IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

COMMUNITY DETECTION IN SOCIAL NETWORKS

Dr.R.Thangaranjan, R.Anusha, B.hema, M.Karthi

Professor, Scholar, Scholar, Scholar
Department of Information Technology,
Kongu Engineering College, Perunduari, Erode, India

Abstract: Community detection is a vital task in the analysis of social networks, where the goal is to identify groups of nodes that are highly interconnected and exhibit similar characteristics. This problem has received significant attention due to its wide range of applications in various fields, such as marketing, recommendation systems, and sociology.In recent years, numerous algorithms have been proposed to solve this problem, ranging from traditional clustering techniques to modern deep learning-based methods. In this paper we experiment with several algorithms to find overlapping and disjoint communities. Louvain and Label propagation algorithms are used to detect overlapping communities. Girvan-Newman and K-clique are used to detect the disjoint communities. A Graph Neural Network (GNN) is a type of neural network designed to operate on graph-structured data, which is a collection of nodes and edges that represent relationships between them. A GNN based community detection is also carried out and compared with existing methods

***Index Terms* - community detection, overlapping, disjoint, social networks.**

I. INTRODUCTION

Community detection in social networks is the process of identifying groups or communities of nodes in a network that are more densely connected within themselves than with the rest of the network. In other words, community detection aims to partition a network into cohesive and densely connected groups of nodes, where nodes within a group have similar properties or play similar roles in the network.Social networks, such as Facebook, Twitter, and LinkedIn, are examples of networks where community detection is particularly relevant. Social networks consist of nodes representing individuals or entities, and edges representing the connections or relationships between them. Community detection in social networks can provide insights into the structure and dynamics of the network, help identify influential nodes, detect anomalies or outliers, and improve recommendation systems.Community detection algorithms are based on different criteria, such as connectivity, similarity, and modularity. Some commonly used algorithms for community detection include Louvain algorithm, Label Propagation algorithm, Girvan-Newman algorithm, and k-clique algorithm. These algorithms differ in their approach to identifying communities, and their suitability depends on the characteristics of the network and the specific research questions being addressed.Overall, community detection in social networks is an important research area with numerous applications in various domains, including social science, computer science, and data science.

II. LITERATURE REVIEW

Hedia Zardi ,Bushra Alharbi, Walid Karamti, Hanen Karamti and Eatedal Alabdulkreem proposed a technique for finding overlapping communities in dynamic social networks[1]. This model makes it possible to monitor how communities change over time. To assess the node's relationship to a community, define a similarity metric. Three crucial factors were taken into account for this measurement: the message distribution, attribute similarity, and network topology. As a result, each community's members are similar, they communicate effectively with one another, and they are structurally connected. Like the majority of the proposed studies for community detection, assume that people are solely connected by "positive relationships".)

E.A.Abdulkreem , H.Zardi and H.Karamti proposed a model called DSCAN(Dynamic Structural Clustering Algorithm for Networks) and GAD(Genetic Algorithm). In terms of runtime speed and modularity, DSCAN and GAD are on par with or even superior to their static community detection counterparts. When dealing with big dynamic social networks, which are continually changing their structure, this speed increase and negligible loss of modularity are excellent.

It is possible to execute DSCAN after just a few network modifications, keeping the community structure updated more frequently. It is expensive and even impossible to analyse social networks using static community detection approaches given their exponential growth in size

D.Jin , X.Wang,D.He, J.Dang, W.Zhang proposed a novel Bayesian probabilistic model for link community discovery and devised an effective variational approach for learning the model, which examines network topologies and link contents.[17]In this paper, we mainly concentrated on creating an accurate model and, when c is unknown, finding the number of communities c via model selection. Since the datasets we used have those two numbers being equal, we took into consideration situations where the numbers of communities c and subjects k are the same. Our approach works well in circumstances where c not equal to k.

Yunlei Zhang, Bin Wu, Nianwen Ning,Chenguang Song and Jinna Lv defined a model of Dynamic topical community detection. A DTCD model that unifies network structure, text, and time was presented by us. The community and issue are modelled by DTCD as latent variables, and the temporal fluctuations of the community and topic are modelled by a multinomial distribution over time. Topic, community, and their temporal fluctuations can be discovered using DTCD by inferring the latent variables. On two real-world datasets, we also ran experiments and carried out dynamic topical community detection. In jobs requiring community detection and topic extraction, DTCD performs better than other comparison techniques. Furthermore, DTCD discovers temporal fluctuations in communities and subjects

H.Zare,M.Hajiabadi,M.Jalili proposed a Network community structure and nodal feature detection using a generative probabilistic technique. The suggested method, PFCD, took into account both nodal characteristics and network topology. In our suggested paradigm, the various effects of nodal properties on community structures are explored. An effective probabilistic approach is used to infer the proposed model. The suggested model was assessed using a range of small to big real network datasets using conventional evaluation metrics. The results on actual networks demonstrated the suggested model's great performance and offered extremely encouraging findings for the recognition of community structures based on a network aligned with nodal features.

M.Lu, Z.Zhang, Z.Qu, and Y.Kang proposed a novel label-based overlapping community called LPANNI(Label Propagation Algorithm with Neighbor Node Influence). Bigger will increase the time cost, while LPANNI uses parameter to adjust the topological information between node pairs to estimate node similarity. It is only possible to use LPANNI on homogeneous networks.

Wenjian Luo , Senior Member, IEEE, Daofu Zhang, Hao Jiang, Li Ni, and Yamin divide the community identification method into three steps and use dynamic membership functions in order to identify the local community. This can address the issues with the earlier techniques. Several synthetic and actual datasets are used to test our methods, and the results are contrasted with those of the earlier techniques. The techniques presented in this paper perform well on undirected networks. The new dynamical membership functions should be developed for the directed network. In the coming research, we'll keep examining this problem.

Yongping Luo, Li Wang, Shiwen Sun, and Chengy Xia proposed a novel community detection algorithm based on local data, and we have also proposed a more generalized signed modularity function and the signed local modularity function to assess the quality of the neighbourhood based on the existing modularity function. A dynamic expansion approach based on local knowledge for global community detection is suggested using the proposed modularity function. By dynamically increasing the community size and adding nodes via an initial node that has a community label, the method eventually uncovers the community structure buried within the network. The suggested algorithm's output in various networks demonstrates that it can produce respectable results in both signed and unsigned networks.

Panagiotis Liakos, Katia Papakonstantinopoulou, Alexandros Ntoulas, and Alex Deli proposed COEUS, a cutting-edge graph stream community discovery method that builds communities out of seed sets of nodes. As far as we are aware, COEUS is the first streaming algorithm to perform community detection utilizing space that is proportional to the number of edges without placing any limitations on the order in which edges enter the stream. When processing a stream of edges, COEUS keeps a small amount of data on each network, including the degrees of the nodes, the nodes participation in communities, and the nodes that make up each community we're interested in. Additionally, we suggest two techniques that considerably increase the efficacy of our strategy.

III. Proposed Work

Community detection in social networks can be a valuable tool for understanding the structure and dynamics of these networks. Several machine learning and deep learning techniques are proposed to find the communities with similar properties. The dataset for the experiment is collected from stand-ford large network dataset collection a public API. Musae- github dataset contains large social network of GitHub developers. The csv file dataset is converted into graphs using Pytorch package. The community detection algorithms are applying on the converted dataset. To detect overlapping communities Louvain and Label propagation algorithms are used. To find the disjoint communities Girvan-Newman and k-clique percolation are used. Also the GNN is used to train the dataset and detect the communities. The performance of the algorithms are measure in terms of modularity and accuracy.

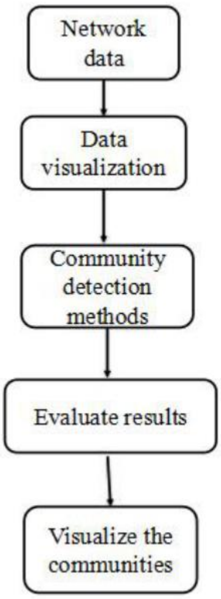

Figure 3.1 Architecture

3.1Dataset description

The data set for the experiment is collected from an public API .It is a large social network of GitHub developers. Github users who have minimum of ten starred repositories are represented as nodes and the mutual followers relationship are represented as edges.The location, starred repositories, employer, and email address are used to extract the node features.

Directed	No
Node features	Yes
Edge features	No
Node labels	Yes
Nodes	37700
Edges	289003

Figure 3.2 Dataset statistics

3.2 Dataset pre-processing

A.CSV to graph conversion

Pytorch - Pytorch is a fully featured framework for building deep learning models, which is a type of machine learning that's commonly used in applications like image recognition and language processing.The csv file dataset is loaded using pandas. Convert the data into graph using torch.tensor() which takes a list and converts it to a PyTorch tensor.Create a PyTorch dataset object using the tensor. The dataset object is used to hold the data and labels for the machine learning task. Split the dataset into training and test sets using PyTorch's SubsetRandomSampler or DataLoader classes.

3.3 Community detection methods

a) Louvain method

The Louvain algorithm is a community detection algorithm that aims to identify the underlying community structure of a network. The algorithm works by iteratively optimizing a quality function called modularity, which measures the strength of division of a network into communities. The algorithm starts by assigning each node in the network to its own community. Then, it iteratively optimizes the modularity by merging communities that would result in a higher modularity. This process continues until no further improvement in modularity is possible.In the first step of each iteration, the algorithm computes the modularity gain that would result from merging each node's community with its neighbors' communities. The modularity gain is defined as the difference between the modularity of the network before and after the merge. The algorithm then selects the merge that results in the largest modularity gain and updates the community assignments accordingly. .

Modularity Q = (1/2m) ∑[(Aij - ki kj/2m) δ(ci,cj)]

In the second step of each iteration, the algorithm constructs a new network where the communities identified in the previous step are collapsed into single nodes. This new network is used as the input for the next iteration of the algorithm.The Louvain algorithm is a highly efficient and scalable algorithm that can be applied to very large networks. It has been used in a wide range of applications, including social network analysis, biological network analysis, and network visualization.

b) Label propagation

Label propagation is a type of semi-supervised learning method that aims to infer labels for unlabeled data points based on the labeled data points. The process of label propagation starts with a set of labeled data points, which are used to train a model. The labeled data points are assigned a specific label or class based on their characteristics or attributes. Once the model has been trained on the labeled data points, it is used to predict labels for the unlabeled data points.The process involves propagating the labels from the labeled data points to the unlabeled data points. This is done by using the similarities between the data points. The assumption is that data points that are similar are likely to belong to the same class. The similarity between data points is usually measured using a distance metric, such as the Euclidean distance or cosine similarity.Algorithm works by first initializing the labels of the labeled data points. These initial labels are then propagated to the neighboring unlabeled data points. The propagation of the labels is done iteratively, with each iteration updating the labels of the unlabeled data points based on the labels of their neighboring data points.The label propagation algorithm continues iterating until the labels of all the data points have converged or until a maximum number of iterations have been reached. The final labels assigned to the unlabeled data points are used to evaluate the performance of the label propagation algorithm.

c) Girvan-Newman

The Girvan-Newman algorithm is a popular community detection algorithm used to identify the underlying community structure in a network. The algorithm works by iteratively removing edges from the network based on their edge betweenness centrality, which is a measure of how frequently an edge appears on the shortest paths between all pairs of nodes in the network. At each step of the algorithm, the edge with the highest betweenness centrality is removed from the network, which results in the network being split into two or more disconnected components. This process is repeated until all edges have been removed or the desired number of communities has been identified.

$$\textit{\textbf{Betweeness}}=b/(n-1)(n-2)$$

The Girvan-Newman algorithm is based on the intuition that edges that lie between communities have higher betweenness centrality than edges that lie within communities. By removing these edges, the algorithm aims to break apart the network into its constituent communities.One of the drawbacks of the Girvan-Newman algorithm is that it can be computationally expensive, especially for large networks. To address this, several modifications and optimizations have been proposed, such as using approximate methods for calculating edge betweenness centrality or using parallel computing techniques.

d) K-clique percolation

The k-clique is a process in graph theory that involves identifying sub graphs within a larger graph that are complete, meaning that all nodes within the sub graph are connected to each other. This process is useful in many applications, including social network analysis, image segmentation, and machine learning.To perform the k-clique process, one must first select a value for k, which represents the size of the sub graphs to be identified. Then, the algorithm searches for all sub graphs within the larger graph that have k nodes and are complete. This involves checking all possible combinations of k nodes within the graph to determine if they form a complete sub graph.Once all k-cliques have been identified, they can be used to gain insights into the structure of the larger graph. For example, k-cliques can be used to identify groups of closely connected nodes within a social network, or to segment an image into regions that are highly connected. In machine learning, k-cliques can be used as features for classification tasks, or to identify patterns within data that may be useful for predicting future outcomes.

Overall, the k-clique process is a powerful tool for analyzing complex graphs and identifying meaningful substructures within them. By identifying complete sub graphs of a specific size, researchers can gain insights into the underlying patterns and relationships within the larger graph, and use this information to make predictions or inform decision-making.

e) GNN(Graph Neural Network)

Graph Neural Networks (GNNs) have been widely used in community detection tasks due to their ability to learn and encode node features and graph structure. The process of using GNNs for community detection typically involves several steps.Firstly, the graph is represented as a matrix or set of matrices, where each row and column represents a node and each element in the matrix represents the weight or strength of the edge between two nodes. This matrix is then transformed into a graph data structure that can be fed into the GNN model.Secondly, the GNN model is trained using a supervised or unsupervised approach, depending on the availability of labeled data. In unsupervised learning, the model learns to cluster nodes into communities based solely on the graph structure, while in supervised learning, the model is trained to predict the community label of each node based on labeled data.Thirdly, the GNN model is used to perform community detection on the input graph. During this process, the model propagates information across the graph to update the node representations based on the features of their neighbors. This allows the model to capture the community structure of the graph and assign each node to a particular community. Finally, the quality of the community detection results is evaluated using various metrics such as modularity, conductance, and coverage. These metrics measure the degree of homogeneity within communities and the degree of separation between communities.

IV. Analysis

In this study, several approaches is used to find the communities in the network which have similar properties. The performance evaluation of each algorithm must be measured to determine which method provides the highest accuracy. Numerous evaluation criteria, including accuracy, modularity, NMI and f1- score, are taken into account when evaluating the project.

$$\text{Accuracy} = (TP + TN) / (TP + FP + FN + TN)$$

TP: True Positive - Observation rightly detected as positive
TN: True Negative - Observation rightly detected as negative
FP: False Positive - Observation wrongly detected as positive
FN: False Negative - Observation wrongly detected as negative

Modularity is a measure of the quality of a community structure in a network, which compares the actual number of edges within communities to the expected number of edges under a null model. Modularity is widely used in community detection to evaluate the quality of the identified communities.

$$\text{Modularity } Q = (1/2m) \sum[(Aij - ki\ kj/2m)\ \delta(ci,cj)]$$

$NMI = 2 * I(C, P) / (H(C) + H(P))$
I(C, P): Mutual information between sets C and P
H(C): Entropy of set C
H(P): Entropy of set P

V. Result and Discussions

The communities of the github network were detected by communities detection algorithms for both overlapping and disjoint communities. Louvain and label propagation algorithms were used to detect overlapping communities. Girvan Newman and K-clique percolation were used to find disjoint communities. GNN is used to find the K(number of communities) value to detect in the communities.On comparison of overlapping community detection algorithm Louvain performs better than label propagation. To find disjoint communities K-clique performs better than Girvan Newman algorithm .

Algorithm	Modularity	NMI
Louvain algorithm	0.8	0.9
Label propagation	0.6	0.7
Girvan Newman	0.5	0.7
K-clique	0.6	0.7

Figure 5.1. Performance comparison

a) Performance of louvain method
Louvain algorithm is used to find the overlapping communities in the network The algorithm acquires 70% of modularity and 90% of NMI score.

b) Performance of K-clique
To detect disjoint communities Girvan-newman and K-clique are used. K-clique Percolation acquires 60% of modularity and 70% of NMI score.

c) Performance of GNN
Gnn is used to evaluate the k-value that is used in neither Girvan-Newman or K-clique algorithm.It acquires 87% of accuracy.

VI. Conclusion

The project was implemented using several community detection algorithms for detect the group of nodes which have similar characteristics as a community. A github musae dataset is collected from an open API. The performance of the algorithms is calculated in terms of modularity and NMI. The detected communities were visualized using pytorch. Based on the experiment, it is found that the Louvain algorithm and k-clique percolation performs better than other algorithms. Similarly, using GNN the k-value is estimated and hybridized with convolutional algorithms.

VII. References

[1] E. A. Abdulkreem, H. Zardi, and H. Karamti, ‘‘Community detection in dynamic social networks: A multi-agent system based on electric field,’’ Int. J. Adv. Comput. Sci. Appl., vol. 10, no. 1, pp. 493–504, 2019.

[2] P. Agarwal, R. Verma, A. Agarwal, and T. Chakraborty, ‘‘DyPerm: Maximizing permanence for dynamic community detection,’’ 2018, arXiv:1802.04593. [Online]. Available: https://arxiv.org/abs/1802.04593

[3] Y. Asim, R. Ghazal, W. Naeem, A. Majeed, B. Raza, and A. Kam-ran, ‘‘Community detection in networks using node attributes and modularity,’’ Int. J. Adv. Comput. Sci. Appl., vol. 8, no. 1, pp. 382–388,2017.

[4] G. A. Bello Lander, ‘‘Multi-objective graph mining algorithms for detecting and predicting communities in complex dynamic networks,’’ M.S. thesis, North Carolina State Univ., Raleigh, NC, USA, 2017.

[5] S. K. Bisma and A. N. Muaz, ‘‘Network community detection: A review and visual survey,’’ 2017, arXiv:1708.00977

[6] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, ‘‘Fast unfolding of communities in large networks,’’ J. Stat. Mech.: Theory Exp., vol. 2008, no. 10, Oct. 2008, Art. no. P10008.

[7] R. Cazabet, ‘‘Detection of dynamic communities in temporal networks,’’ Ph.D. dissertation, Univ. Paul Sabatier-Toulouse III, Toulouse, France, 2013.

[8] T. Chakraborty and A. Chakraborty, ‘‘OverCite: Finding overlapping communities in citation network,’’ in Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining (ASONAM), Aug. 2013, pp. 1124–1131.

[9] A. Clauset, M. E. J. Newman, and C. Moore, ‘‘Finding community structure in very large networks,’’ Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top., vol. 70, no. 6, Dec. 2004, Art. no. 066111.

[10] E. Desmier, M. Plantevit, C. Robardet, and J.-F. Boulicaut, ‘‘Cohesive co-evolution patterns in dynamic attributed graphs,’’ in Proc. Int. Conf. Discovery Sci., 2012, pp. 110–124.

[11] S. Fortunato, ‘‘Community detection in graphs,’’ Phys. Rep., vol. 486, nos. 3–5, pp. 75–174, Feb. 2010.

[12] S. Fortunato and D. Hric, ‘‘Community detection in networks: A user guide,’’ Phys. Rep., vol. 659, pp. 1–44, Nov. 2016.

[13] M. Girvan and M. E. J. Newman, ‘‘Community structure in social and biological networks,’’ Proc. Nat. Acad. Sci. USA, vol. 99, no. 12, pp. 7821–7826, Jun. 2002.

[14] D. Greene, D. Doyle, and P. Cunningham, ‘‘Tracking the evolution of communities in dynamic social networks,’’ in Proc. Int. Conf. Adv. Social Netw. Anal. Mining (ASONAM), N. Memon and R. Alhajj, Eds. Aug. 2010, pp. 176–183.

[15] J. Hopcroft, O. Khan, B. Kulis, and B. Selman, ‘‘Tracking evolving communities in large linked networks,’’ Proc. Nat. Acad. Sci. USA, vol. 101,no. 1, pp. 5249–5253, Apr. 2004.

[16] S. S. Hoseini and S. H. Abbasi, ‘‘A new method for community detection in social networks based on message distribution,’’ Int. J. Comput. Sci. Netw. Secur., vol. 45, no. 5, pp. 298–308, 2017.

[17] D. Jin, X. Wang, D. He, J. Dang, and W. Zhang, ‘‘Robust detection of link communities with summary description in social networks,’’ IEEE Trans. Knowl. Data Eng., early access, Dec. 10, 2019, doi: 10.1109/TKDE.2019.2958806.

[18] K. Guo, T. Zhu, and G. Hui Li, ‘‘Incremental dynamic community discovery algorithm based on improved modularity,’’ in Proc. 2nd IEEE Int. Conf. Comput. Commun. (ICCC), Oct. 2016, pp. 2536–2541.

[19] A. Mahfoudhi, H. Zardi, and M. A. Haddar, ‘‘Detection of dynamic and overlapping communities in social networks,’’ Int. J. Appl. Eng. Res., vol. 13, no. 11, pp. 9109–9122, 2018.