Polbooks**
Ground truth	*Q*	0.38	0.55	0.37	0.41
HCCS	*NMI*	0.89	0.9	0.71	0.55
	Q	0.46	0.6	0.42	0.52
Louvain [4]	*NMI*	0.48	0.88	0.59	0.51
	Q	0.52	0.6	0.42	0.52
Fast Greedy [5]	*NMI*	0.61	0.7	0.69	0.53
	Q	0.5	0.55	0.38	0.5
Infomap [12]	*NMI*	0.5	0.92	0.7	0.49
	Q	0.52	0.6	0.4	0.52
LPA [9]	*NMI*	0.69	0.92	0.7	0.57
	Q	0.5	0.6	0.4	0.5
Eigenvector [8]	*NMI*	0.45	0.52	0.68	0.71
	Q	0.49	0.47	0.39	0.49
Node2Vec [13]	*NMI*	0.85	0.19	0.73	0.57
	Q	0.38	0.38	0.34	0.5

To visually validate the effectiveness of HCCS, Figure 4 illustrates the communities detected by the algorithm on the Dolphins, Karate, and Polbooks ground truth networks, respectively, where each color represents a different detected community. These visualizations align with the quantitative results in Table III, confirming the ability of HCCS to detect meaningful communities that match real-world structures.

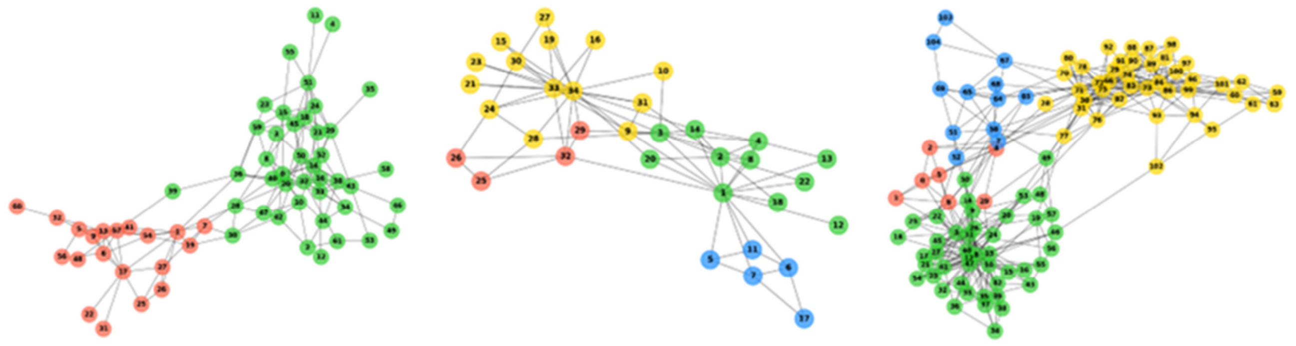
B. *Performance of the Proposed Similarity Measure*

To validate the superiority of the proposed similarity measure *NoD*(u,v), experiments were carried out by replacing *NoD* with two widely used similarity metrics, Jaccard similarity and cosine similarity, while retaining all other parameters and steps of the HCCS algorithm. The goal is to isolate the impact of the similarity measure on community detection accuracy. The results are summarised in Table IV, which compares the *NMI* and *Q* values across similarity measures on ground-truth networks.

TABLE IV. COMPARISON OF SIMILARITY MEASURES

		Dolphins	**Football**	**Karate**	**Polbooks**
HCCS-NoD	*NMI*	0.89	0.90	0.71	0.55
	Q	0.46	0.60	0.42	0.52
HCCS-Jaccard	*NMI*	0.89	0.90	0.71	0.55
	Q	0.38	0.60	0.40	0.52
HCCS-Cosine	*NMI*	0.89	0.90	0.80	0.55
	Q	0.38	0.60	0.37	0.52

The results demonstrate that while the HCCS-NoD achieves comparable *NMI* to Jaccard/cosine in networks such as Dolphins and Football, its integration of common neighbor density ensures superior modularity (e.g., 0.46 vs. 0.38 in Dolphins), reflecting stronger structural cohesion. In Karate, HCCS-NoD balances *NMI* (0.71) and *Q* (0.42), avoiding over-merging seen with Cosine (*NMI*=0.82 but lower *Q*=0.37). For simpler networks, such as Polbooks, all metrics perform equally, underscoring *NoD*'s value in complex, ambiguous structures where both neighborhoods overlap and internal connectivity defines communities. This hybrid approach is critical for applications requiring interpretable, tightly knit groups, such as fraud detection or social influence analysis.



(a) (b) (c)

Fig. 4. The visualizations of communities detected by HCCS: (a) Dolphins, (b) Karate, (c) Polbooks

C. *Real Networks without Ground Truth*

Table V shows five different-scale networks without ground truths. These were analyzed using HCCS and the six other algorithms. Modularity was used to assess the quality of the community detection results for these networks.

TABLE V. REAL NETWORKS WITHOUT GROUND TRUTHS

Networks	$\|V\|$	$\|E\|$	Description
Jazz_collab	198	2742	Network of collaborations among jazz musicians [40]
Lesmis	77	254	Network of characters scenes in Victor Hugo's novel "Les Misérables" [41]
Polblogs	1222	16714	US Political Blogs Network [39]
Power	4941	6594	The Western States Power Grid of the United States [42]
Uni_email	1133	10903	Network of e-mail interchanges [43]

Table VI shows the modularity values in the networks without ground truths. The HCCS algorithm performed well on all five networks without ground truths. The achievement of comparable results with Louvain, which has been validated by empirical evidence [25], serves to substantiate the efficacy and effectiveness of the proposed HCCS algorithm.

TABLE VI. PERFORMANCE COMPARISON ON NETWORKS WITHOUT GROUND TRUTHS

	Jazz_collab	Lesmis	Polblogs	Power	Uni_email
HCCS	0.44	0.54	0.42	0.93	0.55
Louvain	0.44	0.56	0.42	0.93	0.54
Fast Greedy	0.44	0.5	0.43	0.93	0.51
Infomap	0.28	0.55	0.42	0.82	0.52
LPA	0.39	0.53	0.42	0.83	0.49
Eigenvector	0.28	0.55	0.43	0.81	0.28
Node2Vec	0.29	0.31	0.42	0.4	0.33

V. CONCLUSION

The HCCS algorithm introduces a hybrid approach to community detection by unifying local centrality and global similarity measures, addressing the persistent challenge of balancing granular node interactions with the overarching network structure. By leveraging degree centrality to identify influential leader nodes and a novel similarity heuristic NoD to quantify cohesion among shared neighbors (which outperforms traditional metrics such as Jaccard and cosine), HCCS outperforms modularity-centric methods such as Louvain in detecting overlapping communities while avoiding the instability of purely local approaches such as label propagation. This dual focus enables robust identification of cohesive groups, validated through experiments on real-world networks such as Football and Dolphins, where HCCS consistently achieved high modularity and alignment with ground-truth communities. The practical value of HCCS spans applications like social influence analysis and fraud detection, where its hybrid design precisely identifies cohesive communities. However, its computational complexity limits scalability in large networks such as Polblogs, underscoring a trade-off between accuracy and efficiency. Despite this, its deterministic approach ensures reliable results compared to unstable methods such as label propagation.

Future efforts will prioritize multiobjective optimization to balance modularity, resolution, and speed along with scalability enhancements through parallel computing and dynamic adaptations for real-time networks. Expanded evaluations on synthetic benchmarks (e.g., LFR) and domain-specific datasets will further validate adaptability. In addition, comparisons with emerging state-of-the-art algorithms, such as GNNs, can benchmark HCCS against evolving computational paradigms. These advances will solidify its role as a versatile tool for analyzing complex systems across disciplines.

REFERENCES

[1] J. Scott, *Social Network Analysis: A Handbook*, 2nd ed. SAGE Publications Ltd, 2000.

[2] M. Girvan and M. E. J. Newman, "Community structure in social and biological networks," *Proceedings of the National Academy of Sciences*, vol. 99, no. 12, pp. 7821–7826, Jun. 2002, https://doi.org/10.1073/pnas.122653799.

[3] M. E. J. Newman, "Modularity and community structure in networks," *Proceedings of the National Academy of Sciences*, vol. 103, no. 23, pp. 8577–8582, Jun. 2006, https://doi.org/10.1073/pnas.0601602103.

[4] V. D. Blondel, J. L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2008, no. 10, Jul. 2008, Art. no. P10008, https://doi.org/10.1088/1742-5468/2008/10/P10008.

[5] A. Clauset, M. E. J. Newman, and C. Moore, "Finding community structure in very large networks," *Physical Review E*, vol. 70, no. 6, Dec. 2004, Art. no. 066111, https://doi.org/10.1103/PhysRevE.70.066111.

[6] B. Zarei and M. R. Meybodi, "Detecting community structure in complex networks using genetic algorithm based on object migrating automata," *Computational Intelligence*, vol. 36, no. 2, pp. 824–860, 2020, https://doi.org/10.1111/coin.12273.

[7] S. Fortunato and M. Barthélemy, "Resolution limit in community detection," *Proceedings of the National Academy of Sciences*, vol. 104, no. 1, pp. 36–41, Jan. 2007, https://doi.org/10.1073/pnas.0605965104.

[8] M. E. J. Newman, "Finding community structure in networks using the eigenvectors of matrices," *Physical Review E*, vol. 74, no. 3, Sep. 2006, Art. no. 036104, https://doi.org/10.1103/PhysRevE.74.036104.

[9] U. N. Raghavan, R. Albert, and S. Kumara, "Near linear time algorithm to detect community structures in large-scale networks," *Physical Review E*, vol. 76, no. 3, Sep. 2007, Art. no. 036106, https://doi.org/10.1103/PhysRevE.76.036106.

[10] B. Zarei, M. R. Meybodi, and B. Masoumi, "Detecting community structure in signed and unsigned social networks by using weighted label propagation," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 30, no. 10, Oct. 2020, Art. no. 103118, https://doi.org/10.1063/1.5144139.

[11] P. Pons and M. Latapy, "Computing Communities in Large Networks Using Random Walks," in *Computer and Information Sciences - ISCIS 2005*, 2005, pp. 284–293, https://doi.org/10.1007/11569596_31.

[12] M. Rosvall and C. T. Bergstrom, "Maps of random walks on complex networks reveal community structure," *Proceedings of the National Academy of Sciences*, vol. 105, no. 4, pp. 1118–1123, Jan. 2008, https://doi.org/10.1073/pnas.0706851105.

[13] A. Grover and J. Leskovec, "node2vec: Scalable Feature Learning for Networks," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, May 2016, pp. 855–864, https://doi.org/10.1145/2939672.2939754.

[14] T. N. Kipf and M. Welling, "Semi-Supervised Classification with Graph Convolutional Networks." arXiv, Feb. 22, 2017, https://doi.org/10.48550/arXiv.1609.02907.

[15] B. Zarei, M. R. Meybodi, and B. Masoumi, "A New Evolutionary Model Based on Cellular Learning Automata and Chaos Theory," *New Generation Computing*, vol. 40, no. 1, pp. 285–310, Apr. 2022, https://doi.org/10.1007/s00354-022-00159-1.

[16] G. Palla, I. Derényi, I. Farkas, and T. Vicsek, "Uncovering the overlapping community structure of complex networks in nature and society," *Nature*, vol. 435, no. 7043, pp. 814–818, Jun. 2005, https://doi.org/10.1038/nature03607.

[17] P. Goyal, S. R. Chhetri, and A. Canedo, "dyngraph2vec: Capturing network dynamics using dynamic graph representation learning," *Knowledge-Based Systems*, vol. 187, Jan. 2020, Art. no. 104816, https://doi.org/10.1016/j.knosys.2019.06.024.

[18] B. Zarei, B. Arasteh, M. Asadi, V. Majidnezhad, S. T. Afshord, and A. Bouyer, "Multiplex Community Detection in Social Networks Using a Chaos-Based Hybrid Evolutionary Approach," *Complexity*, vol. 2024, no. 1, 2024, Art. no. 1016086, https://doi.org/10.1155/2024/1016086.

[19] B. Zarei, M. R. Meybodi, and B. Masoumi, "Chaotic memetic algorithm and its application for detecting community structure in complex networks," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 30, no. 1, Jan. 2020, Art. no. 013125, https://doi.org/10.1063/1.5120094.

[20] G. R. Abdi, A. H. Refahi Sheikhani, S. Kordrostami, B. Zarei, and M. Falah Rad, "Identifying communities in complex networks using learning-based genetic algorithm," *Ain Shams Engineering Journal*, vol. 15, no. 12, Dec. 2024, Art. no. 103031, https://doi.org/10.1016/j.asej.2024.103031.

[21] S. Su, J. Guan, B. Chen, and X. Huang, "Nonnegative Matrix Factorization Based on Node Centrality for Community Detection," *ACM Transactions on Knowledge Discovery Data*, vol. 17, no. 6, Oct. 2023, Art. no. 84, https://doi.org/10.1145/3578520.

[22] C. Feng, J. Ye, J. Hu, and H. L. Yuan, "Community Detection by Node Betweenness and Similarity in Complex Network," *Complexity*, vol. 2021, no. 1, 2021, Art. no. 9986895, https://doi.org/10.1155/2021/9986895.

[23] X. You, Y. Ma, and Z. Liu, "A three-stage algorithm on community detection in social networks," *Knowledge-Based Systems*, vol. 187, Jan. 2020, Art. no. 104822, https://doi.org/10.1016/j.knosys.2019.06.030.

[24] J. Lei, W. X. Juan, and Z. Yong, "The independence of the centrality for community detection," *International Journal of Modern Physics C*, vol. 29, no. 07, Jul. 2018, Art. no. 1850060, https://doi.org/10.1142/S0129183118500602.

[25] S. Ahajjam and H. Badir, "Community Detection in Social Networks," in *Principles of Social Networking: The New Horizon and Emerging Challenges*, A. Biswas, R. Patgiri, and B. Biswas, Eds. Springer, 2022, pp. 91–107.

[26] J. Sheykhzadeh, B. Zarei, and F. Soleimanian Gharehchopogh, "Community Detection in Social Networks Using a Local Approach Based on Node Ranking," *IEEE Access*, vol. 12, pp. 92892–92905, 2024, https://doi.org/10.1109/ACCESS.2024.3420109.

[27] J. A. Bondy and U. S. R. Murty, *Graph Theory with Applications*. American Elsevier Publishing Company, 1976.

[28] K. Mouley and M. A. Tahraoui, "Locating the Source of Information in Social Networks using Critical Nodes," *Engineering, Technology & Applied Science Research*, vol. 15, no. 1, pp. 19136–19142, Feb. 2025, https://doi.org/10.48084/etasr.9283.

[29] J. Scott, Social Networks: Critical Concepts in Sociology. Taylor & Francis, 2002.

[30] A. Bavelas, "Communication patterns in task-oriented groups," *Journal of the Acoustical Society of America*, vol. 22, pp. 725–730, 1950, https://doi.org/10.1121/1.1906679.

[31] L. C. Freeman, "A Set of Measures of Centrality Based on Betweenness," *Sociometry*, vol. 40, no. 1, pp. 35–41, 1977, https://doi.org/10.2307/3033543.

[32] P. Bonacich, "Power and Centrality: A Family of Measures," *American Journal of Sociology*, vol. 92, no. 5, pp. 1170–1182, Mar. 1987, https://doi.org/10.1086/228631.

[33] P. Jaccard, "Étude comparative de la distribution florale dans une portion des Alpes et du Jura," *Bulletin de la Societe Vaudoise des Sciences Naturelles*, vol. 37, no. 142, pp. 547–579, 1901, https://doi.org/10.5169/SEALS-266450.

[34] G. Salton, Introduction to Modern Information Retrieval. McGraw-Hill, 1983.

[35] M. E. J. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Physical Review E*, vol. 69, no. 2, Feb. 2004, Art. no. 026113, https://doi.org/10.1103/PhysRevE.69.026113.

[36] L. Danon, A. Díaz-Guilera, J. Duch, and A. Arenas, "Comparing community structure identification," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2005, no. 09, Jun. 2005, Art. no. P09008, https://doi.org/10.1088/1742-5468/2005/09/P09008.

[37] D. Lusseau, "The emergent properties of a dolphin social network," *Proceedings of the Royal Society of London. Series B: Biological Sciences*, vol. 270, pp. S186–S188, Nov. 2003, https://doi.org/10.1098/rsbl.2003.0057.

[38] W. W. Zachary, "An Information Flow Model for Conflict and Fission in Small Groups," *Journal of Anthropological Research*, vol. 33, no. 4, pp. 452–473, Dec. 1977, https://doi.org/10.1086/jar.33.4.3629752.

[39] L. A. Adamic and N. Glance, "The political blogosphere and the 2004 U.S. election: divided they blog," in *Proceedings of the 3rd international workshop on Link discovery*, May 2005, pp. 36–43, https://doi.org/10.1145/1134271.1134277.

[40] P. M. Gleiser and L. Danon, "Community structure in jazz," *Advances in Complex Systems*, vol. 06, no. 04, pp. 565–573, Dec. 2003, https://doi.org/10.1142/S0219525903001067.

[41] D. E. Knuth, *The Stanford GraphBase: a platform for combinatorial computing*, First paperback printing. ACM Press, 2009.

[42] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world' networks," *Nature*, vol. 393, no. 6684, pp. 440–442, Jun. 1998, https://doi.org/10.1038/30918.

[43] R. Guimerà, L. Danon, A. Díaz-Guilera, F. Giralt, and A. Arenas, "Self-similar community structure in a network of human interactions," *Physical Review E*, vol. 68, no. 6, Dec. 2003, Art. no. 065103, https://doi.org/10.1103/PhysRevE.68.065103.