ISSN 2277-2685
IJESR/Oct. 2020/ Vol-10/Issue-4/1-09
Sumit Kumar Gupta *et. al.,* / International Journal of Engineering & Science Research

CBLA: A Clique Based Louvain Algorithm for Detecting Overlapping Community

Sumit Kumar Gupta[a,]
Assoc.Profesor
Dept of CSE

Sumit Kumar Gupta@ujk.in

Abstract

Graph mining is one of the significant tasks in the field of computer science. Most of the applications generate a vast amount of data which is represented with the help of a graph. Due to this graph representation, these applications have become complex and increased in size. Finding relevant information from that graph becomes a complicated task. For this community detection algorithms play a vital role in graph partitioning to retrieve relevant information. Finding communities in a graph reduces the complexity of the graph due to related data comes closer to forming a community. Many algorithms have been introduced in the last decade; the Clique percolation method (CPM) is the benchmark algorithm for finding an overlapping community. But in this method, some nodes remain unclassified, nodes that are not part of the clique. Paper proposed the clique-based Louvain algorithm(CBLA), which can classify the non-classified node (NCN) obtained after finding cliques in one of the communities by applying the Louvain algorithm. Louvain algorithm is used to classify the non-overlapped community, but with the help of cliques, it will also detect the overlapped nodes. This paper compared the proposed algorithm with four other benchmark algorithms. The proposed algorithm gives equal or enhanced performance among all compared algorithms.

Introduction

Community detection algorithms[1] play an essential role in numerous real-life applications like classifying the various research papers into their relevant categories, social networking, data mining for retrieving relevant information using a web crawler, and many more. Community detection uses the unsupervised learning concept for categorizing the graph into pertinent information. Community detection is one where densely connected nodes are categorized into a group called a community, while sparsely connected nodes come across the communities. Community detection is broadly classified into two categories[2, 3]: Non-overlapping community detection, where one node can be present

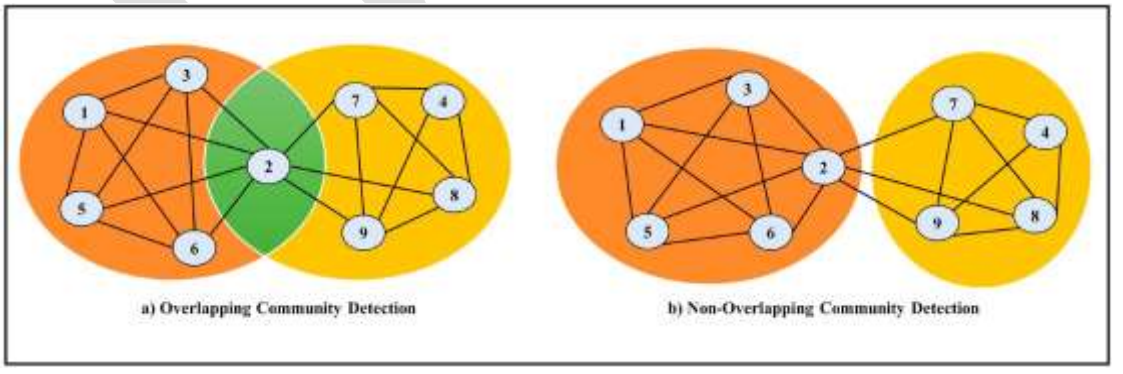


Fig. 1: Computed community after applying overlapping and non-overlapping community detection algorithms.

in at most one community. It is mainly used in medical science, where accurate detection is required like a protein- protein interaction network[4]. Another is the overlapping community detection algorithm[5, 6], where two or more communities can have same node. It is mainly used where one item is related to more than one group, like in Facebook social networking[7, 8] and the recommendation system[9] for finding a user with a similar interest in an e-commerce application[10], where one person may relate to more than one group. Moreover, link prediction[11] helps to find social network users who are likely to become friends. Network compression[12] is another useful application of community detection to reduce and visualize the graph properly.

Figure 1 shows the example to demonstrate overlapping and non-overlapping community. The figure contains 9 nodes and 16 edges, in part (a) Applying an overlapping community detection algorithm got two communities where node 2 present in both communities. While in part (b), Applying a non-overlapping community detection algorithm got two communities where every node is present in only a single community.

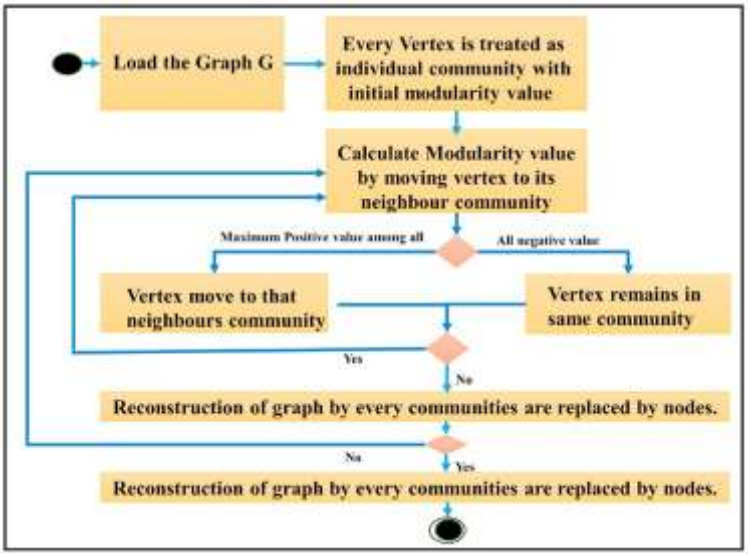
The remaining paper is organized as follows: Section II describes some benchmark community detection algorithms. Section III Gives a detailed description of the proposed CBLA algorithm. Section IV tells the experimental results and analysis. Finally, the conclusion of the paper is discussed in section V.

COMMON COMMUNITY DETECTION ALGORITHM

Lots of state of art community detection algorithms are available in literature to partition the graph into relevant information. Some of the benchmark algorithms are discussed below.

1.1. The Louvain Algorithm

The Louvain algorithm, a quick community discovery method based on modularity matrices, was introduced by Blondel et al[13]. It is developed on an approach known as agglomerative hierarchical clustering. Louvain algorithm is a non-overlapping algorithm where a node can be present in only one community. It consists of two basic steps. Figure2 shows the basic flow of the algorithm. Initially, every node is considered as a community, and initial modularity is calculated. Modularity values can range from -1 to 1. For building a community, every vertex is tried to move in its neighbouring community, which gives maximum modularity value. Then, the vertex moves to that neighbour community, which offers maximum positive modularity value. Suppose all the neighbouring community provides zero or negative value, then that vertex remains in its community. After all, the vertex processed algorithm moves into the second step, known as graph rebuilding. In graph rebuilding, every community is considered a node, the sum ofinter-community edges is put into self-edge, and the sum of intracommunity edges is placed as the weight between two community edges. The same process continues up to the threshold set by the user. It gives the best results in mostof the benchmark data and is best suitable for finding a non-overlapping community.



Corresponding Author

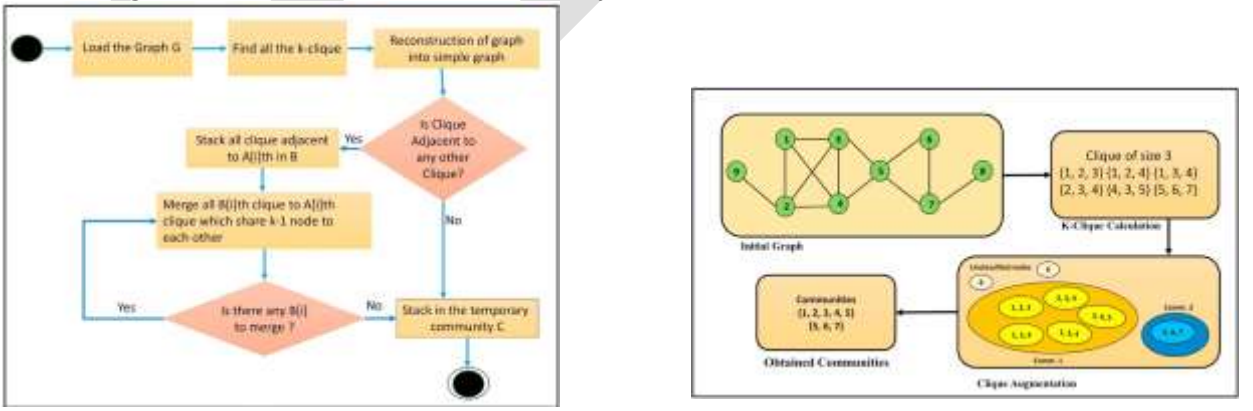
Fig. 2: Computed community after applying overlapping and non-overlapping community detection algorithms.

1.2. Clique Percolation Method

The clique percolation method (CPM)[14, 15] is one of the efficient techniques introduced by Palla et.al. One node may be present in one or more communities, making it the most popular algorithm for identifying overlappingcommunities. It is based on the graph cliques, as the name would imply. Figure 3(a) shows the processing of the algorithm. This algorithm is based on the size of cliques called k. The first step finds all the k-size cliques in the graph.After that cliques merging is carried out based on the property, two cliques can be merged if they share k-1 vertex. Lastly, every merged clique is treated as a community. But the main drawback of the algorithm is the non-classified nodes(NCN) mean nodes that are not part of any cliques remain unclassified. For the classification of those nodes, various algorithms are coming like the PercoMCV algorithm [16], which uses the eigenvector centrality method[17] for the classification of NCN obtained after applying the CPM algorithm. OLCPM [18] is another algorithm that tries to classify NCN obtained after applying the CPM algorithm with the help of a label propagation algorithm[19]. Label propagation algorithm is applied to the temporary communities obtained after the CPM algorithm.
Figure 3(b) shows a basic example of the clique percolation method. In this example graph consists of 9 nodes and 13 edges. After executing the first step, where the value of k is 3, get the six cliques shown in the figure. In the second step, cliques are merged if they share k-1, i.e., 2 vertices common. So, in the second step, 5 cliques are merged because they share 2 vertexes common. Finally, obtain the two communities 1,2,3,4,5 and 5,6,7. Simultaneously two vertices 8 and 9 remain unclassified because neither are part of any community called a non-classified node (NCN), which is the main problem of the CPM method.

1.3. Literature Review

Cliques for Combining Numerous Clustering Algorithm is a unique algorithm developed by Mimaroglu et al. [20]that unifies multiple clusters into a single cluster based on a maximal complete graph (CLICOM). In a graph, the similarity is determined using a co-associated matrix depicted object-wise, while Jaccard similarity is determined and depicted cluster-wise. CLICOM offers object-wise and cluster-wise granularity, which can be used for finer and coarser granularity, respectively.
The influence Maximization Approach Based On Clique(IMC) was a proposed heuristic algorithm by Li et al. [21]. In order to maximize the diffusion of influence throughout the network, a small number of nodes are eliminated from the network using the IMC method, which is used to classify the influence node.



(a) Flow chart of the various steps of the CPM method.

(b) It shows an example of the CPM method.

Fig. 3: Demonstration of the CPM method.

The GVG Mine algorithm was proposed by Lee et al. [22] in order to extract frequent maximal cliques from databases of the larger graph. Additionally, they developed a global view graph (GVG) to summarise the data in the graph database.

By removing the low-weighted edges and dividing the global view graph into many bi-connected components, the GVG Mine method finds the maximal frequent cliques.
The Parallel Enumeration of Cliques Using the Ordering (PECO) algorithm was proposed by Svendsen et al. [23] to maximise clique enumeration (MCE). The MCE algorithm splits the input graph into numerous sub-graphs, which are then each individually examined to count the cliques.
The Clique Removal Heuristic technique was first developed by Schmitt et al. [24]. The Clique percolation ap- proach is practical for finding clusters that overlap. Instead of the k-clique approach, CPM employs a maximal clique enumeration algorithm. The cluster overlap and maximum clique adjacency are determined using the clique removal heuristic using a binary tree. The Clique Removal Heuristic technique was first developed by Schmitt et al. The Clique percolation approach is effective at finding clusters that overlap. Instead of the k-clique approach, CPM employs a maximal clique enumeration algorithm. The cluster overlap and maximum clique adjacency are determined using the clique removal heuristic using a binary tree.
The sequential clique percolation algorithm (SCP) was introduced by Kumpula et al. [25] for quick clique perco- lation in weighted and unweighted graphs. The weighted clique percolation method is applied using SCP. Instead of acquiring k-communities at a single weight threshold value of k as in the maximal clique algorithm, this algorithm sequentially integrates the edges into the network in decreasing order of their weights, allowing for the detection and emergence of k-clique communities at specified multiple weight thresholds values of k, and simultaneously generates a dendrogram representation of the hierarchical structure.
Because the clique percolation approach does not cover the entire graph, Maity et al. citec70 introduced the Ex- tended Clique Percolation Method (ECPM). The author's main goal is to completely cover all nodes in the linked network. The suggested approach ensures complete node coverage for the attached graph, and the modularity of the community structure is used to gauge its quality. Therefore, ECPM provides an accurate community structure when compared to traditional CPM.
Clique percolation clustering was suggested by Zhang et al. [27] to identify overlapping network modules of the protein-protein interaction (PPI) network. The discovery of k-cores, the Markov clustering algorithm, restricted neighbourhood search clustering, and edge-betweenness clustering are a few of the many methods suggested for grouping the network that has been used to analyse PPI networks. Finding functional modules in the PPI network is the major goal of this paper. Use P-value to determine a module's practical characteristics when a function annotation wasn't given to a specified module with a P-value minimum.
Using a weak clique percolation approach(WCPM), Zhang et al.[28], Instead of looking for cliques, only evaluates weak cliques. Even in networks with high levels of overlapping diversity and density, CPM is accurately classified as overlapping communities, but they struggle with high computation costs and complexity. By identifying weak

cliques in the network and combining these weak cliques into communities based on maximum priority and maximum similarity, WCPM lessens this issue.

2. MATERIALS AND METHODS

This paper proposed an efficient overlapping algorithm based on the clique percolation method and Louvain al- gorithm. This algorithm tries to classify all those nodes which remain unclassified in the basic clique percolation method. It is a hybrid approach of the CPM algorithm and Louvain algorithm; that's why called Clique based Louvain Algorithm(CBLA). CBLA consists of two phases; the First phase applies the clique percolation method for finding the all possible k-clique. In the second phase, apply the Louvain algorithm after the graph rebuilding for finding the final community.

Proposed CBLA Algorithm

The proposed algorithm tries to find all possible overlapping communities present in the graph. The clique perco-lation method determines the initial communities based on cliques. After that graph, rebuilding is carried out where every clique is converted into meta nodes. Finally, the Louvain algorithm is applied to the updated graph to find the final communities. As the Louvain algorithm is a non-overlapping algorithm, but two or more than two cliques may contain the same nodes, which can be classified into two different communities. During the finding of the community, the connectivity of the graph is preserved. The working of the CBLA shown in the Algorithm 1. The Framework of the proposed CBLA algorithm is shown in Figure 4(a).

Algorithm 1 :Proposed CBLA Algorithm.

Input: Graphs G(V, E) & Size of clique k.
Output: Vertex V classifies into communities.
Step 1: Find all the k-clique in the given graph G.
Step 2: All clique is converted into a single node with a self-loop of a number of edges in the clique.
Step 3: The graph contains two types of nodes node of clique and non-classified node (NCN) from step 1, and edges remain the same as the original graph.
Step 4: Louvain algorithm is applied to the reconstructed graph.
4.1 : Every vertex is treated as a community, and the initial modularity value is calculated.
4.2 : Maximize the modularity by moving vertex to its neighboring community.
Step 5: Repeat step 4 until all the NCN are classified into communities.

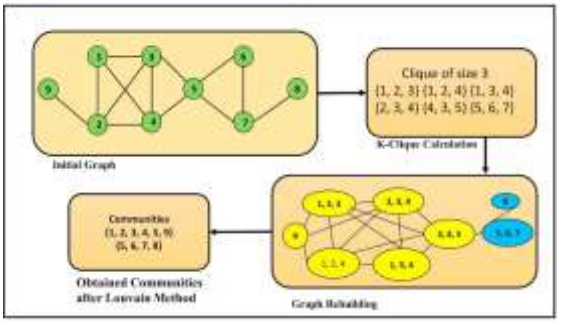
2.1. Complexity of Algorithm

The time complexity of an algorithm exhibits its performance. In our proposed algorithm, the time complexity is the sum of the clique finding process and the execution of the Louvain algorithm. Let n be the number of vertexes in the graph, m be the edges in the graph, and k be the size of cliques.
The time required for finding cliques in the graph = $O(n)$.
The time complexity of applying the Louvain algorithm = $O(m)$.So, the total complexity = $O(n) + O(m)$
But in the newly constructed graph, the number of edges decreases and is approximately equal to n.So, Final complexity is approximately equal to $O(n)$.

Example of Proposed CBLA Algorithm

For better understanding of the CBLA algorithm a example of the small graph is demonstrated in the Figure 4(b).In the example, the graph consists of 9 vertices and 13 edges. After applying the first phase of the proposed CBLA

(a) Shows the various steps flow chart of the CBLA algorithm.



(b) Show step by step process of CBLA method with the help of an example.

Fig. 4: Demonstration of the Proposed CBLA method.

algorithm, it identified the 6 cliques when the value of k is 3. In the next step, the graph rebuilding process is carried out where all the cliques are converted into meta vertices in the new graph, and edges are placed according to the old graph. Finally, the Louvain algorithm is applied to the graph obtained after the second step in the final step. Based on the modularity value Louvain algorithm classified all the non-classified nodes(NCN). After applying the Louvain algorithm, this example got the two communities named as {1, 2, 3, 4, 9} and {5, 6, 7, 8}.

3. RESULTS AND DISCUSSION

The computational environment is built for analyzing the performance of proposed CBLA Algorithm effectively. The i^7 CPU with 8.0 GB RAM, 1TB hard disk and Integrated graphics card with windows 10 is used to perform these experiments. The modules are developed in the python programming language. For the experiment use the anaconda tool, which mainly supports the Python language. To evaluate the performance of proposed CBLA algorithm omega index value and complexity are used as parameter. To test the proposed approach, it use the data from Zachary et al. in [29]. This graph become popular for examining community structure which shows the relationship among the members of the karate club. it consists of the 78 edges which shows the interaction among the members.

3.1. Synthetic network simulation

The main objective of the synthetic network is to check the correctness of the algorithm. In this paper synthetic graph consists of 15 nodes and 25 links; after performing the first step of the algorithm, it finds the 13 clique and 2 unclassified nodes. After the graph reconstruction, the Louvain algorithm is applied to it, which classifies all the 15 nodes into 3 different communities. The evaluated graph is shown in Figure 5. The obtained three different communities are shown with different color. When this graph passed to the CPM algorithm one node remains unclassified. So which clearly tells that proposed algorithm performed well as compared to CPM algorithm.

3.2. Real-world network simulation

Zachary karate club is one of the popular real-world data set used in community detection. Karate club[29] consists of 34 members of the university in the united states having some relationship with each other. After the disputes of two members, the karate club is divided into two groups. Zachary uses the relationship and further divide the club based on their relationship. Most of the algorithm uses this data set because its community structure is known in advance. This

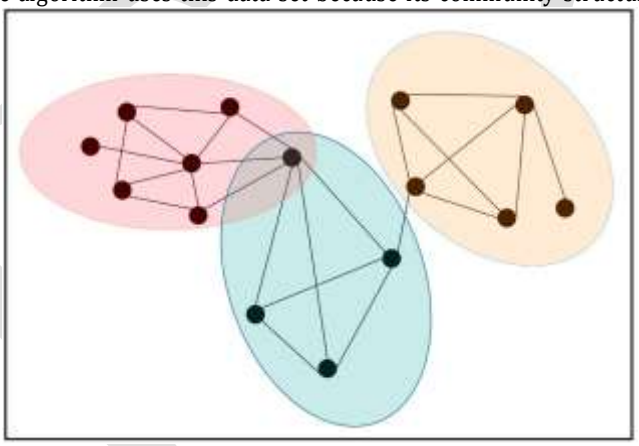


Fig. 5: Shows the Communities obtained after applying CBLA Algorithm on the above mentioned example.

simulation's goal is to evaluate the proposed CBLA algorithm's performance compared to another well-known benchmark algorithm. Table 1 presents the evaluation of this experiment based on the number of communities identified, the algorithm's difficulty, and the omega index's value. The complexity of the suggested method and the algorithms under comparison are nearly identical. The proposed CBLA algorithm outperforms among all the algorithms in term of accuracy. The accuracy of the proposed algorithm is calculated with the help of the omega index value. The omega index value is ranges from 0 to 1, where higher the value means higher accuracy. Figure 6 shows the omega index values obtained on Zachary karate club dataset after applying the proposed CBLA algorithm and other state of art algorithms. It clearly shown that the proposed algorithm has higher omega index value which means the higher accuracy among all compared algorithms.

Table 1: Comparative study between various benchmark algorithms and proposed algorithm.

Compared Algorithms	Communities Detected	Time Complexity	Omega Value
PercoMCV	3	$O(n)$	0.404
CPM	4	$O(n^2 + (h+n)s)$	0.291
Label Propagation	3	$O(n)$	0.372
NEDIOUI M. A	3	$O(n)$	0.127
CFinder	3	$O(n)$	0.095
CBLA	3	$O(n)$	0.491

4. CONCLUSION

This paper propose the efficient and new overlapping algorithm for community detection called a CBLA algorithm. It is a hybrid approach focusing on the Louvain algorithm and the CPM algorithm concepts. The Louvain method is primarily employed in the proposed algorithm to categorise the unclassified nodes produced by the CPM algorithm.The proposed CBLA algorithm is compared with some state of art overlapping community detection algorithms considering three different parameter community detected, Complexity, and omega index value. The proposed CBLA algorithm is outperforms among the all compared algorithm. In future the CBLA algorithm can be implemented in a parallel environment, as finding cliques is an independent task.

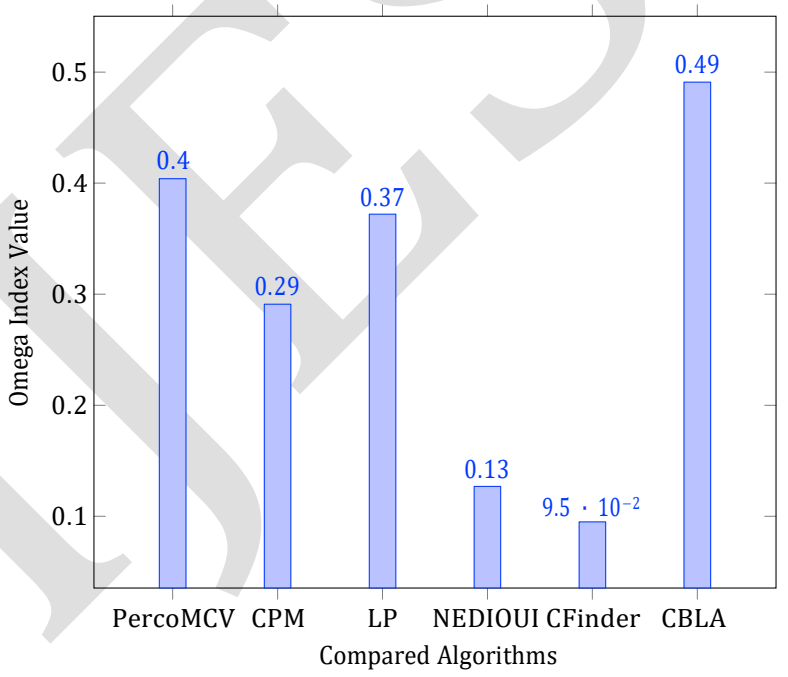


Fig. 6: Omaga index values of different algorithms when applied on Zachary karate club data-set

References

Fortunato, Santo. (2010) "Community detection in graphs." *Physics reports* **486** (3): 75–174.
Gupta, Sumit and Singh, Dhirendra Pratap. (2020) "Recent trends on community detection algorithms: A survey." *Modern Physics Letters B*

34(35): 2050408.
Gupta, Sumit Kumar and Singh, Dhirendra Pratap and Choudhary, Jaytrilok. (2022) "A review of clique-based overlapping community detec-tion algorithms." *Knowledge and Information Systems*: 1–36.
Yu, Zhongtang and Morrison, Mark. (2004) "Improved extraction of PCR-quality community DNA from digesta and fecal samples." *Biotech-niques* **36**(5): 808–812.
Lancichinetti, Andrea and Fortunato, Santo. (2009) "Community detection algorithms: a comparative analysis." *Physical review E* **80**
(5):056117.
Gupta, S.K., Singh, D.P. (2022) "Seed Community Identification Framework for Community Detection over Social Media." *Arabian Journalfor Science and Engineering*: 1–15.
Bedi, Punam and Sharma, Chhavi. (2016) "Community detection in social networks." *Wiley Interdisciplinary Reviews: Data Mining andKnowledge Discovery* **6**(3): 115–135.
Missaoui, Rokia and Sarr, Idrissa. (2015) "Social network analysis-Community detection and evolution." *Springer*.
Gasparetti, Fabio and Sansonetti, Giuseppe and Micarelli, Alessandro. (2021) "Community detection in social recommender systems: a survey."
Applied Intelligence **51**(6): 3975–3995.
Cai, Qing and Ma, Lijia and Gong, Maoguo and Tian, Dayong. (2016) "A survey on network community detection based on evolutionarycomputation." *International Journal of Bio-Inspired Computation* **8**(2): 84–98.
De Bacco, Caterina and Power, Eleanor A and Larremore, Daniel B and Moore, Cristopher. (2017) "Community detection, link prediction, andlayer interdependence in multilayer networks." *Physical Review E* **95**(4): 042317.
Zhao, Xingwang and Liang, Jiye and Wang, Jie. (2021) "A community detection algorithm based on graph compression for large-scale socialnetworks." *Information Sciences* **551**: 358–372.
Blondel, Vincent D and Guillaume, Jean-Loup and Lambiotte, Renaud and Lefebvre, Etienne. (2008) "Fast unfolding of communities in largenetworks." *Journal of statistical mechanics: theory and experiment* **2008** (10): 10008.
Palla, Gergely and Barabási, Albert-László and Vicsek, Tamás. (2007) "Quantifying social group evolution." *Nature* **446** (7136): 664–667.
Palla, Gergely and Derényi, Imre and Farkas, Illés and Vicsek, Tamás. (2005) "Uncovering the overlapping community structure of complex networks in nature and society." *Nature* **435** (7043): 814–818.
Kasoro, Nathanaël and Kasereka, Selain and Mayogha, Elie and Vinh, Ho Tuong and Kinganga, Joël. (2019) "PercoMCV: A hybrid approach of community detection in social networks." *Procedia Computer Science* **151**: 45–52.
Zafarani, Reza and Abbasi, Mohammad Ali and Liu, Huan. (2014) "Social media mining: an introduction." *Cambridge University Press*. Boudebza, Souâad and Cazabet, Rémy and Azouaou, Faiçal and Nouali, Omar. (2018) "OLCPM: An online framework for detecting overlap- ping communities in dynamic social networks." *Computer Communications* **123**: 36–51.
Xie, Jierui and Szymanski, Boleslaw K. (2013) "Labelrank: A stabilized label propagation algorithm for community detection in networks."
IEEE 2nd Network Science Workshop (NSW): 138–143.
Mimaroglu, Selim and Yagci, Murat. (2012) "CLICOM: Cliques for combining multiple clusterings." *Expert Systems With Applications* **39**(2):1889–1901.
Li, Huan and Zhang, Ruisheng and Zhao, Zhili and Yuan, Yongna. (2019) "An efficient influence maximization algorithm based on clique insocial networks." *IEEE Access* **7**: 141083–141093.
Lee, Guanling and Peng, Sheng-Lung and Kuo, Shih-Wei and Chen, Yi-Chun. (2012) "Mining frequent maximal cliques efficiently by global view graph." *9th International Conference on Fuzzy Systems and Knowledge Discovery*: 1362–1366.
Svendsen, Michael and Mukherjee, Arko Provo and Tirthapura, Srikanta. (2015) "Mining maximal cliques from a large graph using mapreduce: Tackling highly uneven subproblem sizes." *Journal of Parallel and distributed computing* **79**: 104–114.
Schmitt, Rafael and Ramos, Pedro and Santiago, Rafael and Lamb, Lu´ıs. (2017) "Novel Clique enumeration heuristic for detecting overlapping clusters." *IEEE Congress on Evolutionary Computation*: 1390–1397.
Kumpula, Jussi M and Kivelä, Mikko and Kaski, Kimmo and Saramäki, Jari. (2008) "Sequential algorithm for fast clique percolation." *Physical review E* **78**(2): 026109.
Maity, Sumana and Rath, Santanu Kumar. (2014) "Extended Clique percolation method to detect overlapping community structure." *Interna- tional Conference on Advances in Computing, Communications and Informatics*: 31–37.

Zhang, Shihua and Ning, Xuemei and Zhang, Xiang-Sun. (2006) "Identification of functional modules in a PPI network by clique percolationclustering." *Computational biology and chemistry* **30**(6): 445–451.
Zhang, Xingyi and Wang, Congtao and Su, Yansen and Pan, Linqiang and Zhang, Hai-Feng. (2017) "A fast overlapping community detection algorithm based on weak cliques for large-scale networks." *IEEE Transactions on Computational Social Systems* **4**(4): 218–230.
Zachary, Wayne W. (1977) "An information flow model for conflict and fission in small groups." *Journal of anthropological research* **33**(4):452–473.