

K-Clique Based Community Detection using Union Find: KCUF

¹Atul Srivastava, ²Anuradha Pillai, ³Dimple Juneja Gupta

¹ Department of Computer Engineering, Pranveer Singh Institute of Technology, Kanpur, India

²Department of Computer Engineering, YMCA University of Science & Technology, Faridabad, India

³Department of Computer Applications, NIT, Kurukshetra, India

E-mail: ¹atul.nd2@gmail.com, ²anuangra@yahoo.com, ³dimplejunejagupta@gmail.com

ABSTRACT

Community detection has gained enormous popularity in the field of network analysis. Detecting communities in a graph is a complicated graph and it is directly related to cliques. Clique is a closed group of nodes in a graph. Community in a network is also a closed group of nodes but extent of closeness in communities is lesser than that in cliques. Finding clique in a network is NP-hard problem. Finding communities have similar characteristics, therefore community detection requires exponential time. In this paper, an improved k-clique detection algorithm based on union-find structure is proposed. The time efficiency of community discovery in highly overlapped complex network is improved and communities are identified within approximately linear time. The proposed algorithm KCUF uses union-find structure to store identified communities and reduces the number of unnecessary intersection test. The efficiency and effectiveness of the algorithm is investigated on two standard datasets, COND-MAT and Amazon goods network. The experimental results establish that the proposed algorithm is correct and its time efficiency is better than other overlapping community detection algorithms.

Keywords: *Social Graph; Clique, Community Detection; Social Network Analysis;*

1. INTRODUCTION

The idea of digital world doesn't only have made people modernize but with increasing awareness social networking usage has gain its pace. Thus the friendly sites including Facebook, Twitter, LinkedIn not only works for connecting around with the world but has become an integral asset in peoples life. With over 500 million online population the statistics suggests that 68% of the total population uses networking sites for either of any reasons including maintaining friendly links , digital media usage with news and web series gaining importance, or simply adding up their profiles to access banking facilities. Most of the people have now accessibility and their accounts registered online in the form of their social profiles [1]. The social profiles authorised to the social networking sites allows people to connect with their friends, family and with around the world. The groups and communities that share common characteristics, interests and behaviour are formed which allows people to share their experiences and thoughts amongst each other [2]. Understanding social network with this vast and distributed network is extremely hard task. Thus with mining techniques these networks are visualised and represented in the form of a graph where the connecting links amongst the people gives the idea of bonding characteristics between them. These characteristics allow people to form a connected component or a community. The community discovery is just the other name for community detection where people share the common interests and the groups are formed such that people are so strongly connected that their bonding traits are further used to study the trends and uncovering of the patterns in the network is done [3]. Networks are represented and better visualized as graphs $G(V,E)$, where V denotes the actors or the individual nodes in a network and E denotes the connecting links or relationship between the actors.

Community detection has gained its importance over the years with various real world applications such as propagating the information efficiently, targeted marketing, and detecting the symptoms to control the infections.

1. Information is propagated in a manner that people having the similar interests share their experiences and spread them amongst people adding up to form a large community.
2. Targeted marketing addresses people on the particular product or service so as to increase the core audiences and profit rates incrementing the marketing value.
3. The symptoms of the infections are verified and the group infected with the same are classified into the same category where the treatment can be done properly and the control measures can work efficiently in studying the future remedies.

The epidemic study of the patterns reduces the risk factor and prevention measures can be taken. These applications are examined using methods described by certain researchers. Some of the community detection methods are Louvain algorithm where communities of people sharing common traits are extracted from large networks [4], infomap algorithm where mapping the communities is based upon the map equation determination and the framework examines the communities from large network [5], the assigning of the label to the unlabeled data set in a network is the label propagation method [6], Newman's leading eigenvector method examines the densely connected community by finding the eigenvector and depicting the modularity in the network between nodes. These

method study the relationship links closely enough to know the bonding strengths and also manages to illustrate the ties based on friendship, follower and following links. These methods study the structure statically where one to one communication links are examined. One such example can be seen on the twitter network where links are formed when the friendship or follower relation is mutually accepted as these links would exist for a large time or lifetime. Fig 1(a) depicts that the dataset extracted doesn't rely on the follower links as there is no interaction based on these ties as the interaction between the people is based on tweets where even if the followership is not mutual one can interact on the posts. From the studies conducted the stats suggest that over 70% people does not depend upon the followership links to communicate with leads to a conclusion that a small number of people actually follow the links that are mutually two sided followership [8]-[10]. The cohesiveness accuracy in such communities could be improved by ensuring that the relationship links carry meaningful information and develop stronger bonds of relations with assigning the weights that enhances the temporal communication measure. One such way to improve the cohesiveness accuracy is to shift the focus from defining the static structures to the dynamic structure where large coverage of the network makes the identification of the communities meaningful. The dynamic network structure represents the edges linked if the communication occurs in the time interval t Fig 1(b). The edges linked in a network are measured depending upon the frequency of interactions held which has been overlooked by the researchers until now and is an important factor to be considered so as to better understand the activities occurrences between the edges which could speed up the propagation of the information communicated via these links. Similarly, targeted marketing strategies and should address in a way to convey and bring up the core audiences for any specific product or services. The target audiences are the people whose frequency measure of interaction on a particular product is based on similar liking and interest. This brought up to consider the fact that the dynamic network studies on the weighted networks gives the idea of the behavior of the attached users including their interests, likings and also predicts the interaction process based on the activity mechanism leaving behind its static characteristics which are also of utmost importance and would be efficient to be considered for future references and researches. The studies conducted so far on the community detection methods considers the input to be an unweighted graph [4], [5], [11], [12], and generally has its main focus on determining the cohesive measure in a structure rather than considering the frequency of interactions amongst users which with time measure would definitely affect the behavioral changes in the community. In addition, the standard community detection or search methods concentrate on finding the large communities if tends to add up the outliers or the singleton users in the communities. Thus addressing to this issue recent researches have been studied. Zheng et al. [13] manages to detect only the relevant community by considering the attributes of the individual nodes and query nodes. Li et al. [14] determines the top-k influential communities that have its influence value over the network too high. Although these methods are determined on the weighted graphs finding the

relevant communities but their weight attribute do not depict the activities of the users.

In this paper, a novel algorithm to find communities based on k-clique is presented. The algorithm works on the principle that adjacent k-cliques will fall into same community. Major challenge in doing so is how to decide that two k-cliques are adjacent. The idea is to find common nodes between these cliques, which requires costly intersection operation to be carried out between cliques. By the introduction of an efficient data structure Union Find, the algorithm optimizes number of intersection tests. This results in significant improvement in time.

2. RELATED WORK

Community detection methods are categorized in two ways: first covering the disjoint communities and second considering the overlapping communities. Disjoint community detection methods includes approaches such as Louvain [4], infomap [5], label propagation [6], Newman's leading Eigenvector mechanism [7] and fast greedy approach [18]. Communities' measures are given by the metric results including modularity [19] and the conductance [20]. In the disjoint community detection a node in a graph is taken and joined to form a community based on its attachment to the neighboring nodes so that any node could belong to exactly one community. While in an overlapping community a node can be a part of any number of communities depending on the structure of the community based on their interests and behavior [21]. The clique percolation method (CPM) [11], is a community detection approach covering the overlapping communities. The method finds the cliques of k-size where clique is the complete sub graph. Let us consider a clique of size = 3 is found from a network thus the clique of 3 size is represented as a triangle and if in a network there are similar cliques which share the common vertex adjacent to the other clique could be joined together to form a community.

Overlapping communities can be depicted in the real world as a single person belongs to many people around them in the form of family, friends, colleagues, well-wishers etc. Thus recent research focuses on proposing methods which covers the overlapping communities. Wen et al. [22] has proposed an evolutionary algorithm whose objective is to find the maximal cliques incorporating to an overlapping community. Our proposed work is based on the CPMw for weighted graphs [15], with the fact that we didn't consider the CPMw threshold metric but instead determine the influence propagation model and the density metrics for weighted graphs so as to detect the active communities from the network. Moreover, the time interval and the interaction aspects such as @ and RT are considered and focus is to cover the complete network rather than only followership links. Thus, people actual interests and intentions are studied and evaluated based on the hashtags used in the tweets.

There are three components which together incorporates the local community detection mechanism. The first is to model the real world problem in either a binary or a weighted graph network; the second includes the metrics measure characterization for efficiently featuring the community; and the third is to design an algorithm for finding the communities that measures excellent metrics results. According to Cui et al. [23], such methods are said to be community search with maximality constraint, which are different to the community search with threshold constraint. For binary networks, edges are either connected or not, Wu et al. [24] proposed three types of goodness metrics. The first assembles the set of nodes to form a community if they are densely connected to each other. The metrics that are calculated in this category are classic density [25], edge surplus [26], and minimum degree [27]. In the second category the internal density and the external sparseness both the attributes are optimized. Metric measures in this category are subgraph modularity [28], density isolation [29], and external conductance [30]. The third category considers the nodes on the boundary of the community to be strongly connected with nodes inside the community and loosely coupled with the nodes outside the community. Local modularity [31] which measures the sharpness of the boundary is optimized in this case. Having considering the fact that even though these measures are best fit but some of these measures might include irrelevant subgraphs in the local community that contains the query node which is termed as free rider effect. Wu et al. [24] proposed a query - biased goodness metric so that the free riders could be avoided. Mostly the local community detection methods considers the weighted networks where the goodness metrics are modified while considering binary networks to evaluate the edge weights as well. For example, let us consider a subgraph $C = (V, E, W)$ with k nodes then a density metric which measures the internal connectedness of a subgraph can be defined as the ratio of the total edge weights to the number of vertices in a subgraph.

$$D = \frac{\sum_{u,v \in V} w(u,v)}{|V|} \quad (1)$$

Where $w(u, v)$ is the weight between node u and v and $|V|$ is the cardinality of V .

3. PROPOSED WORK

K-Clique is a complete graph of k nodes. A large graph contains many k -cliques where k ranges from 3 to a large number. $k=2$ does not make any sense as it has only one edge. Two k -cliques are said to be adjacent if they share $(k-1)$ nodes. Reachability if a k -clique from any other k -clique depends on the availability of adjacent k -cliques between them. A community in a graph is set of reachable k -cliques. If value of k is smaller, then a community can be divided into several k -cliques. Similarly if k is large then there will be lesser k -cliques in a community.

Intersection test is carried out to determine whether two cliques are adjacent to each other or not. If a clique has more than k nodes then it must contain several k -cliques which will fall into same community. It is not necessary to divide the clique

into smaller k -cliques. Intersection test is conducted to decide which of these maximal cliques are in the same community.

In a large network there are several overlapping structures. Many maximal cliques are already there and the nodes which are not part of maximal clique can form cliques within themselves. Some cliques are connected, which means they share atleast one node. To decide communities to which these cliques will belong requires intersection test to be conducted between each pair of maximal cliques found in the graph. This is a lengthy process and incurs huge amount of time. This can be optimized by having intersection test conducted among only those maximal cliques who are connected (at least one node is common). Following data structures are maintained to accomplish this:

1- Node to Clique Mapping: A set of cliques is maintained corresponding to each node u such that u is part of every clique in this set. This helps in finding the connected cliques quickly.

2- Clique to Clique Mapping: A set of cliques is maintained corresponding to a particular clique such that every clique in the set is adjacent to mapped clique. Node to clique mapping helps to maintain this data structure. Every maximal clique is assigned a `clique_id`. Connected cliques are intersected in increasing order of `clique_id`, which eliminates the repeated intersections between same maximal cliques.

3-Union Find: Union find is a tree data structure which generally incorporates two operations Union and Find. Union operation merges two different union find trees. While merging the trees it must be ensured that these trees are different trees and not subtrees of the same tree. This is ensured by find ancestors of both of the trees using Find operation. If they have atleast one of their ancestors common then they are already part of the same tree and there is no need to merge these. Union find is used to decide whether two cliques are part of the same community. Building Union Find data structure with n elements requires $O(n)$ time and to find query is run in $O(1)$ time.

In k -clique community detection the key point to improve time is to reduce the intersect tests. The nodes in a maximal clique are stored in increasing order of their IDs. Therefore binary search can be applied while comparing common nodes between two cliques. But this arrangement does not put significant impact on time as the number of nodes in the clique is very less as compared to total number on nodes in the graph. However the number of maximal cliques is very large in the graph and comparison conducted among the cliques is major contribution in total time taken. The major challenge is how to find cliques adjacent to each other quickly. Node to clique and clique to clique mappings are used to accomplish this challenge. This paper presents an effective algorithm based on k -cliques for community detection, which is scalable up to very large graph. The algorithm is divided into two phases: finding the maximal clique and discover the communities.

A. Finding maximal cliques: This phase is responsible for finding all maximal cliques of the graph in such a way that no maximal clique is sub graph of any other maximal clique. Maximal Clique Finder (MCF) algorithm presented by Srivastava et al. [16] is used to find all maximal cliques in the graph whose size is atleast k. Finding cliques having more than k nodes reduces the total number of cliques for intersection test because several k-cliques will be part of these maximal cliques. MCF [16] is an efficient algorithm to find maximal clique as it uses four different pruning techniques and runs very fast. An arbitrary graph of n nodes may contain up to $n^{n/3}$ maximal cliques [31].

B. Community Detection: After finding all maximal cliques in the phase one, this phase starts community detection. First node to clique is prepared for all the maximal cliques discovered in previous phase. Node to clique mapping is used to build clique to clique mapping. And then, according to the clique-clique mapping, targeted comparing test can be made. Next, Unionfind structure is used to store the divided communities, so that we can find out all reachable sets in almost time. Assume there are n maximal cliques and consider n maximal cliques as n independent communities. Create an array community_array with n elements and convert it into Union-Find structure, at the same time, set the corresponding subscript to the value of corresponding element. Then start traversing from first clique by making use of clique-clique mapping. Operation Find is used to determine whether two cliques are in the same community or not. Operation Find is used to identify the two ancestor nodes in the two cliques respectively, if ancestors are same, these two cliques are reachable and next clique is traversed directly.

For example, the clique 1 has already been determined to be adjacent to clique 2 and 3, so clique 2 and clique 3 are reachable. Now there is no need of intersection tests to be conducted between these two cliques. Number of intersection tests is reduced significantly. Otherwise two maximal cliques are compared to determine if they have k-1 or more nodes common. If two maximal cliques have more than k-1 nodes common then they are merged into one community by Union operation. Community_array is updated and size of the communities found so far is also updated respectively. Repeat until all maximal cliques are traversed, the community_array records all discovered communities.

As per the description of algorithm given above, for those k-cliques which are contained in a same maximal clique, the intersection tests are not needed, since they must be in a same community, namely they are reachable. Hence, the number of intersection tests can be greatly reduced in the communities with a lot of overlapping structure. Additionally maximal cliques are stored in order of cliques numbers, and the nodes within maximal cliques are stored sequentially, so while conducting intersection tests, binary search can be used to speed up query time, especially when the number of nodes within the maximal clique is very large. This method is

very effective. Besides, using the improved Union-Find structure and the corresponding algorithms, all reachable maximal cliques can be found in approximately linear time. In this case that there are not many nodes, the cost of traversing all nodes within a maximum clique is low. And when traversing all the maximum cliques, the Union-Find structure is being updated at the same time. So in such a network, all communities can be discovered in nearly linear time.

```

G(V,E) : Input Graph
Max_Cliques : Set of maximal cliques found in G(V,E)
Community_Array : Map for Union_Find
Node_Map(u) : Set of Cliques having node u
Clique_Map(C) : Clique to clique map
KCUF(G)
  Max_Clique = MCF(G);
  for each maximal clique C ∈ Max_Clique
    for each node u ∈ C
      update(Node_Map, u, C);
    end for
  end for
  for each node u ∈ Node_Map
    for each maximal clique C ∈ Node_Map(u)
      update(Clique_Map, C, Node_Map(u));
    end for
  end for
  BuildUnionFind(Community_Array, Max_Cliques);
  while(Max_Clique ! Empty)
    Ci = Extract_Min(Max_Clique);
    for each maximal clique Cj ∈ Clique_Map(Ci)
      if(FindSet(Ci) != FindSet(Cj))
        Union(Community_Array(Ci), Community_Array(Cj));
      end if
    end for
  end while
  return Community_Array;

```

Fig. 1 K-clique based community detection algorithm: KCUF

4. EXPERIMENTS AND RESULT ANALYSIS

The algorithm is tested for its efficiency and time complexity. The proposed algorithm is tested on the real datasets. In order to reflect the reality, two baseline data sets which are very typical and used often in the real network are selected: Cond-MAT dataset and Amazon product online joint purchasing dataset. In terms of checking effectiveness, the scale of community detection results and the number of communities can be mainly considered comprehensively. For the cost of time, proposed algorithm based on k-clique using union find (KCUF) is compared with the existing classic algorithm, algorithm proposed by Kumpula et al. [32] (KUM) and algorithm proposed by Reid et al.[33] (REID). Due to the efficiency of the algorithm is associated with the value of k, in this paper the experiment will be taken under the different condition for the value k to check the performance of each algorithm.

A. COND-MAT DATASET

Cond-MAT dataset is from e-print arXiv, this library was built in Aug,1991 in American Los Alamos National Laboratory. This dataset network has 30561 nodes and 125959 edges. Through experiment by the three different overlapping community partition algorithms, the number of communities under different k can be recognized for every algorithm. Table 1 shows communities detected by all three algorithms under different values of k, it's obvious that the three algorithms can recognize the same number of communities. This indicates effectiveness of our algorithm KCUF as existing typical algorithms; they can all recognize the overlapping community and has the same accuracy. The efficiency of the algorithm we proposed can be checked by the scale of the community and the number of communities. Figure 2 shows distribution of scales of the communities for k = 4. According to Figure 2, it is obvious that most scales of community structures obey uniform distribution, and the scale for a small part of community structures is relatively abnormal, this shows that our algorithm is effective and reasonable.

TABLE 1 NUMBER OF COMMUNITIES IN COND-MAT FOR DIFFERENT K

k	KUM	REID	KCUF
3	3559	3559	3559
4	4309	4309	4309
5	3309	3309	3309
6	2439	2439	2439
7	1705	1705	1705
8	1121	1121	1121
9	682	682	682
10	390	390	390
11	168	168	168

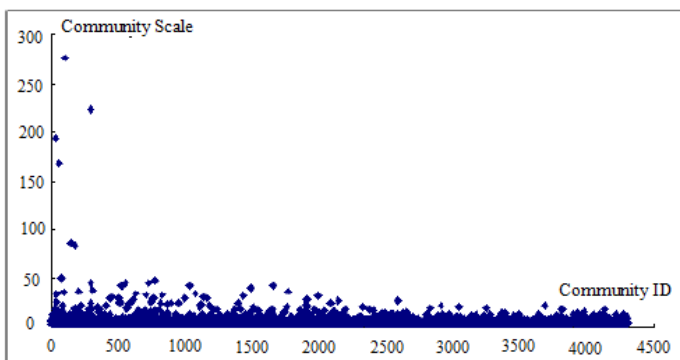


Fig. 2 Community distribution in COND-MAT for k=4

Figure 3 shows time taken by all three algorithms, KUM algorithm, REID algorithms and KCUF algorithm for different values of k.

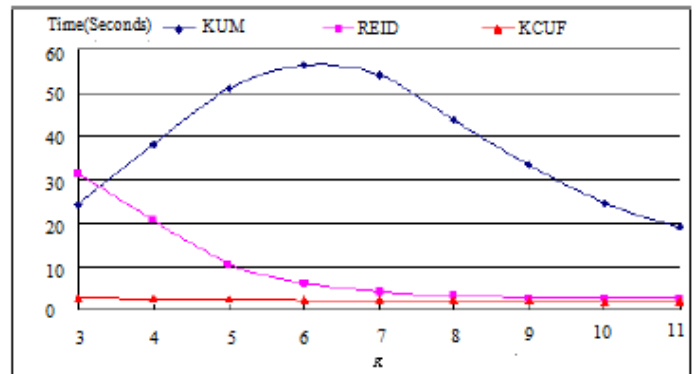


Fig. 3 Time taken to detect communities in COND-MAT for different values of k

As shown in Figure 3, it's apparent that our algorithm (KCUF algorithm) under different values of k, costs less time as compared to the time taken by KUM algorithm and REID algorithm for different values of k. When k = 3, KUM algorithm uses less time than REID algorithm, but for higher values of k time consumption of KUM algorithm increases as compared to REID algorithm. It is evident that KCUF algorithm performs better than both of these algorithms for every value of k.

B. AMAZON PRODUCT ONLINE JOINT PURCHASING DATASET

Amazon dataset is based on the sales relations according to the online orders. If a consumer buys two different products at the same time on Amazon, the two goods will build a connection. When the product p is often purchased with product q, an edge between p and q is put in the graph of goods. The dataset is gathered by the crawlers in Amazon.com, and is downloaded from SNAP. It has 334,863 nodes, 925,872 edges. Table 2 shows results of community detection by KUM algorithm, REID algorithm and KCUF algorithm for different values of k. The consequence is the same as the experiment 1, although the scale of network become larger, also the interior structure become more complicated, all three algorithms can recognized the same number of communities, Which indicates our algorithm is accurate for both simple networks as well as complicated networks

TABLE 2 NUMBER OF COMMUNITIES IN AMAZON GOODS GRAPH FOR DIFFERENT K

k	KUM	REID	KCUF
4	23134	23134	23134
5	10942	10942	10942
6	2621	2621	2621
7	30	30	30

Figure 4 shows distribution of scale of the communities of k=6 detected by KCUF algorithm. This distribution verifies rationality of community detection. Here also the results are similar to that, obtained in COND-MAT dataset. Most scales of community structures which KCUF algorithm recognizes obey

uniform distribution, and the scale for a small part of community structures are relatively abnormal; this indicates that our algorithm is effective and reasonable.

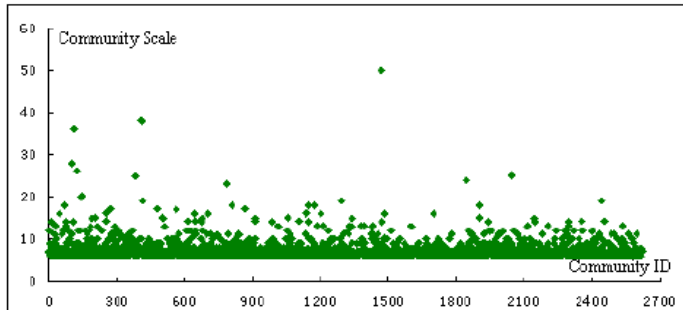


Fig. 4 Community distribution in Amazon for k=6

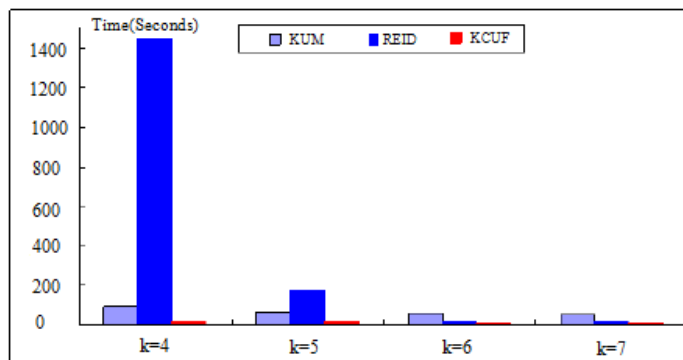


Fig. 5 Time taken to detect communities in Amazon for different values of k

Figure 5 shows time consumption of each algorithm. KCUF algorithm always spends the least time with any value of k. When k=4, the time consumption of REID algorithm doing one time partition is 1451.63 sec, while KUM algorithm takes 90.12 sec, but our algorithm KCUF takes 19.073 sec. In addition, KUM is faster than REID till k = 5. After k = 5, REID beats KUM but KCUF is best for any value of k.

5. CONCLUSION

Social network analysis has major application of community detection in social graphs as a powerful tool for network analysis. Generally most of the social graphs are very large and complex. Community detection in large and complex graphs incurs huge time complexity. The social graphs have overlapping structures. The proposed algorithm KCUF is based on the k-clique and is suitable for finding overlapping communities. This algorithm uses union find to reduce time required for merging two communities and reduces number of intersection test to be conducted among maximal cliques. This implies significant improvement in time. The algorithm is tested on two standard data sets COND-MAT and Amazon Goods network. The results obtained by KCUF are correct and rationale. KCUF algorithm also beats two standard clique percolation based algorithms KUM [32] and REID [33] in for any value of k. Amazon goods network is a social network of

items sold on amazon. This proves the prominent applicability of KCUF on social graphs.

Most of the significant social networks are dynamic in nature, which means the association among nodes may change over time. This perspective of dynamism in social networks can be seen as future aspect for KCUF algorithm.

REFERENCES

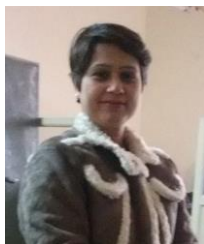
- [1] Sensis. (Jun. 2016). Sensis Social Media Report 2016: How Australian People and Businesses are Using Social Media. [Online]. Available: www.sensis.com.au/asset/PDFdirectory/Sensis_Social_Media_Report_2016.PDF
- [2] A. Java, X. Song, T. Finin, and B. Tseng, "Why we twitter: Understanding microblogging usage and communities," in Proc. ACM 9th WebKDD 1st SNA-KDD Workshop Web Mining Soc. Netw. Anal., 2007, pp. 56–65.
- [3] 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, p. 066111, 2004.
- [4] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," J. Stat. Mech., Theory Experim., vol. 2008, no. 10, p. P10008, 2008.
- [5] M. Rosvall and C. T. Bergstrom, "Maps of random walks on complex networks reveal community structure," Proc. Nat. Acad. Sci. USA, vol. 105, no. 2, pp. 1118–1123, 2008.
- [6] U. N. Raghavan, R. Albert, and S. Kumara, "Near linear time algorithm to detect community structures in large-scale networks," Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top., vol. 76, no. 3, p. 036106, 2007.
- [7] M. E. J. Newman, "Finding community structure in networks using the eigenvectors of matrices," Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top., vol. 74, no. 3, p. 036104, 2006.
- [8] H. Chun, H. Kwak, Y.-H. Eom, Y.-Y. Ahn, S. Moon, and H. Jeong, "Comparison of online social relations in volume vs interaction: A case study of cyworld," in Proc. 8th ACM SIGCOMM Conf. Internet Meas., 2008, pp. 57–70.
- [9] C. Wilson, B. Boe, A. Sala, K. P. N. Puttaswamy, and B. Y. Zhao, "User interactions in social networks and their implications," in Proc. 4th ACM Eur. Conf. Comput. Syst., 2009, pp. 205–218.
- [10] K. H. Lim and A. Datta, "An interaction-based approach to detecting highly interactive twitter communities using tweeting links," Web Intell., vol. 14, no. 1, pp. 1–15, 2016.
- [11] G. Palla, I. Derényi, I. Farkas, and T. Vicsek, "Uncovering the overlapping community structure of complex networks in nature and society," Nature, vol. 435, no. 7043, pp. 814–818, 2005.

- [12] S. Fortunato, "Community detection in graphs," *Phys. Rep.*, vol. 486, nos. 3–5, pp. 75–174, 2010.
- [13] D. Zheng, J. Liu, R.-H. Li, C. Aslay, Y.-C. Chen, and X. Huang, "Querying intimate-core groups in weighted graphs," in *Proc. IEEE 11th Int. Conf. Semantic Comput. (ICSC)*, Jan./Feb. 2017, pp. 156–163.
- [14] R.-H. Li, L. Qin, J. X. Yu, and R. Mao, "Influential community search in large networks," *Proceedings VLDB Endowment*, vol. 8, no. 5, pp. 509–520, 2015.
- [15] I. Farkas, D. Ábel, G. Palla, and T. Vicsek, "Weighted network modules," *New J. Phys.*, vol. 9, no. 6, p. 180, 2007.
- [16] Atul Srivastava, Anuradha, Dimple Juneja Gupta, "Maximum Clique Finder : MCF", *International Journal of Information Technology and Electrical Engineering*, April, 2018, Volume 2, Issue 7, pp. 18-25 .
- [17] N. Alduaiji, J. Li, A. Datta, X. Lu, and W. Liu, "Temporal interaction biased community detection in social networks," in *Proc. 12th Int. Conf. (ADMA)*, Gold Coast, QLD, Australia, Dec. 2016, pp. 406–419.
- [18] T. Aynaud and J.-L. Guillaume, "Static community detection algorithms for evolving networks," in *Proc. IEEE 8th Int. Symp. Modeling Optim. Mobile, Ad Hoc Wireless Netw. (WiOpt)*, May/June 2010, pp. 513–519.
- [19] M. E. J. Newman, "Modularity and community structure in networks," *Proc. Nat. Acad. Sci. USA*, vol. 103, no. 23, pp. 8577–8582, 2006.
- [20] J. Leskovec, K. J. Lang, and M. Mahoney, "Empirical comparison of algorithms for network community detection," in *Proc. ACM 19th Int. Conf. World Wide Web*, 2010, pp. 631–640.
- [21] J. Xie, S. Kelley, and B. K. Szymanski, "Overlapping community detection in networks: The state-of-the-art and comparative study," *ACM Comput. Surv.*, vol. 45, no. 4, 2013, Art. no. 43.
- [22] X. Wen et al., "A maximal clique based multiobjective evolutionary algorithm for overlapping community detection," *IEEE Trans. Evol. Comput.*, vol. 21, no. 3, pp. 363–377, Jun. 2017.
- [23] W. Cui, Y. Xiao, H. Wang, and W. Wang, "Local search of communities in large graphs," in *Proc. SIGMOD*, 2014, pp. 991–1002.
- [24] Y. Wu, R. Jin, J. Li, and X. Zhang, "Robust local community detection: On free rider effect and its elimination," *Proceedings VLDB Endowment*, vol. 8, no. 7, pp. 798–809, 2015.
- [25] B. Saha, A. Hoch, S. Khuller, L. Raschid, and X.-N. Zhang, "Dense subgraphs with restrictions and applications to gene annotation graphs," in *Research in Computational Molecular Biology*. Berlin, Germany: Springer, 2010, pp. 456–472.
- [26] C. Tsourakakis, F. Bonchi, A. Gionis, F. Gullo, and M. Tsiarls, "Denser than the densest subgraph: Extracting optimal quasi-cliques with quality guarantees," in *Proc. SIGKDD*, 2013, pp. 104–112.
- [27] M. Sozio and A. Gionis, "The community-search problem and how to plan a successful cocktail party," in *Proc. SIGKDD*, 2010, pp. 939–948.
- [28] F. Luo, J. Z. Wang, and E. Promislow, "Exploring local community structures in large networks," in *Proc. Conf. WI*, 2006, pp. 233–239.
- [29] K. J. Lang and R. Andersen, "Finding dense and isolated submarkets in a sponsored search spending graph," in *Proc. CIKM*, 2007, pp. 613–622.
- [30] R. Andersen, F. Chung, and K. Lang, "Local graph partitioning using pagerank vectors," in *Proc. FOCS*, 2006, pp. 475–486.
- [31] Moon J, W Moser. On cliques in graphs. *Israel Journal of Mathematics*, 1965, 3(1): 23-28.
- [32] SCP J Kumpula, M Kivel , K. Kaski, etc. Sequential algorithm for fast clique percolation. *Physical Review E*, 2008, 78(2).
- [33] Fergal Reid, Aaron McDaid, Neil Hurley. Percolation Computation in Complex Networks. *2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining* , 2012: 274-281

AUTHOR BIOGRAPHIES



Atul Srivastava received the B. Tech. degree in Information Technology from the UP Technical University, Lucknow, India, in 2008, and M.Tech. degree in Information Technology from YMCA University of Science and Technology, Faridabad, India in 2012. He is a research student of YMCA University of Science and Technology, Faridabad, India. He is currently Assistant Professor at Pranveer Singh Institute of Technology, Kanpur, India.



Dr. Anuradha Pillai has received her M. Tech and PhD in Computer Engineering from MD University Rohtak, in the years 2004 and 2011 respectively. She has published 30 research papers in various International journals and conferences. She has more than 13 years of teaching experience. Presently she is serving as Assistant Professor at YMCA University of Science & Technology, A State Govt. University, Faridabad Haryana. Her research interests include Web Mining, Data Structures and Algorithms, Databases.

Research Papers listed in database.



Dr. Dimple Juneja Gupta is working as Faculty at Department of Computer Applications at NIT Kurukshetra, India.