Boosted Constrained K-Means Algorithm for Social Networks Circles Analysis

Intisar M. Iswed[1]*, Yasser F. Hassan[2], Ashraf S. Elsayed[3]
Mathematics and Computer Science Department, Alexandria University, Alexandria, Egypt

***Abstract*—The volume of information generated by a huge number of social networks users is increasing every day. Social networks analysis has gained intensive attention in the data mining research community to identify circles of users depending on the characteristics in the individual profiles or the structure of the network. In this paper, we propose the boosting principle to find the circles of social networks. Constrained k-means clustering method is used as a weak learner with the boosting framework. This method generates a constrained clustering represented by a kernel matrix according to the priorities of the pair-wise constraints. The experimental results show that the proposed algorithm using boosting principle for social network analysis improves the performance of the clustering and outperforms the state-of-the-art.**

***Keywords*—Constrained clustering; boosting; social networks; k-means; kernel matrix**

I. Introduction

Due to the evolution in computer science and Internet, social network (virtual society) is considered a positive change in our society where a huge number of people communicate with each other, exchange information, ideas, news, etc. [1]. Social network is a social structure of individuals called “nodes” who are connected by one or many specific kinds of inter connection such as common interest, kinship, friendship, knowledge and relationships of beliefs [2]. Social network analysis has gained intensive attention in the data mining research community to identify the groups (circles) of the individuals depending on the characteristics in the individual profiles or the structure of the network (relationship between individuals). The problem of community detection in social network has been studied from three perspectives:

- Graph-based computing [3] and Graph-partitioning [4, 5, 6] which are based on the information extracted from the structure of the network.
- Machine learning principle which is based on supervised and unsupervised clustering methods that are related to the existence of labelled and unlabeled database respectively. Some clustering methods are k-mean algorithm [7], k-medoids method [8], Expectation Maximization algorithm [9] and kernel k-mean algorithm [10, 11].
- Computational Intelligence which uses bio-inspired concept in complex environments. Some algorithms based on this principle are ant colony optimization [6, 12], Genetic algorithms [13] and Iterated Greedy algorithms [14, 15].

The amount of information generated by huge number of social networks users is rapidly increasing. Consequently, this makes the analysis of social network difficult. Therefore, the researchers focused their works on Ego network. The ego network has one individual (called 'Ego') centering the network and all other individuals (called 'Alters') are connected to this Ego. Fig. 1 illustrates an example of social network that has 10 individuals, where the black node is the ego network. Social networks clustering which is named unsupervised learning, is improved by side information that are called constrained data clustering (semi-supervised clustering) that uses the pre-given knowledge (ground-truth) about the data pairs for enhancing the clustering accuracy. The two main techniques for semi-supervised clustering are the constraint-based technique [16, 17] and the distance metric learning technique [18]. The first technique supposes that data pairs of must-link constraint belong to the same cluster and data pairs of cannot-link constraint belong to the different clusters. Whereas the second technique interprets the constraint information as the distance of data pairs and computes the pair-wise similarity for data clustering to ensure a small distance for must-link constraints and a large distance for cannot-link constraints.

COP-K means method [19] is used for pair-wise constrained clustering. It is based on the k-means algorithm and it is quick and easy to implement but it generates unsteady clustering results depending on the data assignment order. The authors in [16] have modified COP K-means (MC-KM) using a mechanism that satisfies data pairs constraints in order to verb their pre-given priorities. The boosting approach is used to improve the performance of MC-KM algorithm [17]. Boosting principle is one of machine learning techniques that make a highly accurate prediction rule from relatively inaccurate rules. The boosting strategy learns many weak hypotheses by adaptive control for probability distribution of data occurrence and combines them to learn a single strong hypothesis. Adaboost algorithm [20] is the first boosting algorithm that could be used in different applications. After that, there are many boosting algorithms that have been proposed to enhance the performance of the classification methods [21, 22] and clustering methods [17, 23]. The framework of boosting for data clustering is able to enhance the performance of the clustering method using the pair-wise constraints.

MC-KM can be used as a weak learner for the boosting framework. It generates a constrained clustering represented by kernel matrix according to the priorities of the constraints that are given by the boosting approach. The elements of the kernel matrix indicate whether or not the corresponding data pair belongs to the same cluster.

*Corresponding Author

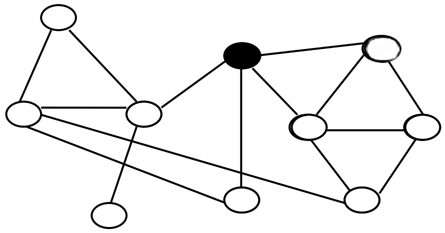


Fig. 1. An Example of Ego Network.

In this paper, we employ the boosting principle by learning constraints priorities for social networks circles discovery. The proposed method finds the communities in different social networks dataset. It uses two types of data to perform social networks clustering; profile information given by users and the topological structure of the network.

This paper is structured as follows: Section II presents the problem under consideration. Section III introduces the boosted constrained k-means method for social networks circles discovery. Section IV discusses the experimental results of the proposed method. Finally, the paper is concluded in Section V.

II. Problem Definition

The formal definition of social network circle discovery is described below. Given m group of Ego-networks $EGO = \{EGO_1, EGO_2, \dots, EGO_i, \dots, EGO_m\}$ where m is the number of Ego-network, $EGO_i = (V_i, E_i)$ is the user *i's* Ego-network i.e. the network of connections between *i's* friends, V_i is the set of users and E_i is the set of edges in EGO_i ego-network.

An edge $(u, v) \in E_i$ refers to the connection between *u* and *v* users where $(u, v) \in V_i$. The connection means that *u*, *v* users are friends, study in the same faculty, work in the same company, etc., and it depends on the nature of the social networks. Each user has feature vector via profile information or topological structure information.

The social circle discovery means to find a group of circles $C_1, C_2, \dots, C_k$ for each ego-network where C_j indicates a set of users with the same activities.

III. Boosted Constrained K-means Method for Social Networks Circles Discovery

In this section, we will employ the boosted constrained k-means algorithm to find the circles of social networks. We will use two types of the social networks information for feature vector of the users to perform the clustering; profile information given by users and the topological structure of the network.

A. Feature Definition for Social Network Circle Discovery

The circle discovery task for social network may use two types of information to carry out a comprehensive analysis on social network circles. These types are; the information extracted from the user profile and the information extracted from the topological structure of the network.

1) Features based on the user profile: Some information is encoded in the content of the user node which is called the user profile such as Facebook dataset [24]. The profile based features vector in Facebook dataset are birthday, education, first name, last name, gender, hometown, language, locale, location and work.

2) Features based on the topological structure: In common Neighbors metric, the similarity between nodes is relative to the number of their common Neighbors. Jaccard metric is one of the most common Neighbors metric that measures the network topological structure of a user as given in the following formula:

$$Jacc(i, j) = \frac{|\Gamma(i) \cap \Gamma(j)|}{|\Gamma(i) \cup \Gamma(j)|} \tag{1}$$

where $Jacc(i, j)$ is Jaccard metric between node i and node j, Γ(i) and Γ(j) are neighbours set of node i and node j respectively. The neighbours here describe the undirected edges between nodes.

B. Clustering Method based on Boosting Principle

In this paper, a constrained clustering approach based on MC-MK algorithm and the boosting principle [17] is used for community detection in social network.

BMC-KM Algorithm

Input: Data points $U = \{u_1, \dots, u_n\}$, Constraints $Con = \{(i_1, j_1, w_1), \dots, (i_{|con|}, j_{|con|}, w_{|con|})\}$ where $y \in \{1, -1\}$ and (*i*, *j*) is a pair of data index, *w* is the weight (priority) of constraint data pair, *β*,*α* are the parameters of loss function and k is the number of circles.
Output: The set of circles $C = \{C_1, C_2, \dots, C_k\}$
Algorithm steps:

```
1: Initialize the weight of each constraint
   $W_n^0 \leftarrow \frac{1}{|con|} (n = 1 \sim |con|)$
2: For r = 1 to R do
3: Run the algorithm of MC-KM as a weak learner using Con.
4: Create a weak kernel function $K_r$ according to clustering results as
   $K_r(u_i, u_j) = \begin{cases} 1 \text{ if } (u_i, u_j) \text{belongs to same circle} \\ -1 \text{ if } (u_i, u_j) \text{ belongs to different circles} \end{cases}$
5: Compute the error rate $\epsilon_r$ using $K_r$
   $\epsilon_r = \frac{\beta}{2} \cdot \frac{\sum_{n=1}^{|con|} W_n^r (1 - y_n K_r(i_n, j_n))}{\sum_{n=1}^{|con|} W_n^r}$
6: If $\epsilon_r = 0$ then $\delta_r = \delta^*$ and go to step 10
7: If $\epsilon_r \geq 0.5$ then $\delta_r = 0$ and go to step 10
8: Else
   Calculate the value of $\delta_r$ for $K_r$, using $\epsilon_r$
   $\delta_r = ln \left\{ \frac{1 + \epsilon_r}{1 - \epsilon_r} \right\}$
9: Update the weight of each constraint
   $W_n^{r+1} = W_n^r \exp \left\{ \frac{-\delta_r (y_n K_r(i_n, j_n) - \alpha)}{\beta} \right\}$
10: End for
```

11: Compute the final kernel matrix *K*.

$$K = \sum_{r=1}^{R} \delta_r K_r$$

12: Return the final set of circles *C* by running kernel K-means algorithm with *K*.

Boosting principle which is ensemble learning technique integrates weak hypotheses that are generated by a weak learner based on the MC-KM algorithm. MC-KM in step 3 calculates the priorities of the constraints and attempts to satisfy the constraints with higher priorities to provide different clustering results in each round. The clustering is represented by using a kernel matrix in step 4, of which each element indicates the state of data pair belongs to same circle or different circles. The weak hypothesis is used to compute the error rate in step 5 which indicates the rate of unsatisfied constraints. The value of δ_r for kernel matrix is calculated by using the value of ϵ_r. The boosting process stops according to the value of ϵ_r: when ϵ_r equal to zero($\epsilon_r = 0$). This means that all constraints are satisfied, and when $\epsilon_r \geq 0.5$ which means the weak learning condition is violated [17]. On the other hand, when $\epsilon_r < 0.5$, there is updating for the priority of each constraint using step 9. The priorities of unsatisfied constraints in step r of boosting process are increased but the priorities of the satisfied constraints are the same. When the boosting process is finished, the kernel matrices are integrated into a single kernel matrix *K* in step 11. The kernel k-means methods can be used for final clustering results.

The boosting process is interpreted as an optimization process to find the hypothesis that minimizes the loss function which is given as $\sum_{n=1}^{con} \exp(-y_n\, k_r(i_n, j_n))$where $K_r(i_n, j_n)$ is a function to predict where the data pair is a must-link or cannot-link and $y = \{1, -1\}$ points to the label of the data pair .The parameters *β, α* are to soften the gap in the values of the priority between the satisfied and unsatisfied constraints.

IV. Experimental Results

This section provides the experiments to evaluate the performance of the proposed method to find the circles in three datasets with ground-truth communities: Facebook, Cora and Citeseer. The proposed algorithm has been developed in Matlab 2016b and it has been tested in an Intel(R) Core(TM) i7-3630 QM (2.40GHz) and 6 GB RAM.

A. Dataset

We use two types of dataset; non-overlapping ground-truth communities; Cora and Citeseer dataset [25] and overlapping ground-truth communities Facebook dataset [24] to evaluate the proposed method. Table I gives a report about the network's statistics of the dataset where the 'Nodes' mean the users of the network, 'Edges' mean the connection between users and 'Circles' mean the communities that group users with the same activities.

The Facebook data set contains 10 Ego-networks that store and share different kinds of media information like photographs, videos and documents. This data set is considered as a real-world example with ground-truth that is correct definition for different communities of the Ego networks.

B. Evaluation Metrics

We utilize normalized mutual information (NMI) and F1-score as metrics for comparing our results with results in [15]. These metrics give a value between 0 and 1 where 1 is the optimal value. F1-score between the ground-truth circle *C** and predicted circles *C* can be calculated as:

$$F_1(C, C^*) = \frac{2 \times p(C,C^*) \times r(C,C^*)}{p(C,C^*) + r(C,C^*)} \quad (2)$$

where $p(C, C^*)$ is the precision of *C* to *C** and it is defined as:

$$p(C, C^*) = \frac{|\, C \cap C^* |}{|\, C \,|} \quad (3)$$

and $r(C, C^*)$ is the recall of *C* to *C** and it is defined as:

$$r(C, C^*) = \frac{|\, C \cap C^* |}{|\, C^* |} \quad (4)$$

NMI denotes the consistency between the ground-truth circle *C** and the predicted circles *C*. NMI can be calculated as follows.

Let *N* is the number of data points and *K* is the number of circles, *C* is the set of predicted circles and C^* is the set of ground-truth circles, then NMI can be defined as:

$$\mathrm{NMI} = \frac{\sum_{i=1}^{k} \sum_{j=1}^{k} n_{i,j}^{C,C^*} \log\left(\frac{N.N_{i,j}^{C,C^*}}{n_i^C n_j^{C^*}}\right)}{\sqrt{\left(\sum_{i=1}^{k} n_i^C \log\frac{n_i^C}{N}\right)\left(\sum_{i=1}^{k} n_i^{C^*} \log\frac{n_i^{C^*}}{N}\right)}} \quad (5)$$

Where n_i^C is the number of points in i^{th} circle in C, $n_j^{C^*}$ is the number of points in j^{th}circle in *C**, and $n_{i,j}^{C,C^*}$ is the number of points in both the i^{th} circle in *C* and the j^{th} circle in *C**.

TABLE. I. The Report about the Network's Statistics of the Dataset

Dataset	Nodes	Edges	Circles
Cora	2708	5278	7
Citeseer	3312	4536	6
FB ego-network0	348	2852	24
FB ego-network107	1046	27783	9
FB ego-network1684	793	14810	17
FB ego-network 1912	756	30772	46
FB ego-network 3437	548	5347	32
FB ego-network 348	228	3416	14
FB ego-network 3980	60	198	17
FB ego-network 414	160	1843	7
FB ego-network 686	171	1824	14
FB ego-network 698	67	331	13

C. Parameters Setting

The initial cluster centers are assigned using k-mean++ algorithm. The kernel k-mean algorithm with linear kernel and 1000 maximum iterations is used to get the final clustering results. The number of rounds of boosting operation is 100 rounds. The initial error rate is $\epsilon_r = 0$ and the initial $\delta_r^* = 100$. The proposed method is tested with 50% and 80% of constraints.

D. Clustering Performance

The performance of BMC-KM algorithm is evaluated against the MC-KM algorithm which is used as a weak learner. Table II. shows the results of Cora and Citseer dataset when NMI metric is used with three types of feature vector; Profile feature vector, structure network feature vector, and the fusion of the two vectors. The two algorithms are evaluated when 50% and 80% of constraints are used. The boosting principle enhances the results of MC-KM algorithm and the results with profile feature vector are the best of other two feature vectors for Cora and Citeseer dataset. This means that the profile information of users in Cora and Citeseer dataset is more discrimination then the structure of the network. When the number of constraints is increased the results of NMI are also increased. The value of NMI is 0.4243 and 0.5228 when the percentage of the constraints are 50% and 80% respectively with profile feature vector and boosted method with Cora dataset. Furthermore, NMI is 0.3308 and 0.5130 when the constraints percentage is 50% and 80% respectively with Citeseer dataset. The results of the proposed method are compared with the results in [15] as shown in Fig. 2 with Cora and Citeseer dataset. It is found that the proposed method with boosting principle gives better results than the state of the art [15]. Furthermore, the results of the proposed method for different Ego Networks that formed Facebook dataset is shown in Table III using F1-score. The performance of the Ego networks makes important variance because of the variance of the Ego network information. F1-score is increased when the percentage of the constraints that used in the algorithm is also increased. Fig. 3 shows a comparison between the proposed method and the method in [15] using F1-score. The performance of the proposed method is better than the method in [15] with Ego 0, Ego 107, Ego 1912, Ego 348, Ego 3980, Ego 414, Ego 686 and Ego 698 but it is slightly decreased with Ego 1684 and Ego 3437 due to the variance of the Ego network information. Different percentages of constraints can be used to evaluate the proposed algorithm. Once the percentages of constraints are increased, significant information is given to the algorithm and the value of evaluation metrics is increased. However, the algorithm will be slow.

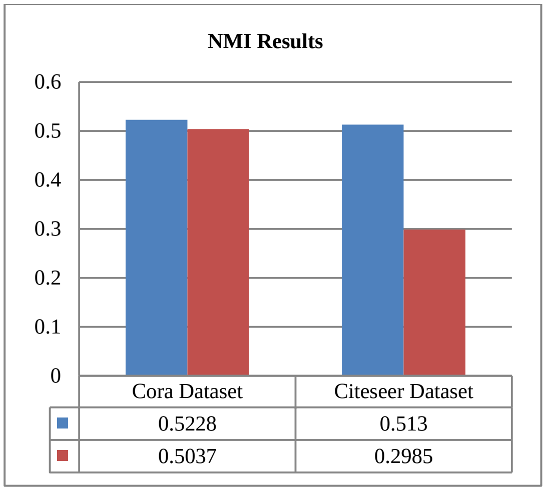


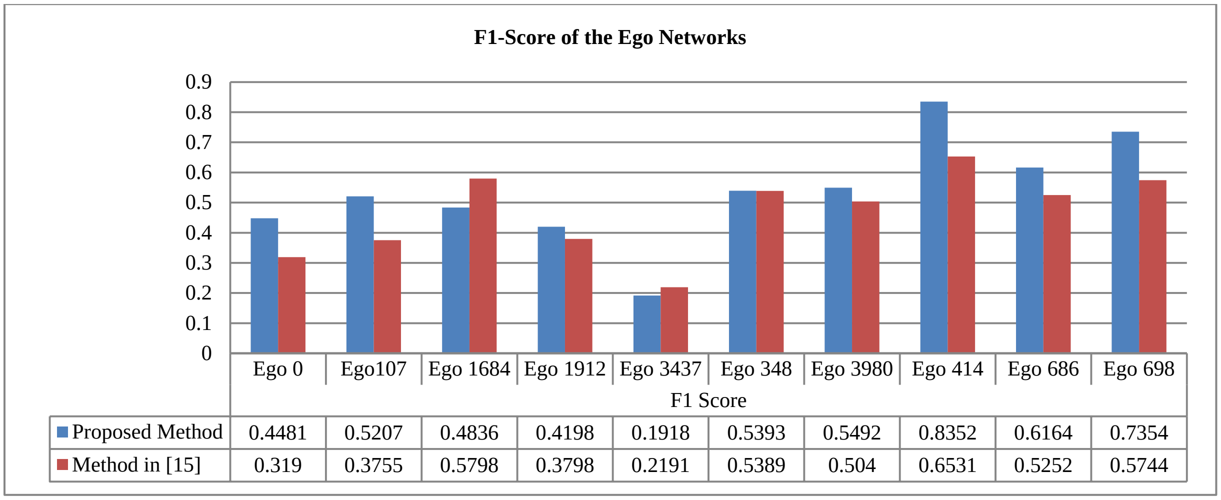
Fig. 2. The Comparison between the Proposed Boosting Method and the Method in [15].

TABLE. II. THE RESULTS OF NMI METRIC WITH THREE TYPES OF FEATURE VECTOR; PROFILE FEATURE VECTOR, STRUCTURE NETWORK FEATURE VECTOR, AND THE FUSION OF THE TWO FEATURE VECTORS WITH BMC-KM AND MC-KM FOR CORA AND CITESEER DATASET

Dataset	**Feature Vector**	**50% Constraints**			**80% Constraints**		
		Profile	***Struct***	***Fusion***	***Profile***	***Struct***	***Fusion***
Cora	BMC-KM	**0.4243**	0.0930	0.1459	**0.5228**	0.2405	0.3949
	MC-KM	0.2267	0.0930	0.0838	0.3245	0.2405	0.2553
Citeseer	BMC-KM	**0.3308**	0.2809	0.3168	**0.5130**	0.4514	**0.5130**
	MC-KM	0.2505	0.2141	0.2952	0.3079	0.2975	0.2662

TABLE. III. THE RESULTS OF F1-SCORE WITH THREE TYPES OF FEATURE VECTOR; PROFILE FEATURE VECTOR, STRUCTURE NETWORK FEATURE VECTOR, AND THE FUSION OF THE TWO FEATURE VECTORS WITH BMC-KM AND MC-KM FOR FACEBOOK EGO NETWORKS

Dataset	**50% Constraints**			**80% of Constraints**		
	Profile	***Struct***	***Fusion***	***Profile***	***Struct***	***Fusion***
Ego 0	0.2693	0.2090	0.2822	0.2926	0.3505	0.4481
Ego 107	0.3157	0.3671	0.3188	0.3555	0.5207	0.3783
Ego 1684	0.3110	0.4101	0.3847	0.4836	0.4248	0.4660
Ego 1912	0.1815	0.2407	0.2261	0.2744	0.4198	0.2549
Ego 3437	0.1139	0.1067	0.0882	0.1918	0.0845	0.1058
Ego 348	0.2738	0.4515	0.4161	0.4025	0.5393	0.4418
Ego 3980	0.2857	0.5273	0.3039	0.3454	0.4084	0.5492
Ego 414	0.4032	0.7681	0.7249	0.7317	0.8352	0.6888
Ego 686	0.3223	0.3543	0.3000	0.6146	0.6050	0.5845
Ego 698	0.3887	0.6685	0.5087	0.7014	0.6880	0.7354



	Ego 0	Ego107	Ego 1684	Ego 1912	Ego 3437	Ego 348	Ego 3980	Ego 414	Ego 686	Ego 698
🟦 Proposed Method	0.4481	0.5207	0.4836	0.4198	0.1918	0.5393	0.5492	0.8352	0.6164	0.7354
🟥 Method in [15]	0.319	0.3755	0.5798	0.3798	0.2191	0.5389	0.504	0.6531	0.5252	0.5744

Fig. 3. The Comparison between the Proposed Method and the Method in [15] using F1-score with Facebook Dataset.

V. Conclusion

This paper presented a clustering method based on boosting principle to find the circles of the social networks. The boosting framework is used with modified COP K-means method that gives priorities to the constraints. The boosting principle enhances the results of the weak learner with the three feature vectors; profile feature vector, structure feature vector and the fusion of the two vectors. The proposed method outperforms the state of the art with three datasets.

VI. Future Work

In the future, we will perform the proposed method on another dataset like Twitter and Google+ which are directed networks. Furthermore, we can use another weak learner algorithm for boosting framework to enhance the performance of social networks circles analysis.

References

[1] C. Wang, W. Tang, B. Sun, J. Fang, and Y. Wang, "Review on community detection algorithms in social networks," in 2015 IEEE International Conference on Progress in Informatics and Computing (PIC), 2015, pp. 551-555.

[2] J. Leskovec and J. J. Mcauley, "Learning to discover social circles in ego networks," in Advances in neural information processing systems, 2012, pp. 539-547.

[3] B. S. Preethi, "Improved BSP Clustering Algorithm for Social Network Analysis," Bonfring International Journal of Software Engineering and Soft Computing, vol. 1, pp. 15-20, 2011.

[4] J. Xie, S. Kelley, and B. K. Szymanski, "Overlapping community detection in networks: The state-of-the-art and comparative study," Acm computing surveys (csur), vol. 45, p. 43, 2013.

[5] D. Sudhakaran and S. Renjith, "Survey of Community Detection Algorithms to Identify the Best Community in Real-Time Networks," Vol-2, Issue-1, pp. 529-533, 2016.

[6] A. Gonzalez-Pardo, J. J. Jung, and D. Camacho, "ACO-based clustering for Ego Network analysis," Future Generation Computer Systems, vol. 66, pp. 160-170, 2017.

[7] W.-L. Zhao, C.-H. Deng, and C.-W. Ngo, "Boost k-means," arXiv preprint arXiv:1610.02483, 2016.

[8] T. Velmurugan and T. Santhanam, "Computational complexity between K-means and K-medoids clustering algorithms for normal and uniform distributions of data points," Journal of computer science, vol. 6, p. 363, 2010.

[9] J. Yin, Y. Zhang, and L. Gao, "Accelerating distributed Expectation–Maximization algorithms with frequent updates," Journal of Parallel and Distributed Computing, vol. 111, pp. 65-75, 2018.

[10] G. Tzortzis and A. Likas, "The global kernel k-means clustering algorithm," in 2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence), 2008, pp. 1977-1984.

[11] M. Filippone, F. Camastra, F. Masulli, and S. Rovetta, "A survey of kernel and spectral methods for clustering," Pattern recognition, vol. 41, pp. 176-190, 2008.

[12] Y. Kao and K. Cheng, "An ACO-based clustering algorithm," in International Workshop on Ant Colony Optimization and Swarm Intelligence, 2006, pp. 340-347.

[13] P. Bedi and C. Sharma, "Community detection in social networks," Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, vol. 6, pp. 115-135, 2016.

[14] X. Wang, P. Cui, J. Wang, J. Pei, W. Zhu, and S. Yang, "Community preserving network embedding," in Thirty-First AAAI Conference on Artificial Intelligence, 2017.

[15] Y. Li, C. Sha, X. Huang, and Y. Zhang, "Community detection in attributed graphs: an embedding approach," in Thirty-Second AAAI Conference on Artificial Intelligence, 2018.

[16] M. Okabe and S. Yamada, "Clustering by learning constraints priorities," in 2012 IEEE 12th International Conference on Data Mining, 2012, pp. 1050-1055.

[17] M. Okabe and S. Yamada, "Clustering Using Boosted Constrained k-Means Algorithm," Frontiers in Robotics and AI, vol. 5, p. 18, 2018.

[18] K. Q. Weinberger and L. K. Saul, "Distance metric learning for large margin nearest neighbor classification," Journal of Machine Learning Research, vol. 10, pp. 207-244, 2009.

[19] K. Wagstaff, C. Cardie, S. Rogers, and S. Schrödl, "Constrained k-means clustering with background knowledge," in Icml, 2001, pp. 577-584.

[20] Y. Freund, R. Schapire, and N. Abe, "A short introduction to boosting," Journal-Japanese Society For Artificial Intelligence, vol. 14, p. 1612, 1999.

[21] D. Nielsen, "Tree Boosting With XGBoost-Why Does XGBoost Win" Every" Machine Learning Competition?," NTNU, 2016.

[22] T. Pi, X. Li, Z. Zhang, D. Meng, F. Wu, J. Xiao, and Y. Zhuang, "Self-Paced Boost Learning for Classification," in IJCAI, 2016, pp. 1932-1938.

[23] D. Frossyniotis, A. Likas, and A. Stafylopatis, "A clustering method based on boosting," Pattern Recognition Letters, vol. 25, pp. 641-654, 2004.

[24] http://snap.stanford.edu/data/

[25] http://linqs.cs.umd.edu/projects/projects/lbc/.