# ENSEMBLE COMMUNITY DETECTION ALGORITHM

Project Submitted to the
SRM University AP, Andhra Pradesh
for the partial fulfillment of the requirements to award the degree of

**Bachelor of Technology**
**in**
**Computer Science & Engineering**
**School of Engineering & Sciences**

submitted by

**Chenna Kesava Jasti(AP20110010502)**

Under the Guidance of

**Dr. Anirban Bhar**


**Department of Computer Science & Engineering**
SRM University-AP
Neerukonda, Mangalgiri, Guntur, 522 240

# DECLARATION

I undersigned hereby declare that the project report **Ensemble community Detection Algorithm** submitted for partial fulfillment of the requirements for the award of degree of Bachelor of Technology in the Computer Science & Engineering, SRM University-AP, is a bonafide work done by me under supervision of Dr. Anirban Bhar. This submission represents our ideas in our own words and where ideas or words of others have been included, I have adequately and accurately cited and referenced the original sources. I also declare that I have adhered to ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in our submission. I understand that any violation of the above will be a cause for disciplinary action by the institute and/or the University and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed the basis for the award of any degree of any other University.

| | | | |
|---|---|---|---|
| Place | : ..... | Date | : May 14, 2024 |
| Name of student | : Chenna Kesava Jasti | Signature | : ..... |
| Name of student | : | Signature | : ..... |
| Name of student | : | Signature | : ..... |
| Name of student | : | Signature | : ..... |

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

**SRM University-AP**

**Neerukonda, Mangalgiri, Guntur, 522 240**


## CERTIFICATE

This is to certify that the report entitled **Ensemble community Detection Algorithm** submitted by **Chenna Kesava Jasti, , Chenna Kesava Jasti, Karthikeya Pala** to the SRM University-AP in partial fulfillment of the requirements for the award of the Degree of Master of Technology in in is a bonafide record of the project work carried out under my/our guidance and supervision. This report in any form has not been submitted to any other University or Institute for any purpose.

| | |
|---|---|
| Project Guide | Head of Department |
| Name : Dr. Anirban Bhar | Name : Prof. Niraj Upadhayaya |
| Signature: ..... | Signature: ..... |

newpage

# ACKNOWLEDGMENT

I wish to record my indebtedness and thankfulness to all who helped me prepare this Project Report titled **Ensemble community Detection Algorithm** and present it satisfactorily.

I am especially thankful for my guide and supervisor Dr. Anirban Bhar in the Department of Computer Science & Engineering for giving us valuable suggestions and critical inputs in the preparation of this report. I am also thankful to Prof. Niraj Upadhayaya, Head of Department of Computer Science & Engineering for encmyagement.

my friends in my class have always been helpful and I am grateful to them for patiently listening to my presentations on my work related to the Project.

Chenna Kesava Jasti
(Reg. No. AP20110010502)
B. Tech.
Department of Computer Science & Engineering
SRM University-AP

# ABSTRACT

Communities, which are collections of closely connected nodes, are examples of hidden structures seen in complex networks. However, because individual detection techniques have limits, locating these groups can be difficult. Detecting ensemble communities takes on this problem a group of professionals. Similar to how mixing different models improves machine learning, this method uses several algorithms to examine the same network data. The secret is to successfully integrate these disparate results to produce a more comprehensive image. We investigated the MeDOF technology in our project. The ability to distinguish between communities that overlap have hazy boundaries or are entirely distinct makes this approach special. One of the earliest methods created especially for these intricate community structures is MeDOF.We conducted comprehensive research on Synthetic networks, where the genuine communities are known, to test their efficacy.

In this project, we examine the importance of community identification procedures in real-time circumstances and the possibility of using ensembled approaches to get around some of the drawbacks of standard methods. We empirically evaluate the effectiveness of ensembled approaches in revealing hidden community structures in complex networks, improving our knowledge of network dynamics and enabling well-informed decision-making across a range of fields. We highlight the significance of group discovery in networks and show how combining different approaches enhances our comprehension and optimization of networks in our environment.

# CONTENTS

# LIST OF TABLES

# LIST OF FIGURES

# Chapter 1

# INTRODUCTION TO THE PROJECT

In many different sectors, community detection has become an indispensable technique for deciphering the complex organizational patterns found within networks. In the current world, networks are all around us. They symbolize networks of interconnected entities, such as people in social networks, proteins in biological networks, and nodes in technical infrastructures. For a variety of reasons, it is crucial to identify communities, or tightly connected groups of nodes, inside these networks.

For a variety of applications in diverse sectors, community detection is essential. For instance, in social networks, community identification aids in understanding social dynamics, locating powerful people, and forecasting patterns of information spread. Community detection in biological networks makes it easier to comprehend intricate biological processes by assisting in the identification of functional modules within cellular pathways. Community detection also helps with vulnerability identification, resource allocation optimization, and system efficiency in technical networks like the Internet and transit systems.

## 1.1 ENSEMBLE COMMUNITY DETECTION

One method to increase the robustness and accuracy of locating communities in complicated networks is ensemble community detection. It takes inspiration from machine learning's ensemble approaches, which combine

several models to provide results that are superior to those of any one model. We use various community detection techniques using identical network data. Every algorithm may possess advantages and disadvantages when it comes to recognizing communities according to distinct network characteristics. The results of each of these distinct algorithms are then aggregated in some fashion, where the outputs may be community assignments for every node or some sort of community structure metric. By combining this data, communities inside the network may be detected in a more robust and trustworthy manner. Ensemble methods can overcome the shortcomings of individual approaches and discover more accurate communities by merging results from numerous algorithms. Compared to single algorithms, ensemble approaches are less vulnerable to noise or outliers in the network data [2].

# Chapter 2

# MOTIVATION

The rationale for selecting an ensemble community detection project is its capacity to overcome the drawbacks of conventional community detection algorithms and enhance the precision and resilience of community identification in intricate networks. Ensemble methods offer a promising way to improve performance, boost resilience to noise and variability, and provide deeper insights into the underlying structure of networks by mixing the results of numerous algorithms and utilizing their respective strengths.

In addition, conducting this kind of study advances the field's understanding of network analysis, machine learning, and data mining—applications that span social networks, the biological sciences, and cybersecurity. In the end, working on this project gives us the chance to investigate cutting-edge methods, address practical issues, and advance the subject.

The potential of ensemble community detection to combine different approaches and improve the resilience and accuracy of community identification algorithms can be a driving force behind choosing this kind of project. It addresses the drawbacks of solitary approaches and provides the chance to investigate state-of-the-art network analysis tools. Furthermore, ensemble approaches frequently produce more trustworthy outcomes, which makes them useful for a variety of applications, including social network analysis.

In the end, ensemble community detection enables us to derive a deeper understanding of complex systems in a variety of domains, such as social networks, biological networks, and beyond, by revealing hidden

patterns and providing better insights into the network's organization.

# Chapter 3

# LITERATURE SURVEY

Chaakraborty et al . [1] proposed a survey on ensemble detection in complex networks by using the MeDOF algorithm that detects both disjoint, overlapping, and fuzzy. They concluded that ensemble methods generally decrease the impact of the degeneracy of solution significantly.

Mukherjee et al. [8] proposed a model for audience co-exposure networks by simulating audience behavior in an artificial media environment. They used eight different detection algorithms to group different types of parameters resulting in sixteen communities.

Yang et al. [2] focused on identifying communities using eight different algorithms on artificial networks. The accuracy and computation time of the findings are measured. This is an excellent attempt to push community detection research into greater heights. Because of its thorough methodology, in-depth analysis, and theoretical insights, it adds value to the body of literature and can be used as a benchmark for other research in the area.

Christensen et al. [4] suggested a method for identifying communities through the use of psychological data. Their study constitutes an important contribution to the domains of psychology and network science, providing researchers with useful advice on the selection of suitable community detection algorithms for the analysis of psychological data.

Lancichinetti et al. [9] paper constitute an important turning point in the study of networks. Researchers are given the means to thoroughly evaluate and contrast community detection methods by creating a common

set of benchmark graphs and assessment measures. In addition to improving our theoretical knowledge of complex networks, this seminal study has real-world implications for information retrieval, computational biology, social network analysis, and many other fields.

# Chapter 4

# DESIGN AND METHODOLOGY

The two main types of community detection techniques are hierarchical and non-hierarchical. By identifying communities at various scales, hierarchical approaches seek to disclose the composition of smaller communities within bigger communities. Conversely, non-hierarchical techniques ignore nested structures in favor of concentrating on discovering communities at a single level. Spectral clustering, which divides the network according to the eigenvectors of the adjacency matrix, modularity optimization, which aims to maximize the quality of community assignments based on network properties, and random walk-based techniques, which communities based on the likelihood of passing through them during random walks on the network, are a few examples of these techniques.

## 4.1 COMMUNITY DETECTION ALGORITHMS

In this research, we preprocessed the dataset before implementing the majority of the detection algorithms on the network. The scale of operations carried out as a function of the number of nodes (N) and edges (E) in the network is represented by the list of algorithms that we have taken into consideration and used notation for.

**INFOMAP:** Rosvall et al. proposed this algorithm. It determines communities by analyzing the information flow over a network using random walks. The first step in this approach is to encode the network into

modules in a way that optimizes the amount of original network information. Subsequently, it transmits the signal over a capacity-limited channel to a decoder. In an attempt to decipher the message, the decoder creates a list of potential candidates for the original graph. More data about the original network has been shared the fewer candidates there are. In O(E), this algorithm executes [1].

**LOVIAN:** Blondel introduced this algorithm. In contrast to the Fast-greedy method, this greedy approach optimizes modularity differently. This approach moves a node to the community of a neighbor with which it earns the maximum positive contribution to modularity, after first assigning each node in the network a different community. Until no more progress is possible, all nodes go through the aforementioned phase again. When there is just one node remaining or when increasing the modularity in a single step is not possible, each community is then treated as a single node on its own, and the second phase is repeated. O(NlogN) is the multilevel algorithm's computational complexity [1].

**SPIN-GLASS**:Reichardt Bornholdt made the initial proposal for this algorithm. Its foundation is the Potts model. The method's fundamental idea is that nodes of the same spin state should be connected via edges, whereas nodes with different states (belonging to separate communities) should be disconnected. Therefore, the goal of this approach is to determine the spin glass model's ground state using a Potts Hamiltonian. The system's free energy has been reduced by the use of simulated annealing. The computational complexity of this algorithm is about O(N**3.2) in a sparse graph [1].

**LABEL PROPOGATION** : According to the algorithm, which was first presented by Raghavan et al., every node in the network is assumed to

be a member of the same community as the majority of its neighbours. This is how it operates: Every node begins with a distinct label, such as "community." Nodes are handled one after the other in a randomised sequence. Every node takes on the label that most of its neighbours have. Until every node gets the same label as the majority of its neighbours, this process is repeated. The label propagation algorithm's computational complexity is commonly represented as O(E), where E is the number of edges in the network. This suggests that the approach is computationally efficient for large-scale networks since its complexity grows linearly with the number of edges in the network [1].

**WALKTRAP** : Pon and Latapy's algorithm is a hierarchical clustering technique that relies on the idea that nodes belong to their respective communities and that short-distance random walks typically stay inside them. It computes the distances between neighbouring nodes. Based on their proximity, two neighbouring settlements are combined into one, and the separations between them are changed.The procedure of merging is carried out (N1) times. This algorithm's computational complexity is O(EN 2). In sparse networks, the complexity decreases to O(N 2 log(N)), where E is substantially lower than N*2. This suggests that the approach is effective for sparse networks because its complexity scales linearly with the number of edges and quadratically with the number of nodes [1].

**LEADING EIGENVECTOR**: Using the modularity matrix's eigenvalues and eigenvectors, Newman's approach maximises modularity. The modularity matrix's leading eigenvector must be determined in order to assess it. Based on this eigenvector, divide the graph into two halves to maximise the improvement in modularity. Maintain the network's division while optimising the gain in modularity at each stage. When there is no

positive modularity contribution, stop. On a sparse graph, the computational complexity of every graph bipartition is O ( N (E + N)), or O(N**2). This indicates that the approach is effective for large-scale networks because its complexity grows linearly with the number of edges and nodes in the network [1].

# Chapter 5

# IMPLEMENTATION

The project began as we got our hands on the networking file with a source node, destination node, and weight, we got onto the CSV file and the first thing that slipped into our was to pass this through base community detection algorithms so that we could observe the output and compare it with ground-truth vector,

We observed that each base algorithm has its unique way of finding communities info-map makes the most optimal pair or triplets of nodes into a community,label-prop works by shuffling the nodes to find the best in community companion for each node,lovian workers by adjusting the modularity and so on, We decided to go for a common graph and community library ”igraph”, after getting through the first phase of passing them through the base community detection algorithm we got sets of multiple communities for each algorithm we saved them in a dictionary for later purposes.

After, we calculated the individual conductance score of each resultant community and store them in a csv file named Conductance-scores the R-project was brimming by the time we completed this so we moved to python to exercise some skills and we calculated the WCA Weighted co-associatition matrix of the community based on the below criteria:

## 5.1 WEIGHTED CO-ASSOCIATION MATRIX

This matrix had multiple criteria where we needed to build a matrix:

$$\widetilde{A} = \{\widetilde{a_{ij}}\}_{N\times N} \tag{5.1}$$

In which:

$$\widetilde{a_{ij}} = \frac{1}{M}\sum_{m=1}^{M} w_{m_i} \cdot \delta_{mij} \tag{5.2}$$

Where:

$i, j$ : nodes existing in the network file,

$M$ : Number of community detection algorithms used,
And:

$$w_{mi} = \text{1-CON}(\text{Clsm}(o_i)) \tag{5.3}$$

Where:

$w_{mi}$ is 1-conductance score of the cluster formed by a base algorithm $m$ which has the node $i, j$ coexist

And:

$$\delta_{mij} = \begin{cases} 1, & \text{if Clsm}(o_i) = \text{Clsm}(o_j), \\ 0, & \text{otherwise.} \end{cases} \tag{5.4}$$

After the WCA matrix is formed we needed to chose an algorithm to pass it into ,we thought about it and decided to go with spectral clustering as suggested by our mentor we passed the matrix through the spectral clustering algorithm and we got a certain number of communities as output we decided to compare them with the ground truth vector.

# Chapter 6

# HARDWARE/ SOFTWARE TOOLS USED

This chapter discusses the details of the hardware used in the implementation of the Project along with the software tools.

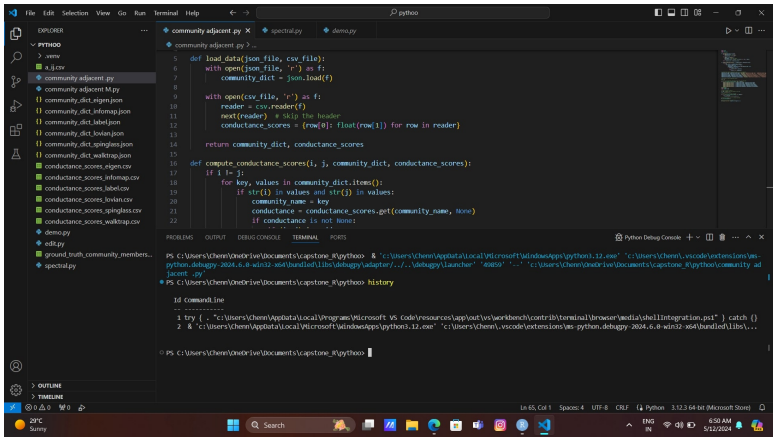

Figure 6.1: Visual Studio IDE

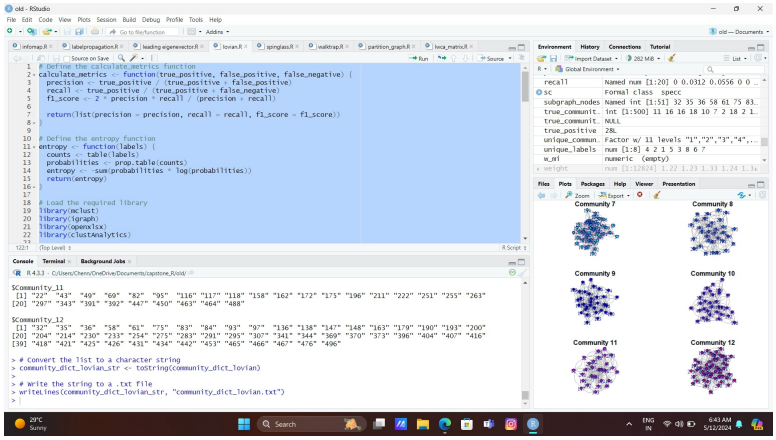

Figure 6.2: R Studio

# Chapter 7

# RESULTS & DISCUSSION

## 7.1 DATASET

We received an Artificially generated Synthetic network from the LFR benchmark algorithm we used $\mu$ =0.6 which is a comparatively disrupted network the dataset set contained 500 nodes and each edge existing has a weight of its own we got three columns Source node destination node and the weight between these nodes in the format.

Table 7.1: Dataset Info

| Source node | Destination node | weight |
|---|---|---|
| 1 | 3 | 0.54 |
| 1 | 6 | 0.24 |
| 1 | 4 | 0.40 |
| 2 | 5 | 0.76 |
| 2 | 3 | 0.45 |
| . | . | . |
| . | . | . |
| . | . | . |
| 500 | 499 | 0.52 |

## 7.2 RESULT

The initial stages of the project where we passed the network file into the base detection algorithm resulted in outstanding communities however there were a few challenges regarding the label propagation and infomap as it makes optimal communities of twins or triplets of nodes, whereas the label

propagation algorithm swaps the nodes already present in the community until an optimal community is observed, we later retrieved these clusters we got from the base community and calculated the conductance scores of the communities,by using these conductance scores and the nodes that are present in the community after applying algorithm we made a weighted Co-association Matrix(WCA) which is stored in a csv file the below table represents the f1 score and Nmi of the base community detection algorithm compared to the ground truth vector given by the benchmark:

Table 7.2: Clustering Evaluation Metrics

| Method | NMI | F1 Score |
|---|---|---|
| Infomap | 0.870699 | 0.5 |
| Label Propagation | 0.5729572 | 0.06576402 |
| Eigen Vector | 0.8196966 | 0.4054795 |
| Lovian | 0.9823472 | 0.7252747 |
| Spinglass | 0.826323 | 0.4052288 |
| Walktrap | 0.9562413 | 0.8918919 |

## 7.3 WCA MATRIX

Getting the WCA matrix was a complex task as we needed to implement the formula to all 500 nodes as a pair of i and j the matrix is calculated on the basis of existence of nodes i and j in a cluster and the conductance score of that particular cluster

## 7.4 SPECTRAL CLUSTERING

Later,moveing on to the final part of the code we applied spectral clustering on the WCA matrix we got to get a distinctive set of communities based on out criteria ”n”, Where n is the number of clusters expected

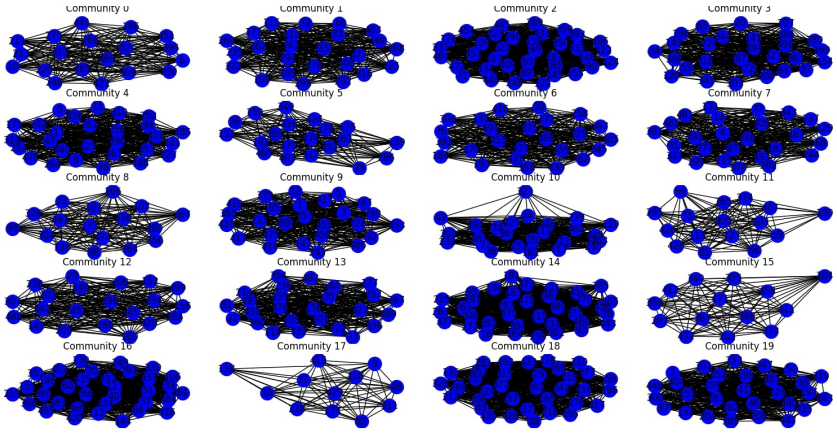


Figure 7.1: Final community

**F1 Score: 0.6037909870582451**
**NMI: 0.5334529503294169**

# Chapter 8

# CONCLUSION

Coming to the Conclusion We got to know that the custom Function Present in Igraph and Cluster packages whether it is Python or R the algorithms used Greatly effects the results you can change the resultant clusters of the base algorithm by simply giving one more input parameter like Spins in spin-glass, Steps in walktrap and Resolution in Lovian the results are greatly affected by the input you give for these particular parameters, so all the synthetic networks that are given by LFR benchmark datasets also give a Ground truth vector along with the networking file without the reference of this ground truth vector there are infinite number of combinations to make a community. however, coming to the view of the most optimal community it may exist at Some value where:

$$-\infty < \text{Resolution} < \infty$$

$$\mathbf{0} < \mathbf{steps} < \infty$$

So calculating this is not a simple task however we came a little close to finding the communities by monitoring F1 score and NMI scores of the communities to the ground truth vector, at the end of the project.

## 8.1 SCOPE OF FURTHER WORK

### 8.1.1 What is future direction in a project?

we may plan to go with the diverse Coverage of applications, Instead of taking the conductance score of the community we may Use the ECI of the community the node I belongs to, By applying hierarchical clustering after getting the WCA matrix we can cut the resultant dendrograms dynamically to get the minimum number off clusters needed, Instead of trying to apply the base algorithm with default functions we can create our own custom functions,we can Observer the count of communities as we change Resolution, spins, and Steps after all this is just half a step towards a newer community to observe it, we can tune the number of clusters parameter in spectral clustering.

# REFERENCES


[1] **Chakraborty, T. (2020).** Ensemble Detection and Analysis of Communities in Complex Networks, vol.1. IIIT Delhi, India.

[2] **ZhaoYang. (2016).** A Comparative Analysis of Community Detection Algorithms on Artificial Networks, vol.2.

[3] **Duncombe, J. U. (2019).** Locally Weighted Ensemble Clustering. IEEE Transactions on Cybernetics, version 3 December 2019.

[4] **Toth, C. (2021).** Community Detection via Random Walk Modeling. Synwalk Press, vol.36.

[5] **Li, Y., Liu, G., Lao, S.-y. (2013).** Overlapping Community Detection in Complex Networks Based on the Boundary Information of Disjoint Community, vol.22, Issue 11. [Paper presented at a conference or published in a journal in November 2013].

[6] **Newman, M. E. J. (2006).** Modularity and Community Structure in Networks, vol.74, Issue 3. [Paper presented at a conference or published in a journal in June 2006]

[7] **Javed. M A, Younis, M. S. Latif, S. Qadir, J. Baig, A.(2018).** A Comprehensive Review of Community Detection Algorithms for Large-Scale Networks, vol.108. [Paper presented at a conference or published in a journal in April 2018].

[8] **Mukerjee, S. (2021).** A Systematic Comparison of Community Detection Algorithms for Measuring Selective Exposure in Co-exposure Networks. Scientific Reports, vol.11.

[9] **Lancichinetti. A, Fortunato. S Radicchi. F.(2008).**Benchmark graphs for testing community detection algorithms. vol.78 Issue 4. [Paper presented at a conference or published in a journal in May 2008].