Journal of Theoretical and Applied Information Technology
15th May 2024. Vol.102. No 9

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

AN ENHANCED COMMUNITY DETECTION METHOD USING LABEL PROPAGATION ALGORITHM WITH ANT COLONY OPTIMIZATION TECHNIQUE

D. DHANALAKSHMI[1], DR. G. RAJENDRAN[2]

[1]Research Scholar, Periyar University, Salem -11 and Assistant Professor of Computer Science and Applications, Vivekanandha College of Arts and Sciences for Women (Autonomous), Tiruchengode, Namakkal, Tamilnadu, India

[2]Associate Professor and Head, Department of Computer Science, Govt. Arts and Science College, Modakkurichi, Erode, Tamilnadu, India

E-mail:[1]dhanalakshmiphd2022@gmail.com [2]guru.rajendran@yahoo.com

ABSTRACT

The structure of a community is a crucial element in understanding complex networks. It provides valuable insights into both the arrangement and function of the network, aiding our comprehension of dynamic phenomena like epidemics and information propagation. While the Label Propagation Algorithm (LPA) is widely recognized for community detection due to its linear time complexity, it has a notable drawback. In comparison with other algorithms label propagation has advantages in its running time and amount of a priori information needed about the network structure. The biggest advantage of label propagation algorithm is that it owns excellent running time as well as simple algorithm process. LPA generates unstable community assignments, resulting in different combinations of communities with each execution on the same network. This unpredictability fosters instability and the emergence of large, less informative communities. To overcome these drawbacks, an Enhanced Community Detection approach, a combination of the Label Propagation Algorithm and Ant Colony Optimization (ECDLPA-ACO) technique has been proposed in this paper. ECDLPA-ACO not only propagates labels but also optimizes modularity measures by clustering similar vertices based on local similarities within the network. Experimental results on established social network datasets showcase the superiority of ECDLPA-ACO over comparable community detection algorithms like Louvain Algorithm, Infomap Algorithm, and traditional Label Propagation Algorithm. ECDLPA-ACO outshines in scalability, average execution time, modularity, and computational efficiency.

Keywords: *Community Detection, Louvain Algorithm, Infomap Algorithm, Label Propagation Algorithm, Ant Colony Optimization technique*

1. INTRODUCTION

Recently, Social Networks (SN) has gained immense popularity for their role in facilitating digital-era interactions [1]. Broadly, a social network is portrayed as a graph, defined as a system of interactions or relationships. In this representation, nodes symbolize individuals (actors), and edges signify connections or interactions between them [2]. The rise of platforms like YouTube, Facebook, Pinterest, Etsay, Twitter network, and others has propelled the analysis of network data into a critical research domain [3].

In the realm of social networks, communities are clusters of nodes where connections are dense within the group but sparse between groups [4]. Community detection stands as a significant aspect of Social Network Analysis (SNA), drawing substantial attention [5]. This process, deeply rooted in sociology, has now found widespread popularity in computational intelligence. Community detection plays a pivotal role in exploring diverse domains such as urbanization, social marketing, criminology, and beyond [6]. In the era of online social networks, the study of community detection has become integral to targeted marketing and influential campaigning [7]. It facilitates the analysis of intricate networks, allowing for the visualization of structures and the extraction of relationships. As the scope of complex networks, including social networks, con-

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

tinues to expand, researchers are uncovering the mechanics of epidemic spreading through the analysis of community structures in complex networks [8]. The Label Propagation Algorithm is favored for its efficiency, scalability, flexibility, noise tolerance, parameter-free nature, ability to handle overlapping communities, and adaptive characteristics, making it a valuable tool for community detection in a wide range of networks.
Ant Colony Optimization offers an effective and adaptable approach to community detection, providing robustness to noise, and scalability to large networks.

The major problems observed in community detection are,

- Modularity maximization suffers from resolution limits and extreme degeneracy.
- Finding the partition with maximal modularity is a challenging NP-complete problem.
- LPA has an uncertainty and randomness in the label propagation process, which may affect the stability and accuracy of community detection.
- Heuristics algorithms have been introduced to approximate the optimal partition but can be improved.
- Large-scale-free graphs with hubs pose scalability issues in community detection in distributed environments
- Modularity optimization algorithms are the only approach for detecting communities in large graphs [9]. However, their performance is weakened by the resolution limit.

The proposed ECDLPA-ACO algorithm has been developed with an objective to address the above problems. The working of the algorithm as follows:
ECDLPA-ACO extracts initial communities and iteratively updates node labels based on adjacent nodes. Fully connected sub graphs are detected and merged using defined degree functions. Communities are extracted based on node labels, and the method is explained in detail. The proposed Enhanced Community Detection method using Label Propagation Algorithm with ACO (ECDLPA-ACO) is compared with other algorithms. Assessing the method through modularity and computation time reveals positive outcomes across both small and large graphs. Particularly for very large networks, ECDLPA-ACO emerges as the superior option, thanks to its efficient computational time.

2. RELATED WORKS

Community detection stands as a foundational endeavor within social network analysis, with the primary objective of identifying clusters of nodes in a network exhibiting dense internal connections but sparse connections with other clusters. Three widely recognized algorithms for community detection include the Louvain method (Louvain modularity optimization), the Infomap algorithm, and the Label Propagation Algorithm. The subsequent explanation of each of these algorithms provides insight into their role in the context of community detection within social networks.

The Louvain algorithm (LV) [10], extracts community structure in vast networks. Based on modularity optimization, it employs a heuristic approach, with two iterative phases. In the first phase, nodes join communities, optimizing modularity by moving based on gains. The second phase builds a new network with attributed communities, linked by computed weights [11]. LV efficiently detects communities in large networks by optimizing modularity. LV offers speed, scalability, hierarchical structure, yet quality depends on network. One needs to consider size, granularity, efficiency, assumptions and pre processing for best results. Scalability issues arise, particularly in detecting communities in distributed environments with large-scale-free graphs.

Infomap algorithm starts by assigning nodes separate communities. Sequential random moves to neighbors follow, reducing map equation. Iterations continue till no further reductions. Graph is rebuilt, last-tier communities are replaced by nodes. Cycle repeats for new level till no more reductions. Infomap algorithm detects communities using info theory, optimizing walk encoding [12][13]. The results of Infomap algorithm are balanced partition graph, minimal description, inflow and outflow of a vertex. The drawback of infomap algorithm is lower computational time.
The Label Propagation Algorithm (LPA) detects communities in large networks with linear runtime for low-density networks. Nodes start with exclusive labels, adopting prevalent labels from neighbors iteratively. Linked nodes with identical labels form communities. The ad-

Journal of Theoretical and Applied Information Technology
15th May 2024. Vol.102. No 9

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

vantages of LPA includes (i) no parameter needed, (ii) simple and fast, and (iii) structure-based [14][15]. The limitations of LPA includes (i) sensitive to initial labels, (ii) performance varies and (iii) noise impact. The drawback of LPA is that it returns different solutions (some of them of poor quality) in different realizations and lower computational time.

3. ENHANCED COMMUNITY DETECTION USING ECDLPA-ACO

In this research work, we combined the Label Propagation Algorithm with the Ant Colony Optimization Algorithm [16], making certain modification to enhance its community detection capabilities. Our approach, known as ECDLPA-ACO, operates within the realm of social networks, delving into the exploration of concealed connections among individuals. This innovative algorithm is adept at uncovering all potential overlapping communities within the network graph. The undertaking unfolds in three distinct phases:

Initialization and Setup: The initial stage involves the setup process, where a series of algorithms are primed for action. This entails the selection of pertinent social network datasets, fine-tuning algorithmic parameters, and the creation of a unified graph model derived from the chosen datasets.

Construction of Community Detection Framework: The second step centers on the development of a community detection framework, characterized by a comprehensive, abstracted approach to community detection workflows. Here, we begin with the establishment of an initial community structure based on the dataset. Subsequently, nodes of interest are identified, leading to a reconstruction of the graph.

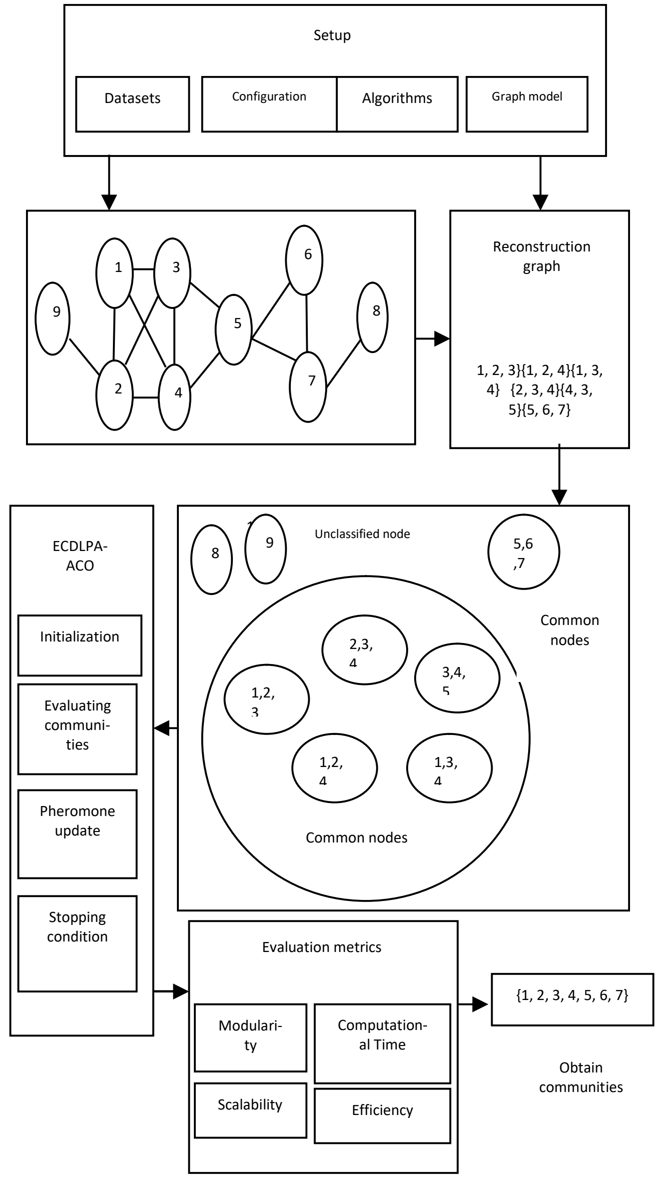

Figure 1: Structure of Proposed ECDLPA-ACO

Further, the algorithm discerns between classified and common nodes. The proposed methodology is then applied to assess communities, with an emphasis on maximizing modularity by reallocating vertices to their neighboring communities.

Community Extraction: The last stage involves extracting communities with the help of a thorough evaluation system that takes into account various aspects of community detection. We present a hybrid method that combines the Label Propagation Algorithm and Ant Colony Optimization Algorithm (ACO). In this work, the graph's edges are weighted based on similarity indices, boosting the algorithm's effectiveness in community detection.

This algorithm leverages Ant Colony Optimization to effectively calculate link rankings. It does so by computing pair wise distances between nodes and subsequently partitioning the network. It then endeavors to propagate labels and identify communities within networks through local optimization of modularity measures with individual ants. Figure 1 illustrates the structure of the Proposed ECDLPA-ACO algorithm.

Proposed ECDLPA-ACO algorithm

Input: Network G = (V, E), with a maximum number of iterations denoted as "master."

Output: Community set setc = {c1, …, ck}, and the number of communities "k".

Step 1: Assign a unique label to each node in the network. The structural influence, tsi(i, j), signifies the impact of node i on node j. If (i, j) is an edge in the network, calculate tsi(i, j) as the ratio of the actual number of connections from node i to the neighbors of j to the maximum possible connections. Therefore, tsi(i, j) is defined as:

$$tsi(i,j)=\frac{1+|T(i)\Omega T(j)}{|T(j)|} \quad (1)$$

Where T(i) represents the neighbor set of node i, deg(i) denotes the number of neighbors of node i (degree of i), and |T(i) Ω T(j)| signifies the count of shared neighbors between node i and j, essentially indicating the number of triangles connecting them. If node i is linked to all of node j's neighbors, it implies a substantial influence from i to j. It is crucial to note that the tsi from node i to node j may differ from the tsi from node j to node i. Therefore, the definition of tsi (i, j) is as follows:

$$tsi(j,i)=\frac{1+|T(i)\Omega T(j)}{|T(j)|} \quad (2)$$

Nodes with dense connections showcase higher tsi values compared to those with sparse connections. This emphasizes that nodes within a community exhibit elevated tsi values among themselves while minimizing their influence on nodes outside the community.

(1) Set the iteration number t = 1.

(2) For every node x ∈ X, update the node label iteratively following either formula (1) or (2). During this iterative process, each node adopts the label held by the largest number of its neighboring nodes. When multiple labels become equally prevalent, LPA randomly selects one of them to update the node label. Label updating methods can be categorized into either synchronous or asynchronous approache. The synchronous update method involves updating the label of node x in the t^{th} iteration based on the label of the adjacent nodes at the $(t-1)^{th}$ iteration. The formula is as follows:

$$c_x(t)=f(cx_1(t-1),\dots,C_{xk}(t-1),x_i\in N(x) \quad (3)$$

Where $c_x(t)$ represents the label of node x in the iteration t, and N(x) represents the neighborhood set of node x. Synchronous updating may lead to oscillation phenomena in a near-structure network, and asynchronous updating provides a viable solution to this issue [17].

In asynchronous updating, the label of a node x in the t^{th} iteration is modified using a combination of labels from two sources (i) a portion of labels from its adjacent nodes, which have already been updated in the current iteration, (ii) another portion of labels from the $(t-1)^{th}$ iteration, which have not yet been updated in the current iteration.

The formula is as follows:

$$C_x(t)=f(c_{x1}(t-1),\dots,c_{xm}(t-1)(t),\dots,c_{xk}(t)$$
$$x_i\in N(x)$$
(4)

Journal of Theoretical and Applied Information Technology
15th May 2024. Vol.102. No 9

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

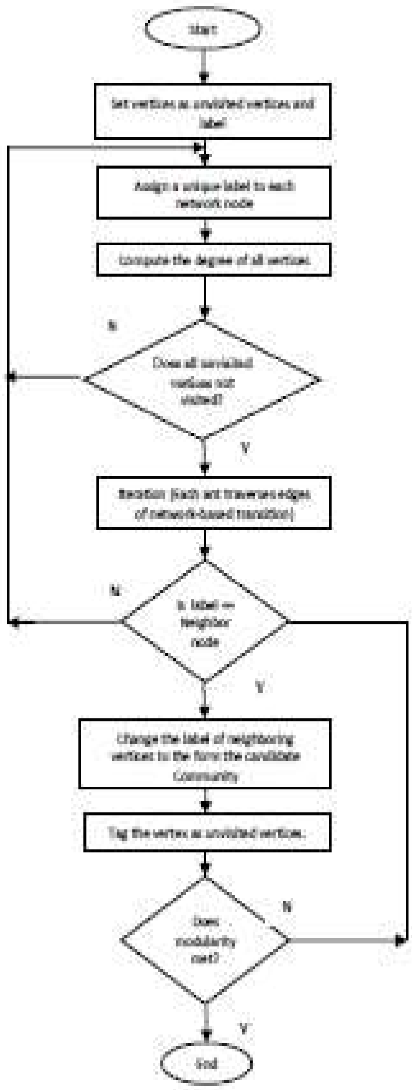

Figure 2: Flow chart of ECDLPA-ACO

Step2: Evaluating Communities

The process of ants visiting unvisited vertices through paths of similar vertices to construct the set of communities, based on their significance for each community, is iteratively performed until all vertices are labeled. Following this, the pheromones of paths traversed by ants are updated. The evaluation of the set of communities discovered by the ants is done using the modularity measures and it proceeds as follows:

$$Q(c) = \frac{1}{2m}\sum_{i,j}(A_{i,j} - \frac{k_i k_j}{2m})\delta(c_i, c_j) \quad (5)$$

Where A indicates the adjacency matrix of the input network, with A_{ij} being one when vertex c_i is connected to vertex c_j and zero otherwise. m represents the total number of edges in the input network, k_i denotes the degree of vertex c_i, and $\delta(c_i, c_j)$ is the delta function, yielding 1 if vertex c_i and vertex c_j are in the same community and 0 otherwise. If the new modularity value surpasses the average of all previously obtained modularity values, the pheromones are increased; otherwise, they are decreased. After estimating the tsi values between nodes, distinctive initial labels are assigned to each node in the network. In this uniform label initialization process, each node's label is adjusted to match the label of its neighbors based on the calculated tsi values (4).

$$l_j^{init} = \arg\max_{i \in r(j)} l(l_i, l_j).f(i, j) \quad (6)$$

Where l_j denotes the initial label of node j, f(i, j) is a function that returns 1 if tsi(i, j) is greater than tsi(j, i) and tsi(i, j) is less than L, where L denotes the label. Equation (6) can also be interpreted as every node in the network attempting to update the label of each of its neighbor nodes with its label if f(i, j) is satisfied.

Step 3: *Stopping conditions*

The cycle of calculating transition probabilities, ant traversal of vertices, label propagation, assembly of a candidate community set, assessment of the derived communities, and pheromone updates continues until further improvement is unattainable or the algorithm's iteration count surpasses a predefined limit.

4. EXPERIMENTAL RESULTS

In this study, scalability, average execution time, modularity, and computational efficiency were employed as the evaluation metrics widely recognized for assessing the quality of detected communities. Modularity is a particularly useful metric for quantifying the shared information between two distinct network partitions. To assess the algorithm's effectiveness, one can compare the discovered partition to a known real partition when the network's community structure is known.

In this work, the performance of ECDLPA-ACO was compared with Louvain algorithm, Infomap algorithm and LPA on three social network datasets (Twitter, Ego-Gplus, Ego-Facebook). The experiments were conducted on a computer with a 3.4 GHz Intel Core i7 CPU and 16.0 GB of RAM, implementing the algorithms in Python using NetworkX.

Journal of Theoretical and Applied Information Technology
15th May 2024. Vol.102. No 9

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Datasets:
1. Twitter: Consists of 'circles' from public sources, including node features, circles, and ego networks.
2. Ego-Gplus: Encompasses 'circles' from Google+, collected manually through the 'share circle' feature, with node features, circles, and ego networks.
3. Ego-Facebook: Comprises 'circles' from Face-book, collected via a survey. It includes node features, circles, and ego networks. For privacy, Facebook-internal ids have been replaced, and feature interpretation is obscured.

Scalability: The scalability of the proposed algorithm was compared with others using the Ego-Facebook, Twitter, and Ego-Gplus datasets. These datasets include a substantial number of nodes (n = 10,000), an average degree of k = 20, and community sizes within the range [minc, maxc] = [100, 500]. The scale-up metric acts as a benchmark to evaluate the efficiency of the par-allel algorithm in handling larger datasets.

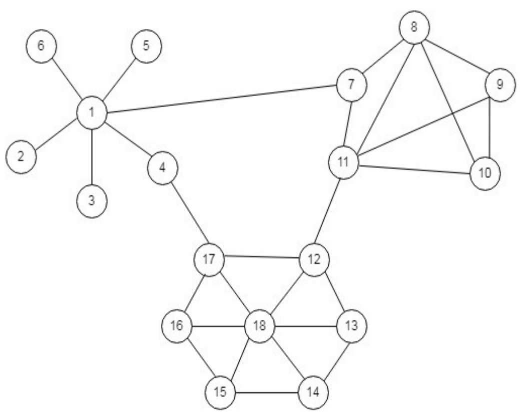

Figure 3: a) Input Graph

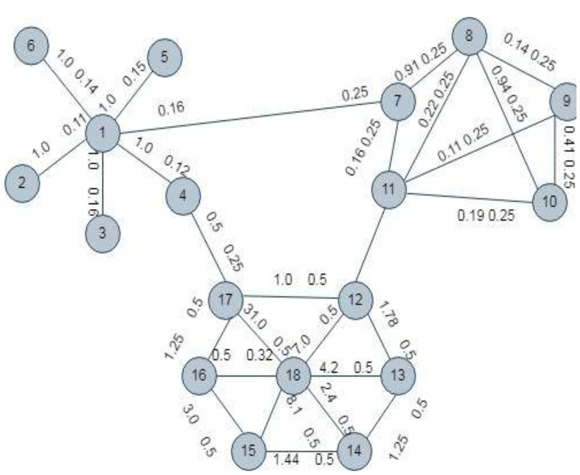

Figure 3: b) Influence as weights to each edge

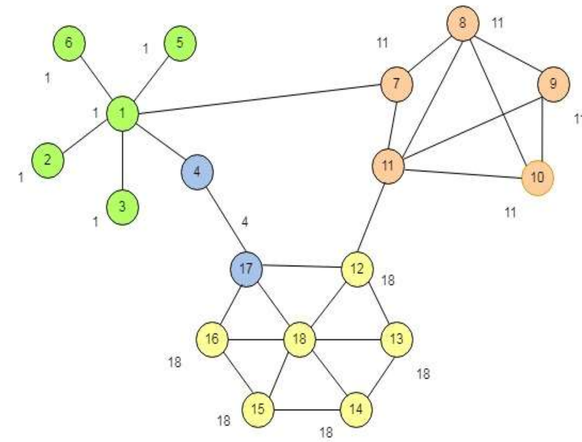

Figure 3: c) Initial label node

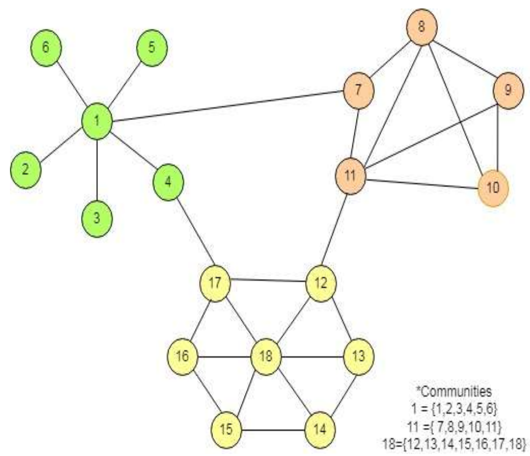

Figure 3: d) Final communities

When more iteration occur, which can be defined as

$$scaleup = \frac{T_1}{\tilde{T}_p}$$

Where T_1 denotes the Sequential Execution Time of the algorithm for processing the given dataset on a single node, and $\tilde{T}_p$ represents the Parallel Execution time of the algorithm for han-dling datasets p times larger on p time larger nodes.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 1: Comparison of scalability for 3 datasets

Algo-rithms	Louvain	Info-map	LPA	ECDLPA-ACO
Ego-Facebook	9.3	7.9	4.5	2.3
Twitter	12.4	10.2	7.2	5.4
Ego-Gplus	14.7	12.3	9.3	8.9

To validate the scalability of ECDLPA-ACO, the dataset size was increased from 100000 to 500000 data points. In Figure 4, the results highlight that ECDLPA-ACO exhibits superior scalability values compared to other algorithms. Both the number of iterations and dataset size demonstrate proportional growth. The proposed ECDLPA-ACO algorithm showcases outstanding scalability and adaptability when dealing with large datasets.

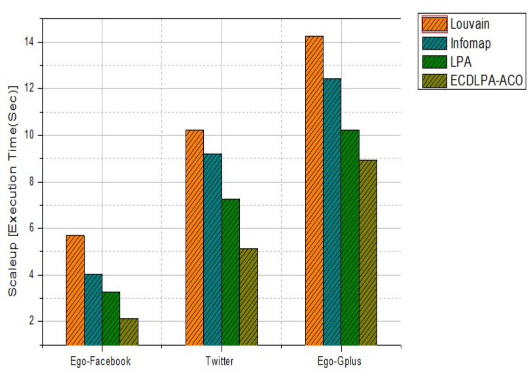

Figure 4: Scalability

Average Running Time: In Figure 5, the average running time of the proposed ECDLPA-ACO algorithm is plotted, demonstrating its superior speed compared to the Louvain, Infomap, and Label Propagation Algorithms.

The average execution times in seconds for the Louvain Algorithm, Infomap Algorithm, Label Propagation Algorithm, and ECDLPA-ACO on Twitter data, with parameters k=40 and [minc, maxc]=[200, 1000], were measured across a range of node numbers n ranging from 100,000 to 500,000. As shown in Figure 5, it becomes evident that the runtime of ECDLPA-ACO follows a linear scaling pattern with the dataset size and is notably faster in comparison to the Louvain, Infomap, and Label Propagation Algorithms.

Table 2: Comparison of Average Running Time for 3 datasets

Algo-rithms	Louvain	Infomap	LPA	ECDLPA-ACO
Ego-Facebook	12.155	9.414	5.918	3.113
Twitter	17.224	13.258	8.412	5.114
Ego-Gplus	22.123	18.245	13.451	8.219

ECDLPA-ACO efficiently completes the community detection process within less than 10.4 seconds for a network containing 500,000 nodes, showcasing both its speed and scalability. Consequently, ECDLPA-ACO outperforms the other tested algorithms in accurately identifying genuine community structures within the networks.

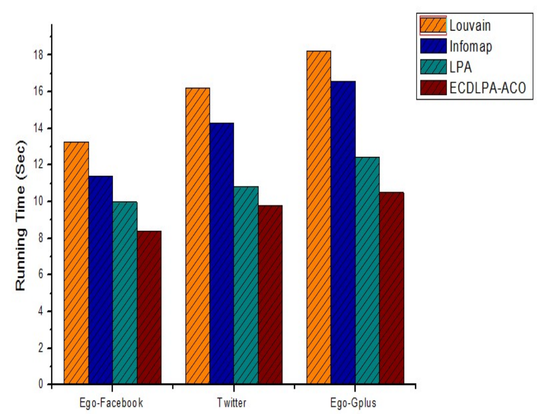

Figure 5: The average running time

The modularity of Communities:

One key metric for evaluating the quality of a community partition is referred to as "Modularity." In the context of an undirected graph G(E,

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

V), where each node is assigned to one of C potential communities, the modularity of the partition is defined as follows:

$$Q = \frac{1}{2m}\sum_{i,j\in v}[A_{ij} - \frac{k_i k_j}{2m}]\delta(c_i, c_j)$$

Where, A_{ij} are the elements of the adjacency matrix of G(E, V), k_i is the out-degree of node I , m = |E| э(c_i , c_j) is equal to 1 if i and j belong to the same community, and is equal to 0 otherwise.

Table 3: Comparison of modularity for 3 datasets

Algorithms	Louvain	Infomap	LPA	ECDLPA-ACO
Ego-Facebook	13.25	11.4	9.97	8.4
Twitter	16.23	14.27	10.8	9.8
Ego-Gplus	18.24	16.57	12.4	10.5

In this study, the modularity metric is employed as a measure to assess the effectiveness of the generated communities. The modularity metric results are visually represented in figure 6.

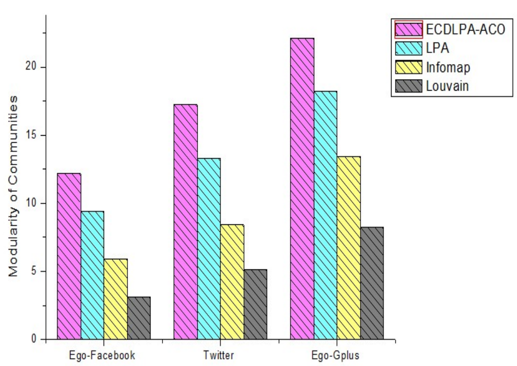

Figure 6: Modularity of Communities

The findings suggest that the proposed algorithm consistently produces commendable modularity values, establishing its efficacy and superiority over alternative algorithms in community detection. The ECDLPA-ACO algorithm consistently outperforms, exhibiting values ranging from 3.113 to 8.219. In contrast, Louvain, Infomap, and LPA exhibit less performance compared to the proposed algorithm.

Computational Time: The comparison involved the ECDLPA-ACO algorithm against the Louvain, Infomap, and Label Propagation algorithms. As illustrated in Figure 7, the proposed method consistently outperformed the baseline algorithms across all datasets and networks, showcasing superior performance, particularly in larger networks with tens of millions of nodes. Notably, the Louvain, Infomap, and Label Propagation algorithms struggled to handle datasets of such magnitude.

Table 4: Comparison of Computational Time for 3 datasets

Algorithms	Louvain	Infomap	LPA	ECDLPA-ACO
Ego-Facebook	5.69	4.03	3.25	2.13
Twitter	10.21	9.21	7.24	5.14
Ego-Gplus	14.25	12.44	10.23	8.92

Furthermore, when tasked with detecting communities of the same size, the ECDLPA-ACO algorithm exhibited a notable advantage in speed, surpassing the compared methods by two to three orders of magnitude. This speed advantage stems from the ECDLPA-ACO algorithm's reduced computation times per iteration. Additionally, its computational efficiency is enhanced as it calculates the modularity gain of a single node only in the two communities where the movement occurs. This sets the ECDLPA-ACO algorithm apart as one of the fastest and most efficient overlapping community detection algorithms available.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

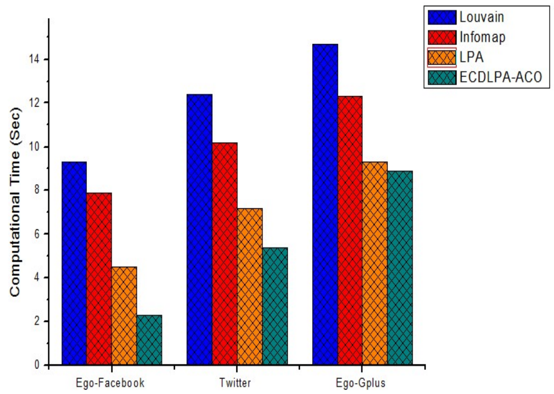

Figure 7: Computational Time

5. CONCLUSION

ECDLPA-ACO is an enhanced Label Propagation Algorithm, incorporated with Ant Colony Optimization to improve community modularity. The proposed algorithms eliminates the observed problems and it outperforms other algorithms viz., Louvain, Info map and Label Propagation Algorithms in terms of scalability, execution time, modularity, and computational efficiency, as demonstrated in experiments on social network datasets.

Future research avenues may explore the adaptability of ECDLPA-ACO for community detection in dynamic networks, where community structures evolve over time. Strategies for updating communities as the network undergoes changes could also be developed. Additionally, the extension of ECDLPA-ACO to support multi-resolution community detection would enable the algorithm to identify communities at various levels of granularity within a network, particularly beneficial for analyzing hierarchical networks. Further exploration could involve hybrid approaches, combining ECDLPA-ACO with other community detection algorithms or machine learning techniques, to capitalize on their complementary strengths.

REFERENCES

[1]. Basuchowdhuri P, Roy R, Anand S, Dr Srivastava R, "Spanning tree-based fast community detection methods in social networks", Innovations in Systems and Software Engineering, pp. 1-10, 2015. Doi:10.1007/s11334-015-0246-6

[2]. Coscia M, Giannotti F, Pedreschi D, "A classification for community discovery methods in complex networks", Statistics Analysis Data Mining, vol. 4, pp. 512-546, 2011. Doi:https://doi.org/10.48550/arXiv.1206.3552

[3]. Zhao Z, Feng S, Wang Q, Huang J. Z, Williams G. J, and Fan J, "Topic oriented community detection through social objects and link analysis in social networks", Knowledge Based Systems, vol. 26, pp. 164-173, 2012. https://doi.org/10.1016/j.jksuci.2015.07.001

[4]. Fortunato S, "Community detection in graphs", Physics Reports, vol. 486, pp. 75-174, 2010. DOI:https://doi.org/10.48550/arXiv.0906.0612

[5]. Xia Z and Bu Z, "Community detection based on a semantic network", *Knowledge-Based Systems,* vol. 26, pp. 30-39, 2012.Doi: 10.1016/j.knosys.2011.06.014

[6]. Liu Z and Hu B, "Epidemic spreading in community networks", Europhysics Letters, vol. 72, no. 2, pp. 315–321, 2005. Doi: http://dx.doi.org/10.1155/2015/316092

[7]. Chaudhary L and Singh B, "Community detection using an enhanced louvain method in complex networks", in Proceedings of the International Conference on Distributed Computing and Internet Technology, pp. 243–250, Springer, Bhubaneswar, India, January 2019. Doi:10.1007/978-3-030-05366-6_20

[8]. Coscia M, Giannotti F, and Pedreschi D, "A classification for community discovery methods in complex networks", Statistical Analysis and Data Mining, vol. 4, no.5, pp.52-546, 2011. Doi: https://doi.org/10.1002/ sam.10133

[9]. Ren G and Wang X, "Epidemic spreading in time-varying community networks", Chaos, vol. 24, no. 2, Article ID 068701, 2014. Doi: 10.1103/PhysRevE.98. 062303

[10]. Fortunato S and Barthélemy M, "Resolution limit in community detection", *Proc. Natl. Acad. Sci. U.S.A.* 104 (2007) 36-41.

[11]. Good B. H, de Montjoye Y A and Clauset A, "The performance of modularity maximization in practical contexts", *Phys. Rev. E* 81 (2010) 046106.

[12]. Khadivi, Ajdari Rad A and Hasler M, "Community-detection enhancement by proper weighting", *Phys. Rev. E* 83 (2011) 046104

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[13]. Brandes U, Delling, D, Gaertler M, Goerke R, Hoefer M, Nikoloski Z. and Wagner D, "Maximizing modularity is hard", (2006). Doi: arXiv:physics/0608255.

[14]. Rajakumar B.R, "The Lion's Algorithm: A New Nature-Inspired Search Algorithm", Procedia Technology, Elsevier, vol. 6, pp. 126-135, 2012. Doi: https://doi.org/10.1016/ j.protcy.2012.10.016

[15]. Bauer H .H. de longh and Di Silvestre I, Lion (Panthera leo), "social behaviour in the West and Central African savannah belt, Mammalian Biology-Zeitschrift fUr Saugetierkunde", Elsevier, vol. 68(4),p p. 239-243, 2003. Doi:10.1078/1616-5047-00090

[16]. Wajeed M.A, Adilakshmi T, "Adopting ant colony optimization for supervised text classification. In Proceedings of the 2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI)", Jaipur, India, 21–24 September 2016; pp. 2562–2566.

[17]. Le B D, Shen H, Nguyen H, Falkner N, "Improved network community detection using meta-heuristic based label propagation", Appl. Intell., vol. 49, no.4,pp 1451 - 1466, 2019