International Journal on Emerging Technologies **11**(1): 101-107(2020)

ISSN No. (Print): 0975-8364
ISSN No. (Online): 2249-3255

Overlapping Community Structure Detection using Twitter Data

Sathiyakumari K.[1] and Vijaya M.S.[2]

[1]Assistant Professor, Department of Information Technology,
PSGR Krishnammal College for Wome, Coimbatore (Tamil Nadu), India.
[2]Associate Professor, Department of Computer Science,
PSGR Krishnammal College for Women, Coimbatore (Tamil Nadu), India.

(Corresponding author: Sathiyakumari K.)
(Received 09 October 2019, Revised 05 December 2019, Accepted 13 December 2019)
(Published by Research Trend, Website: www.researchtrend.net)

ABSTRACT: Overlapping community detection is progressively becoming a significant issue in social network analysis (SNA). Faced with massive amounts of information while simultaneously restricted by hardware specifications and computation time limits, it is difficult for clustering analysis to reflect the latest developments or changes in complex networks. Techniques for finding community clusters mostly depend on models that impose strong explicit and/or implicit priori assumptions. As a consequence, the clustering effects tend to be unnatural and stay away from the intrinsic grouping natures of community groups. In this method, a process of enumerating highly cohesive maximal community cliques is performed in a random graph, where strongly adjacent cliques are mingled to form naturally overlapping clusters. These approaches can be considered as a generalization of edge percolation with great potential as a community finding method in real-world graphs. The main objective of this work is to find overlapping communities based on the Clique percolation method. Variants of clique percolation method such as Optimized Clique percolation method, Parallel Clique percolation method have also been implemented. This research work focuses on the Clique Percolation algorithm for deriving community from a sports person's networks. Three algorithms have been applied for finding overlapping communities in the sports person network in which CPM algorithm discovered more number of communities than OCPM and PCPM. CPM overlapping algorithm discovered 198 communities in the network. OCPM algorithm found 180 different sizes of communities. PCPM algorithm discovered 170 communities and different size of the node in the graph. The community measures such as size of the community, length of community and modularity of the community are used for evaluating the communities. The proposed parallel method found a large number of communities and modularity score with less computational time. Finally, the parallel method shows the best performance is detecting overlapping communities from the sports person network.

Keywords: Clique Percolation Method (CPM), Modularity, Overlapping, Optimized Clique Percolation method (OCPM), Parallel Clique Percolation Method (PCPM), Social Network Analysis.

Abbreviations: CPM, Clique Percolation Method; OCPM, Optimized Clique Percolation method; PCPM, Parallel Clique Percolation Method; SNA, Social Network Analysis.

I. INTRODUCTION

Community Structures can be defined as the division of network into various modules like groups, clusters, communities which are connected to each other. The modules comprises of nodes and edges which have dense connections between the nodes within the same modules but have sparse connections between nodes in the different modules and these modules are formed using graph theory. It emphasizes on finding communities in a network using different algorithms and optimizing the solution. There are number of approaches and tools available to generate community structure.

The Clique Percolation Method is investigation of the changing spatial organization of the network. Non overlapping community detection algorithms assign nodes into exclusive communities and, when results are mapped, these techniques may generate spatially disjointed geographical regions, an undesirable characteristic for geographical study. Alternative overlapping community detection algorithms agree to overlapping membership where a node can be a member of different communities. Such a structure simultaneously accommodates spatial proximity and spatial separation which happen with respect to a node in relation to other nodes in the system. Applying such a structure in geographical analysis helps protect well-established principles regarding spatial relationships within the geography discipline. The result can also be planned for exhibit and correct interpretation. The CPM is chosen in this study due to the complete connection within cliques which enables the formation of highly connected networks. However, the CPM has been shown to be among the best performing overlapping community detection algorithms. Detecting communities in a network only exposes certain characteristics of the spatial organization of the network, rather than providing explanation of the spatial-network patterns revealed. Full interpretation of the prototype must rely on the attribute data and additional information. This may demonstrate the value of an amalgamated approach in geographical analysis using both social network analysis and spatial analysis techniques.

The main objective of this research work is to find overlapping communities based on the Clique percolation method using direct network data. Variants of clique percolation method such as Optimized Clique percolation method, Parallel Clique percolation method have also been implemented. Overlapping community structures are retrieved which helps to identify inter and intra group interaction between the various nodes in the network.

II. RELATED WORK

There are a few existing community detection algorithms that are based on finding cliques to detect highly-overlapping communities. Researchers interested in clustering and social network community detection have designed and investigated different algorithms of various complexities.

The original community detection algorithm based on modularity maximization was Newman's [1] greedy hill climbing agglomerative algorithm; this was followed by a more efficient version, known as the Fast Modularity algorithm [2]. These algorithms begin with a trivial partition, with very low modularity, in which each vertex was in a separate community. The algorithms then repeatedly join the pair of communities those results in the greatest increase in modularity, until the desired number of communities is gained.

The algorithms of [2, 1] gave poor results in some cases and some communities tend to become excessively large, as the hill-climbing algorithm has no information at the beginning about which communities to merge. Wakita and Tsurumi [3] addressed this problem by modifying the quality function to incorporate balanced community sizes as well as modularity.

An alternative method was to start from a partition in which communities contain more than just one vertex. PBD [4] formed initial communities using random walkers. In PKM [5], each initial community was one of the highest-degree vertices and its neighbors. Com Tector [6] first found all maximal cliques, and then converted these to an initial partition by identifying community kernels and then assigned each vertex to the most appropriate kernel.

The clique percolation technique found communities in a fully connected network by finding adjacent cliques in the graph [7]. The k means clustering algorithm [8] partitioned a population in k clusters. Every node in a graph is assigned to cluster with the closest mean. An iterative scan technique was employed [9]. In such an approach, nodes were iteratively added or removed in order to improve a density function. The algorithm was implemented using shared-nothing architectural approach. The approach spreads data on all the computers in a distributed setup and used master–slave architecture for clustering. In such an approach, the master may easily become a bottleneck as the number of processors and the data size increases.

Parallelism has been proposed as a means to alleviate computational costs [10, 11]. Heuristically evaluated the propinquity, i.e. the probability that a pair of nodes is involved in a coherent community [10]. The original network was updated by adding (removing) edges if the propinquity is higher (lower) than a given threshold.

A parallel method was used to update propinquity incrementally, in order to reflect network changes. The system was able to extract meaningful communities from the huge Wikipedia linkage network. Rather than introducing a new definition of community. Sadi *et al.*, [11] proposed a method to reduce the size of the networks. In parallel, they used a heuristic to locate quasi-cliques and assigned them as nodes in a reduced graph to be used with standard community detection methods. These reduced graphs had a size which is approximately one half of the original size. Alternatively than exploiting parallelism, computational costs can be mitigated by designing efficient heuristics and greedy local function optimizations.

From the above literature study it is observed that the clique percolation algorithm can be applied effectively to find communities in a social network. In our previous work, the algorithm found large number of sub communities in the network, whereas it is unable to discover overlapping communities in the network. These overlapping communities are needed to identify the interaction between the various nodes in the network. This work focuses on clique percolation approach for deriving overlapping community from a sports person's network and uses graph measures for evaluating the communities. The graph measures are used size of the community and modularity score of the network.

III. METHODOLOGY

The proposed framework includes three components: (i) input logic (ii) process logic (iii) output logic. The input logic deals with data extraction from twitter data and conversion to network structure. The process logic uses Clique percolation method (CPM) and its variants such as Optimized Clique Percolation Method (OCPM), Parallel Clique Percolation Method (PCPM) to find overlapping communities. The output logic generates sub communities of the input network and its measures. The architecture of the proposed system is shown in Fig. 1.

Clique Percolation is an effective algorithm for detecting overlapping communities in large graphs. Before the creation of the Clique Percolation clustering algorithm, most techniques used to find communities in large networks required the division of networks into smaller connected clusters by the removal of key edges which connect dense sub-graphs.

Groups are totally linked sub-graphs of k vertices. K-group adjacency means two k-groups are nearest if they allocate k1 vertices. A k-clique chain is a sub-graph which is the union of a series of adjacent k-cliques. Two k-cliques are k-clique-connected if they are parts of a k-clique series. A k-clique percolation cluster or component is a maximal k-clique-connected sub-graph, meaning it is the merger of all k-cliques that are k-clique-connected to an exacting k-clique.

The algorithm finds k-cliques, which correspond to fully attached sub-graphs of k nodes. It distinguishes a community as the maximal combination of k-groups that can be achieved from each other during a series of adjacent k-groups. First, all of the existent maximal k-clique percolation clusters for the given k are discovered.

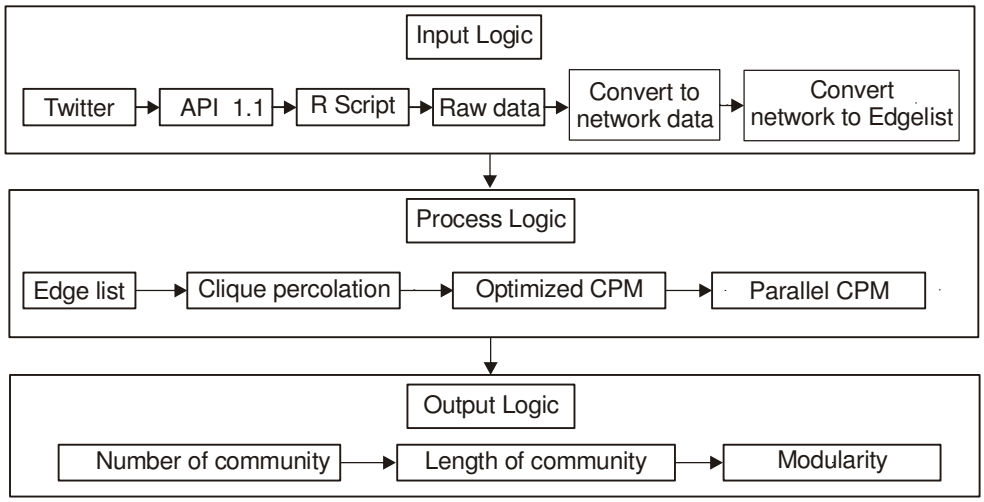


Fig. 1. Clique percolation method overlapping community detection framework.

The k-clique percolation group is a maximal k-clique linked sub-graph. This is unstated as the blending of all k-cliques that are k-clique-connected to an exacting k-clique. The percolation shift obtains place when the chance of two vertices being fixed by an edge conquers the threshold pc (k) = [(k – 1)N] –1/(k–1). It is proven in [12] that the success in overlapping community detection with clique percolation on randomized networks translates to success on real networks. This is because only small clusters are expected for any k at which the network is below the transition point, but large clusters also appear, which corresponds to locally dense structures.

Algorithm: Clique Percolation Algorithm (CPA) to find overlapping community is given below

– The network, G and the group size, k are the input.

– K-clique is a group with k nodes where a group is a complete sub graph.

– Several K-cliques communities are formed from inclusive network.

– From the network K-cliques community forms a combination of all k-groups.

– A merger of k-group is formed which can be accomplished from each other through a series of closest k-groups.

– If and only if two k-cliques are sharing k-1 nodes only than it is said to be adjacent k-cliques.

Input: The network, G and the group size, k.

Output: Community structure, C.

The most arithmetically exclusive segment in the method is the clique-graph making procedure. It performs an extensive seek on the space of groups, looking for couples that distribute k-1 nodes. In the basic execution there are two nested for loops comparing cliques and then executing n*n reiterations.

Variants of clique percolation method: OCPM and PCPM are two variants of CPM. OCPM optimized approaches distinguish couples of groups in the exploration space. The method executes an exhaustive search of couples that distribute k-1 nodes. The two nested *for loops* over the same list of groups communicate to symmetric matrix-based seek space that can be diminished in investigating either the upper or lower part. This execution, therefore decreases the number of reiterations to n*(n-1)/2.

PCPM parallelizes the search of couples of groups utilizing the number of cores that CPU have at their discarding. It also requires defining the number of clusters to parallel the execution. The algorithm can be parallelized and show performance results on a shared-memory platform. The i^{th} iteration of the for loop in the algorithm computes the size of the largest clique that contains the vertex v_i. During such a concurrent computation, different processes discover maximum groups of different sizes and for the pruning steps to be most effective, the current globally largest maximum group size needs to be communicated to all processes as soon as it is discovered.

Modularity: The modularity Q is proposed via Newman and Girvan [13] as a degree of the nice of a selected division of a network, and is defined as follows:

Q = (range of edges inside communities) – (predicted wide variety of such edges)

The modularity Q computes the fraction of the edges within the community that join vertices of the same type, i.e., intra-community edges, less the expected value of the similar number in a community with the equivalent network structure however with random connections among the vertices If the variety of inside community edges is not any higher than random, Q = zero. A price of Q this is near 1, which is the maximum, indicates strong community shape. Q usually falls inside the range from 0.3 to 0.7 and extreme values are rare.

IV. EXPERIMENT AND RESULTS

The real-time network data is collected using the twitter application programming interface 1.1 for this investigates work. Twitter is a fast expanding, free and a very quick social network that has emerged as a major source of information. Twitter is a free micro-blogging social networking service website that launched in March 2006 and has amassed more than 336 million users as in March 2018 and is expanding extremely fast. It is a popular social networking site used globally and is ranked as the most popular micro-blogging website.

Nine thousands records of friends and followers list of the famous cricket player have been crawled from his twitter account. The data is collected at run time from twitter network using R 3.5.1, arithmetical tool. Fig. 2 shows the cricket player's initial community network and Fig. 3 depicts the relationship types of community network such as friends and followers of the initial network. This set of relationships has 6831 vertices and 7095 edges.

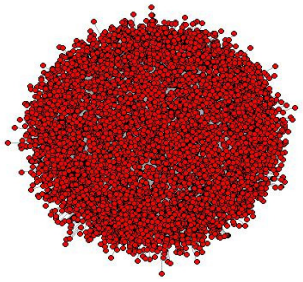


Fig. 2. Cricket player's initial network.

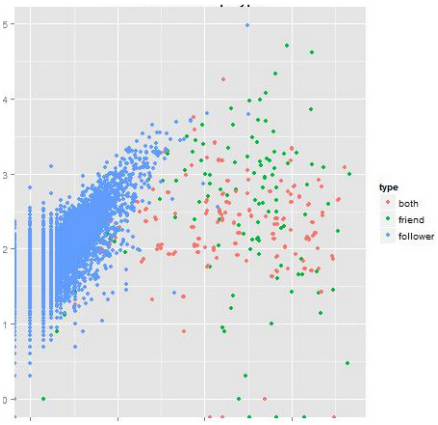


Fig. 3. Friends and followers network

A. Results of Clique Percolation

CPM overlapping algorithm discovered 198 communities in the network out of which eighty nine communities have 134 members in the community network. 89 communities are having the large number of nodes accounting to 110 to 200 sizes of the nodes in the community. Seventy seven communities are having the medium number of nodes that is 80 to 100 in the community. Thirty two communities are having the small number of nodes and community sizes ranging from 40 to 70 in the network. CPM algorithm found 198 dense communities in the network, which depicts this network has large number of nodes and shares the link for each node. Fig. 4 shows the cricket player's CPM overlapping community network.

Almost 50 % (89) communities found by CPM have large number of nodes in the cricket players about network. Each community has dense network and shares more information between each node. Seventy seven communities have the medium number of nodes in the network. Incoming and out coming nodes are interactive between each community. Seventy communities have smaller number nodes in the network which depicts that it is less interactive between nodes in the network. Fig. 5 shows clique percolation overlapping communities.

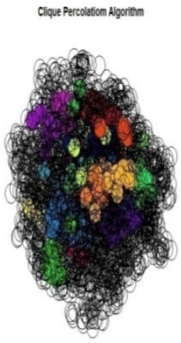
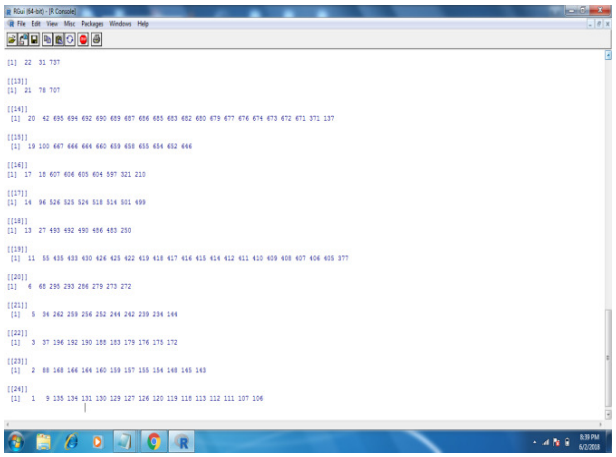


Fig. 4. CPM Community Size and network.



[[1]] 91 100 2295 2294 2287 2286 335] 66 92 1817 1814 1803 73 104 1960 1959 1956 1955 1954 1953 1951 1950 1949 1947 1946 1945 1944 1658 1356 1308 887 796 717 66 92 1817 1814 1803 1191 1190 1188 1187 1186 1185 1184 1180 1176 1111

[[2]] 74 103 1969 1966 1965 1377 726 725 724 722 189 66 92 1817 1814 1803 1191 1190 1188 1187 1186 1185 1184 1180 176 1111

[[3]] 73 104 1960 1959 1956 1955 1954 1953 1951 1950 1949 1947 1946 1945 1944 1658 1356 1308 887 796 717 1698 1191 1190 1188 1187 1186 1185 1184 1180 1176 1111

[[4]] 67 101 1834] 66 92 1817 1814 1803 1698 1191 1190 1188 1187 1186 1185 1184 1180 1176 1111

[[5]] 66 92 1817 1814 1803 1191 1190 1188 1187 1186 1185 1184 1180 1176 1111

[[6]] 61 83 1698 1191 1190 1188 1187 1186 1185 1184 1180 1176 1111

[[7]] 49 79 1413] 66 92 1817 1814 1803 66 92 1817 1814 1803 1191 1190 1188 1187 1186 1185 1184 1180 1176 1111

Fig. 5. Clique percolation different size of community.

B. Results of Optimized Clique Percolation

OCPM overlapping algorithm found 180 communities in the network. Seventy six communities are having the large number of nodes accounting to 120 to 220 sizes of the nodes in the community of the network. Sixty three communities are having the medium number of nodes ranging from 110 to 190 in the community of the network. Forty one communities are having the small number of nodes and community sizes are 70 to 100 in the network. CPM algorithm found 180 dense communities in the network. So this network has large numbers of nodes sharing the link for each node. Fig. 6 shows the cricket player's OCPM overlapping community network.

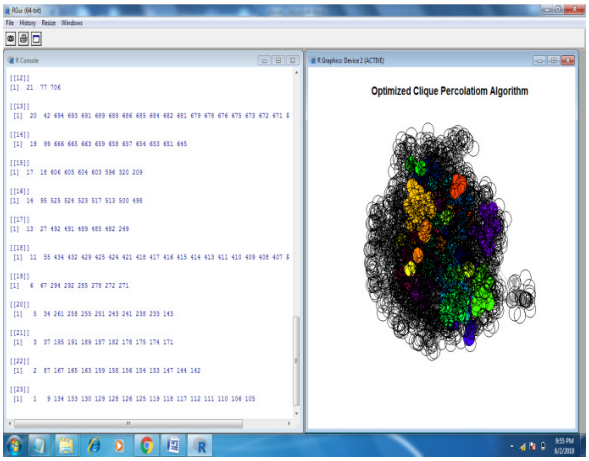


Fig. 6. OCPM community size and network.

OCPM algorithm found 180 different sizes of communities. Seventy six communities have large number of nodes in the cricket player network. These communities are dense and share more information between each node. Sixty three communities have medium number of node in the network. Incoming and out coming nodes are interactive between each community. Forty one communities are having smaller number nodes in the network. It is concluded that this community has less interaction between nodes on the network. Fig. 7 shows clique percolation overlapping communities.

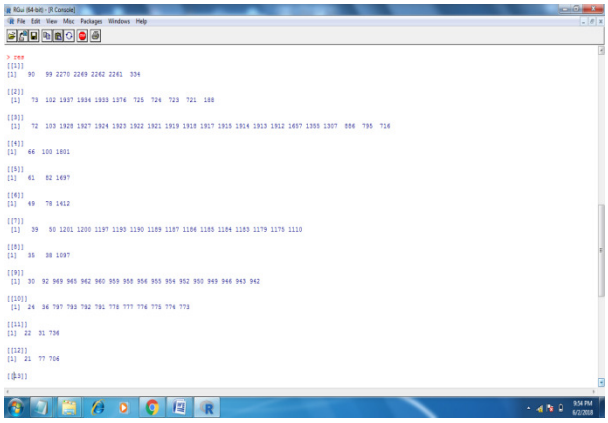


Fig. 7. OCPM different size of community.

C. Results of Parallel Clique Percolation

PCPM overlapping algorithm discovered 170 communities in the network of which 74 communities have huge quantity of nodes that is 150 to 230 nodes in the neighborhood of the network. Fifty three communities are having the medium number of nodes accounting to 80 to 100 in the community of the network. 43 communities are having the small number of nodes and community sizes ranging from 40 to 80 in the network. CPM algorithm found 196 dense communities in the network. So this network has large numbers of nodes sharing the link for each node. Fig. 8 shows the cricket player's PCPM overlapping community network.

Seventy four communities found by PCPM have large number of nodes in the cricket player's network that are dense and share more information between each node. Fifty three communities have medium number of nodes in the network that are interactive between each community. Forty three communities are having smaller number nodes in the network. Fig. 9 shows Parallel Optimized clique percolation overlapping communities.

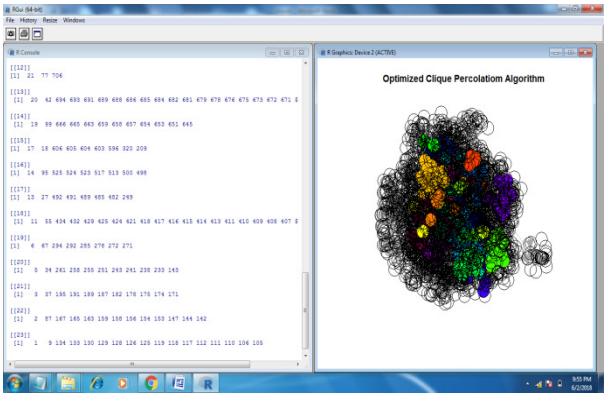


Fig. 8. PCPM community size and network.

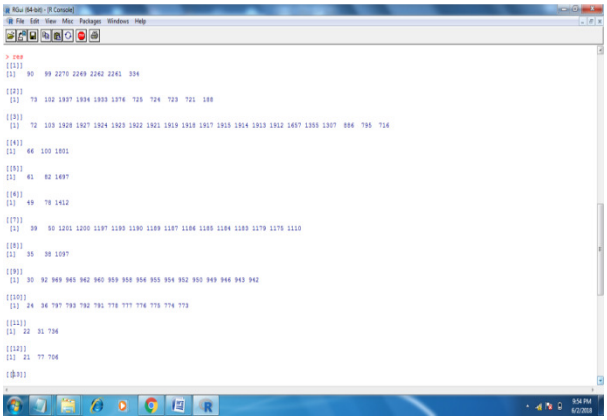


Fig. 9. PCPM different size of community.

D. Comparison of three Clique Percolation Methods

CPM algorithm discovered 198 communities and different size of the node in the graph. OCPM found 180 communities and number of nodes are the dense overlapping community in the network. PCPM exposed 170 communities in the network. It shows better performance than other methods because every overlapping community has large number of nodes in the network. Fig. 10 shows three different sizes of overlapping community detection in the cricket player's network.

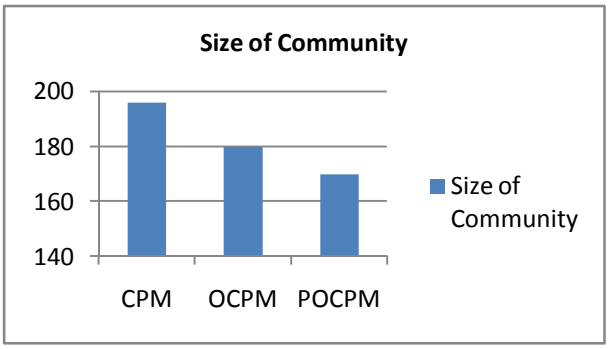


Fig.10. Different size of Community detection.

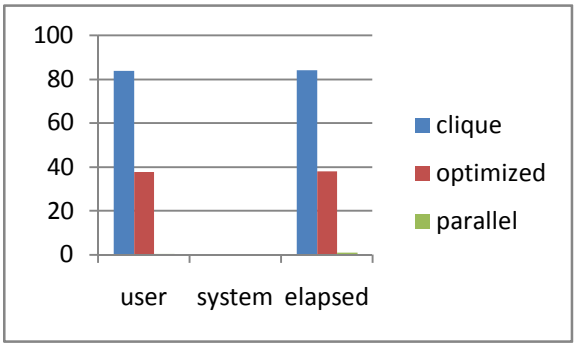


Fig. 11. System elapsed Time.

Table 1: System elapsed time value.

Algorithm	User (seconds)	System (seconds)	Elapsed (seconds)
CPM	83.76	0.15	84.05
OCPM	37.89	0.26	38.11
PCPM	0.29	0.05	0.83

Table 1 shows user, system and elapsed time for clique percolation method. In this process, CPM user, system and elapsed time took 83.76, 0.15 and 84.05 respectively. OCPM algorithm's user, system and elapsed time taken 37.89, 0.26 and 38.11. Parallel CPM is superior performance of other two Techniques. It takes less computation time for user, system and elapsed time in the network. Fig. 11 shows time duration for three different type of Clique Percolation implementation system.

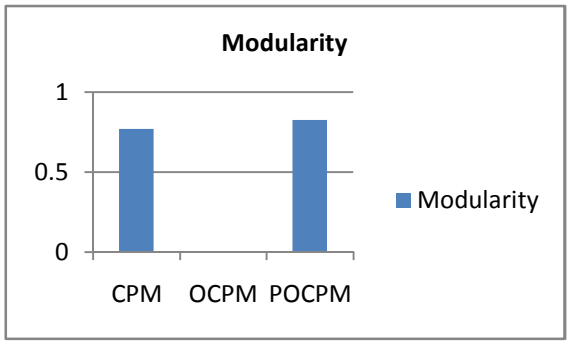


Fig. 12. Modularity Score.

The methods are CPM, OCPM, POCPM, achieved modularity score values of 0.7725246, 0.7865626 and 0.8467654 respectively. POCPM scores better than the other two methods. It has large modularity score than two algorithms. Hence it is more interactive between nodes in the network. Fig. 12 shows modularity score for three different clique percolation methods.

In summary, CPM algorithm discovered more number of communities when compared to OCPM and PCPM. The modularity score of communities detected by PCPM is larger than, that discovered by OCPM and PCPM. The computational time of PCPM algorithm is comparably less than other algorithms CPM and OCPM.

V. DISCUSSION

Three algorithms have been applied for finding overlapping communities in the sports person network in which CPM algorithm discovered more number of communities than OCPM and PCPM. CPM overlapping algorithm discovered 198 communities in the network. OCPM algorithm found 180 different sizes of communities. PCPM algorithm discovered 170 communities and different size of the node in the graph. PCPM yielded better performance compared to other two algorithms as it has used a clique percolation algorithm for detecting clique communities in a network by inserting its edges and keeping track of the emerging community structure. This algorithm applied on twitter networks, has shown that the computational time required to process a network scales linearly with the number of k-cliques in the network. The modularity score of communities detected by PCPM is larger than, that discovered by OCPM and PCPM. The computational time of PCPM algorithm is comparably less than other algorithms CPM and OCPM.

VI. CONCLUSION

In this research work, three algorithms have been applied to finding overlapping communities in the sports person network. CPM overlapping technique found 198 groups in the network. OCPM algorithm found 180 different sizes of communities. PCPM algorithm discovered 170 communities and different size of the node in the graph. Finally, PCPM yields better performance compared to the other two algorithms. It has used a clique percolation algorithm for detecting clique communities in a network by inserting its edges and keeping track of the emerging community structure. This algorithm has specifically been designed for dense networks, where verities and edge based on the links or the cliques formed by them are necessary for obtaining meaningful information on the structure. By applying the algorithm on twitter networks, it is shown that the computational time required to process a network scales linearly with the number of k-cliques in the network.

VII. FUTURE SCOPE

In this work the clique percolation algorithm was utilized based on direct network. This research work can be implemented with different types of overlapping algorithms and also with indirect network.

REFERENCES

[1]. Baumes, J., Goldberg, M., & Magdon-Ismail, M. (2005). Efficient identification of overlapping communities. In *International Conference on Intelligence and Security Informatics* (pp. 27-36). Springer, Berlin, Heidelberg.

[2]. Clauset, A., Newman, M. E., & Moore, C. (2004). Finding community structure in very large networks. *Physical review E*, *70*(6), 1-6.

[3]. Du, H., Feldman, M. W., Li, S., & Jin, X. (2007). An algorithm for detecting community structure of social networks based on prior knowledge and modularity. *Complexity*, *12*(3), 53-60.

[4]. Du, N., Wu, B., Pei, X., Wang, B., & Xu, L. (2007). Community detection in large-scale social networks. In *Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis* (pp. 16-25). ACM.

[5]. Newman, M. E. J. (2004). Fast algorithm for detecting community structure in networks. *Phys. Rev. E 69*, 066133-1-5.

[6]. Newman, M. E., & Girvan, M. (2004). Finding and evaluating community structure in networks. *Physical review E*, *69*(2), 026113-1-15.
[7]. Derényi, I., Palla, G., & Vicsek, T. (2005). Clique percolation in random networks. *Physical review letters*, *94*(16), 1-4.
[8]. Palla, G., Derényi, I., Farkas, I., & Vicsek, T. (2005). Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, *435*(7043), 814-818.
[9]. Pujol, J. M., Béjar, J., & Delgado, J. (2006). Clustering algorithm for determining community structure in large networks. *Physical Review E*, *74*(1), 1-9.
[10]. Hartigan, J. A., & Wong, M. A. (1979). Algorithm AS 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, *28*(1), 100-108.
[11]. Sadi, S., Ögüdücü, Ş., & Uyar, A. Ş. (2010). An efficient community detection method using parallel clique-finding ants. In *IEEE Congress on Evolutionary Computation* (pp. 1-7). IEEE.
[12]. Wakita, K., & Tsurumi, T. (2007). Finding community structure in a mega-scale social networking service. In *IADIS International Conference* WWW*/Internet 2007* (pp. 153-162).
[13]. Zhang, Y., Wang, J., Wang, Y., & Zhou, L. (2009). Parallel community detection on large networks with propinquity dynamics. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 997-1006). ACM.
[14]. Jonnalagadda, A., & Kuppusamy, L. (2018). Overlapping community detection in social networks using coalitional games. *Knowledge and Information Systems*, *56*(3), 637-661.
[15]. Bai, X., Yang, P., & Shi, X. (2017). An overlapping community detection algorithm based on density peaks. *Neurocomputing*, *226*, 7-15.
[16]. Elyasi, M., Meybodi, M., Rezvanian, A., & Haeri, M. A. (2016). A fast algorithm for overlapping community detection. In *2016 Eighth International conference on information and knowledge technology (IKT)* (pp. 221-226). IEEE.
[17]. Ding, Z., Zhang, X., Sun, D., & Luo, B. (2016). Overlapping community detection based on network decomposition. *Scientific reports*, *6*, 24115, pp 1-15.
[18]. Liu, X., Suo, J., Leung, S. C., Liu, J., & Zeng, X. (2015). The power of time-free tissue P systems: Attacking NP-complete problems. *Neurocomputing*, *159*, 151-156.
[19]. Chen, X., Pérez-Jiménez, M. J., Valencia-Cabrera, L., Wang, B., & Zeng, X. (2016). Computing with viruses. *Theoretical Computer Science*, *623*, 146-159.
[20]. Khomami, M. M. D., Rezvanian, A., & Meybodi, M. R. (2016). Distributed learning automata-based algorithm for community detection in complex networks. *International Journal of Modern Physics B*, *30*(8), 1650042-1-20.
[21]. Barbillon, P., Donnet, S., Lazega, E., & Bar-Hen, A. (2017). Stochastic block models for multiplex networks: an application to a multilevel network of researchers. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, *180*(1), 295-314.

How to cite this article: Sathiyakumari, K. and Vijaya, M.S. (2020). Overlapping Community Structure Detection using Twitter Data. *International Journal on Emerging Technologies*, *11*(1): 101–107.