Abstract:

Social Networks possess a unique feature known as clustering or community structures which have no underlying universally accepted definition. Many authors have proposed many algorithms to detect these fascinating structures which have various ranges of applications from E-Commerce to Criminology. Many algorithms proposed for Community Detection are complex and require high computational power but many have low accuracy and are non-deterministic. This paper proposes advancement to the newly proposed local algorithm, LCD-SN. The improved LCD-SN Algorithm gives an iterative approach to calculate the IMP index which is proposed to determine the importance of network nodes using their local characteristics (first- and second-degree neighbors) thus enhancing the accuracy and modularity. Experiments show that improved LCD-SN is effective in identifying communities in social networks with improved accuracy and modularity.

INDEX TERMS: Community structure detection, NMI Score, modularity, nodes ranking, social network analysis

Introduction:

In real world everything can be structured down to links and nodes which establish a graph. In mathematical terms a graph G can be defined as G = (V, E), where V = {v1, v2, ..., vn} represents a set of vertices (nodes) with |V| = n, and E ⊆ V × V denotes a set of edges with |E| = m [1]. An interesting feature that real world networks present is the clustering property under which the graph topology is organized into modules commonly called communities. The essence here is that nodes of the same community are highly similar while on the contrary, nodes across communities present low similarity [2]. Since there is no universally accepted definition of community, it is a complicated task to distinguished community detection algorithms as each of them use a distinguished community detection algorithms as each of them use a different approach, resulting in different outcomes.

Community detection systems are broadly categorized into global and local algorithms, each with distinct characteristics. Global algorithms, such as Graph Partitioning [3], the Girvan-Newman method [4], and the Louvain Method [5], typically offer high accuracy by focusing on optimizing certain metrics, such as minimizing the number of inter-community edges or maximizing modularity. Graph Partitioning divides the network into clusters by minimizing edges between different communities [3], while the Girvan-Newman method identifies communities by iteratively removing edges with high betweenness centrality [4]. The Louvain Method, on the other hand, is known for its efficiency in optimizing modularity through an iterative process of node merging [5]. However, despite their high accuracy, global algorithms are not scalable for large networks due to their computational complexity and high time requirements, making them less practical for handling vast datasets.

On the other hand, local algorithms like Label Propagation (LPA) [5], WLPA [6], and RTLCD [7] prioritize scalability and efficiency, making them suitable for large networks. LPA operates by allowing nodes to adopt labels from their neighbors, propagating community labels throughout the network. While this method is fast, it can be uncertain due to its non-deterministic nature [5]. WLPA improves upon LPA by incorporating communication intensity between nodes, leading to more meaningful community assignments [6]. RTLCD addresses the seed-dependency and core validity issues present in other local methods [7]. Despite their scalability advantages, local algorithms often suffer from instability, as they depend heavily on predefined parameters and can yield lower accuracy compared to global approaches. This makes local algorithms suitable for large-scale applications but less reliable when high precision is critical.

The identification of communities in social networks can represent a critical task in revealing latent patterns, identifying key nodes, and analyzing the topology of social interactions. Local Community Detection in Social Networks (LCD-SN) [8] is a recent approach that focuses on identifying cohesive subgroups in a graph based on node relevance and relational behavior within neighborhoods. However, the classic LCD-SN algorithm faces issues related to scalability and precision when applied to complex networks characterized by overlapping communities or heterogeneous levels of node influence.  
  
To these challenges, we introduce an improved LCD-SN algorithm that enhances the evaluation of node importance, advances techniques of merging communities, and incorporates a stronger method of modularity optimization. This advanced methodology improves computationally efficient calculation for the significance of nodes by considering both first-degree and second-degree neighbors as one group with the least computational burdens but preserving accuracy. Besides, a more refined community merging criterion by the ratio of the intra-community edges and the inter-community edges presents that weak communities tend to merge to stronger ones, thus cultivating stronger community structures.  
  
We demonstrate that this new LCD-SN algorithm outperforms its predecessor in most of the cases, with higher modularity and accordingly better ability to detect communities, from the examination of a few empirical networks, including Karate, Dolphins, Football, PolBooks, and NetScience. This makes the algorithm more robust in handling a larger class of social network structures, bringing the analysis of large complex networks within scalable and effective means.

RELATED WORKS:

The study of community detection in social networks started off with the earliest approaches, which consisted of genetic algorithms and label propagation techniques aimed at uncovering hidden structures by mimicking evolutionary principles and network diffusion. However, as the networks grew larger and more complex, these methods became limited to issues of overlapped communities, computational complexity, and difficulties in handling large amounts of data. Scholars soon realized that reliance on network architecture alone was not sufficient and hence began to include node attributes-such as content similarity-in their approaches. This shift towards local community detection and distributed structures marked a major advancement that opened avenues to even more scalable and applicable solutions. However, despite such developments, the need for algorithms that could handle overlap nodes and fragmentation continued to remain strong, thus further sustaining the research efforts.

Addressing these issues, a novel remedy in the form of an algorithm called Local Community Detection in Social Networks (LCD-SN) [8] has been developed, focusing on the local characteristics of nodes and their neighbors. Unlike typical approaches that study the entire networks, LCD-SN employs a multi-phase approach that starts with the calculation of an Importance Metric (IMP). This measure assesses a node's influence based on the immediate neighbors, both of first and second degree, thereby guaranteeing that nodes with the most influence are given priority in the initial stages of community building. Comprising this analysis locally, the algorithm lowers the computational costs and facilitates scalability, making it highly suitable for large networks. An important contribution of LCD-SN is its ability to detect overlapping communities and merge weaker clusters into more cohesive ones.

The algorithm places overlapping nodes in communities by matching to the highest similarity with neighboring nodes, ensuring that they get a good placement and that the overall configuration improves. In the final step, it computes both within and between-community density of connections to capture smaller or weaker groups. If a community has more external connections than internal ones, it merges with a strong adjacent community, thereby strengthening cohesion and reducing fragmentation. LCD-SN presents minimal time complexity and high effectiveness both for empirical and synthetic datasets, thus making it a significant advancement for community discovery, introducing a new benchmark in the field of precision, scalability, and flexibility.

THE PROPOSED ALGORITHM:

The Proposed Algorithm Improved LCD-SN

This paper introduces an advanced iteration of LCD-SN, an algorithm for local community detection in social networks, designed to enhance computational efficiency and to better identify overlapping communities and clusters with higher cohesion. Improvements made for this version include optimization of the node significance calculation processes, similarity assessment procedures, and merging community processes while considering low time complexity and scalability. The rest of the subsections detail the iterations of the advanced algorithm.

A. Nodes Importance Calculation

The core notion of node importance is straightforward, which makes it easy to determine which nodes are important in the communities. More importantly, in the original LCD-SN algorithm, the importance score calculated for each node was a weighted sum of its first and second level of neighbors determined by coefficients α and β. Such two-level computation has paid the price of increased computational costs. In this version, the influence of all the neighbors are considered together in a single importance computation without redundancy, thus scalable.

Formula for enhanced node importance:

Formula:

Where:

• N(i): Set of all neighbors of node i

In this context, N(i) denotes the complete set of neighbors associated with node i, while α represents the standardized weight. This enhancement reduces the necessity for repetitive weight modifications and streamlines the computational process, thereby enabling the algorithm to manage more extensive networks with increased efficacy. Nodes that exhibit elevated importance scores are emphasized as central nodes within the community formation procedure.

B. Similarity Calculation and Overlapping Node Assignment

Handling overlapping nodes, which belong to multiple communities, is crucial to the strength of community detection. It is through this improved algorithm that utilizes a similarity index that determines the most appropriate community each overlapping node should join. The similarity between a node i and a candidate community is defined as:

Formula:

Where:

• N(i): Set of neighbors of node i

• C: Community

• |N(i) ∩ C|: Number of common neighbors between node i and community C

• |N(i)|: Total number of neighbors of node i

In this sense, N(i) ∩ C is the number of neighbors of node i that belong also to the community C. Overlapping nodes are assigned to the community with which they share most similarities. This method guarantees consistent assignments of nodes, reduces ambiguity, and enhances the cohesiveness of the identified communities by correctly assigning nodes to their most relevant groups.

C. Community Merging Strategy

One of the key challenges in the domain of community detection is dealing with weak or small communities that might destroy the global structure. The new algorithm incorporates a two-stage merging process to assimilate those communities into stronger entities:

1. Community merge: Communities of fewer than a defined number of nodes, say three, are merged together with adjacent communities that have more nodes based on the similarity measure used above.

2. Poor integration of communities: The algorithm evaluates the strength of every community by cross-examining the internal connections E\_in with the number of external connections E\_out. Merging criteria are the following:

Weak Community Merging Condition:

Where:

•: Number of edges within community C

•: Number of edges between community C and the rest of the graph

• V: Set of all nodes in the graph

• I(u,v): Indicator function that is 1 if there is an edge between nodes u and v, and 0 otherwise

• mc: Merging criterion threshold

Such a community is then amalgamated with a neighboring community to form a robust cluster. This parameter, mc, known as the merging threshold, can then be fine-tuned to absorb communities that are badly fragmented or poorly connected.

D. Modularity Computation

To evaluate the effectiveness of the identified community structure, the algorithm computes modularity (Q), a metric that quantifies the degree to which the network is partitioned into communities characterized by strong intra-community links and weak inter-community connections. The expression for modularity is as follows:

Formula:

Where:

• Q: Modularity

•: Adjacency matrix element (1 if there is an edge between nodes i and j, otherwise 0)

•: Degrees of nodes i and j

• m: Total number of edges

• δ(c\_i, c\_j): Delta function that is 1 if nodes i and j are in the same community, and 0 otherwise

A\_ij represents the adjacency matrix element, k\_i and k\_j are the node degrees of nodes i and j, m is the number of edges in total, and δ(c\_i, c\_j) denotes a delta function that returns 1 if both nodes belong to the same community and returns 0 otherwise. Therefore, higher modularity means better division of the community.

E. Verification and Management of Absent Nodes This leads the algorithm into a verification stage that ensures nodes in the network are assigned into a community. The unassigned set of nodes is calculated as follows:

Formula:

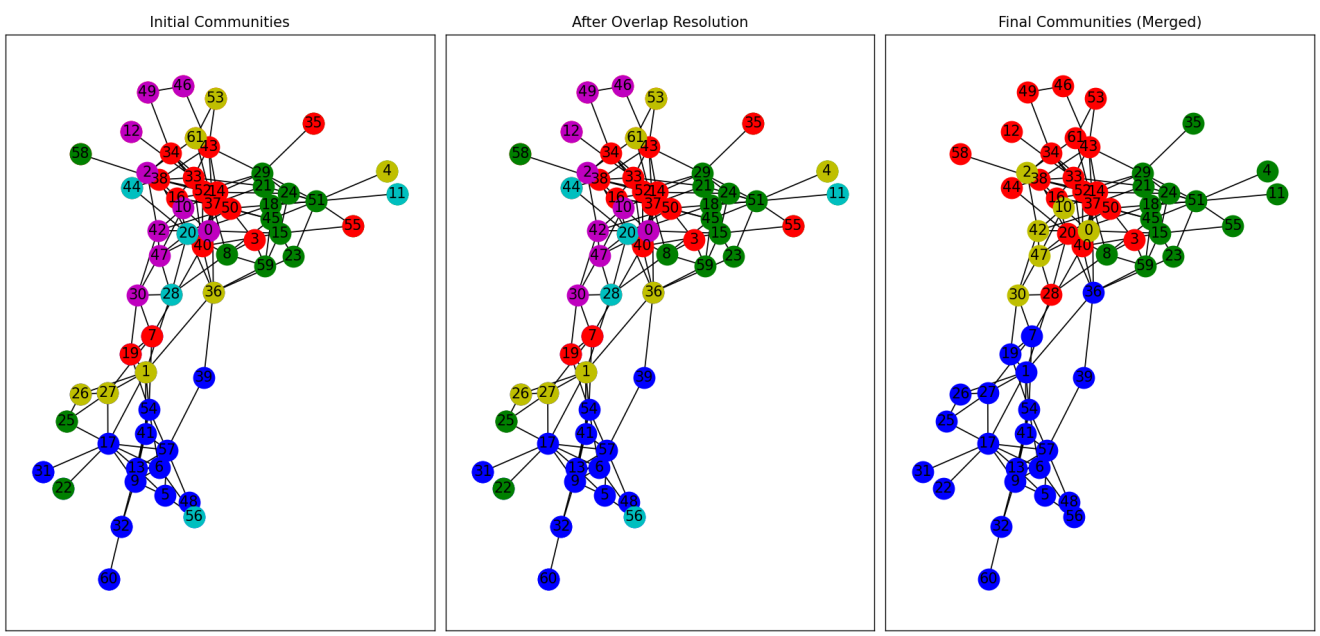
Where:

• V: Set of all nodes in the graph

• : Union of all communities

For any missing nodes, the algorithm attributes them to the community with which they have the highest similarity, so the coverage of the network will be complete.

Phases of the Community Formation Graph



F. Computational Complexity Analysis:

The improved LCD-SN algorithm demonstrates remarkable computational efficiency, making it particularly suitable for large-scale social networks. In Phase 1, which focuses on initial community formation, the importance scores of all nodes are initialized and iteratively updated. This involves traversing the neighbors of every node, resulting in a time complexity of O(γ ⋅ n ⋅ k), where n is the number of nodes, k is the average degree, and γ is the number of iterations. Although sorting nodes by importance incurs an additional O(n log n) complexity, this phase is primarily dominated by O(γ ⋅ n ⋅ k). In Phase 2, which addresses overlapping node assignment, the algorithm calculates similarity scores between overlapping nodes and their respective communities, requiring O(v ⋅ C ⋅ k) for each overlapping node v, where C is the average number of communities containing the node. Phase 3 focuses on community integration, ensuring effective merging of small and weak communities, with complexities of O(L ⋅ k) for small communities (where L is the number of such communities) and O(n ⋅ k) for weak communities based on edge densities. Combining the complexities of all phases, the overall time complexity of the algorithm is O(n ⋅ k), ensuring linear scalability with network size. The space complexity is O(n + m), where m is the number of edges, accounting for the storage of graph structures, importance scores, and community assignments. Overall, the improved LCD-SN algithm efficiently handles overlapping nodes and fragmented communities while maintaining low computational overhead, providing a highly scalable and practical solution for real-world social networks.

G. Conclusion:

This algorithm provides a more effective and scalable solution to community detection through simplification of node importance assessments, valid overlap allocation of nodes, and coalescing smaller or weaker communities. It is suitable for the testing of real world as well as synthetic social networks since it possesses very minimal time complexity with a better modularity value. Being stable in nature along with being accurate and scalable, the improved LCD-SN is a significant tool for large scale communities' detection. The future research deals with the adaptation of this algorithm in multi-layer, weighted and signed networks environments for expanding its flexibility and effectiveness.

ALGORITHM FOR IMPROVED LCD-SN

**Algorithm LCD-SN(G, α, β, γ, mc)**

Input:

G: Social Network Graph

α: Influence of first-degree neighbors

β: Influence of second-degree neighbors

γ: Maximum number of iterations for calculating node importance

mc: Merging criterion threshold

Output: Community structure of G

// Phase 1: Formation of Initial Communities

Initialize importance for each node to 1.0

Repeat γ times:

For each node in G:

neighbors ← Get neighbors of node

imp\_score← 0

For each neighbor in neighbors:

neighbor\_imp← importance[neighbor]

neighbor\_degree← Degree of neighbor

imp\_score←imp\_score + α \* (neighbor\_imp / neighbor\_degree)

Update importance[node] to imp\_score

Initialize empty list of communities

Initialize empty set of visited nodes

For each node in G sorted by importance (descending):

If node not in visited nodes:

Create new community with node and its neighbors

Add new community to list of communities

Add all nodes in new community to visited nodes

// Phase 2: Determining Status of Overlapping Nodes

For each overlapping node v in G:

Initialize max\_similarity to 0

Initialize best\_community to null

For each community C that contains v:

Calculate similarity between v and C

If similarity >max\_similarity:

max\_similarity← similarity

best\_community← C

Assign v to best\_community

// Phase 3: Integration of Communities

// Merge small communities with large ones

Initialize list L of small communities (size < 3)

While L is not empty:

Select a community C from L

Find the most similar neighboring community to C

Merge C into the most similar neighboring community

Remove C from L

// Merge weak communities with strong ones

Initialize final\_communities as empty list

For each community C\_i in list of communities:

Calculate E\_in and E\_out for C\_i

If E\_in ≤ mc \* E\_out:

Find the most similar neighboring community to C\_i

Merge nodes of C\_i into the most similar neighboring community

Else:

Add C\_i to final\_communities

Output final\_communities

Comparison of modularity with LCD-SN AND IMPROVED LCD-SN

In this section, we compare the modularity values obtained by the old LCD-SN algorithm with those produced by the improved LCD-SN algorithm on several real-world benchmark networks. The quality of community detection in these networks is measured by the Q criterion, which stands for modularity. For these algorithms, more significant values of modularity indicate that the communities are more well-defined. A comparison of the old and improved LCD-SN algorithms on a large set of real-world benchmark networks shows substantial improvements in community detection for most of the networks. In the Karate, Dolphins, PolBooks, NetScience, CA-HepTh, LesMis, GD01\_b, and CAG\_mat72 networks, LCD-SN reported higher values of modularity indicating a considerably better defined community structure. For instance, in Dolphins network modularity rises from 0.4486 to 0.4980, and in NetScience network it rose significantly from 0.8162 up to 0.8794. These results reflect the improvement in the performance of the enhanced algorithm. However, for example, in the case of the CA-HepPh network, this improved algorithm with the value of modularity slightly decreased from 0.5028 to 0.4419; this means that even though the improved algorithm generally gives a better version of the old one, there are some values of characteristics specific for the network for which an improved algorithm may not work well. In general, the LCD-SN improved algorithm is much stronger than the earlier versions at providing a more reliable solution for the detection of community structures in real-world networks.

|  |  |  |
| --- | --- | --- |
| **NETWORKS** | **OLD LCD-SN** | **IMPROVED LCD-SN** |
| KARATE | 0.36146614069691 | 0.37146614069691 |
| DOLPHINS | 0.448577983465843 | 0.4979826747359677 |
| FOOTBALL | 0.01687446136167773 | 0.013576558460212527 |
| POLBOOKS | 0.3747743995557406 | 0.4816357381955049 |
| NETSCIENCE | 0.8162313222889432 | 0.8794230695666407 |
| CA-HepPh | 0.5028165052617082 | 0.44194871034692684 |
| CA-HepTh | 0.42314675306170463 | 0.49986432750089 |
| LESMIS | 0.4363987209994051 | 0.44059265318262936 |
| GD01\_b | 0.20982142857142855 | 0.2238520408163266 |
| CAG\_mat72 | 0.20011033527579794 | 0.2198129952441364 |

References:

[1] Malliaros, Fragkiskos & Vazirgiannis, Michalis. (2013). Clustering and Community Detection in Directed Networks: A Survey. Physics Reports. 533. 10.1016/j.physrep.2013.08.002.

[2] Fortunato, Santo & Hric, Darko. (2016). Community detection in networks: A user guide. Physics Reports. 659. 10.1016/j.physrep.2016.09.002.

[3] B. W. Kernighan and S. Lin, ‘‘An efficient heuristic procedure for partitioning graphs,’’ Bell Syst. Tech. J., vol. 49, no. 2, pp. 291–307, Feb. 1970.

[4] Girvan, Michelle & Newman, Mark. (2002). Girvan, M. & Newman, M. E. J. Community structure in social and biological networks. Proc. Natl Acad. Sci. USA 99, 7821-7826. Proceedings of the National Academy of Sciences of the United States of America. 99. 7821-6. 10.1073/pnas.122653799.

[5] S. Cao and J. He, "A Label Propagation Algorithm based on Similarity and Weighted," 2020 IEEE 5th Information Technology and Mechatronics Engineering Conference (ITOEC), Chongqing, China, 2020, pp. 864-870, doi: 10.1109/ITOEC49072.2020.9141630.

[6] C. Tong, J. Niu, J. Wen, Z. Xie and F. Peng, "Weighted label propagation algorithm for overlapping community detection," 2015 IEEE International Conference on Communications (ICC), London, UK, 2015, pp. 1238-1243, doi: 10.1109/ICC.2015.7248492.

[7] Xiaoyu Ding, Jianpei Zhang, Jing Yang, “A robust two-stage algorithm for local community detection,” Knowledge-Based Systems,Volume 152, 2018, Pages 188-199, ISSN 0950-7051

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