LWMC: A Locally Weighted Meta-Clustering Algorithm for Ensemble Clustering

Dong Huang[1], Chang-Dong Wang[2,3,4⋆], and Jian-Huang Lai[2,3,4]

[1] College of Mathematics and Informatics, South China Agricultural University,China
[2] School of Data and Computer Science, Sun Yat-sen University, China
[3] Guangdong Key Laboratory of Information Security Technology, China
[4] Key Laboratory of Machine Intelligence and Advanced Computing, Ministry of Education, China
huangdonghere@gmail.com, changdongwang@hotmail.com, stsljh@mail.sysu.edu.cn

Abstract. The last decade has witnessed a rapid development of the ensemble clustering technique. Despite the great progress that has been made, there are still some challenging problems in the ensemble clustering research. In this paper, we aim to address two of the challenging problems in ensemble clustering, that is, the local weighting problem and the scalability problem. Specifically, a locally weighted meta-clustering (LWMC) algorithm is proposed, which is featured by two main advantages. First, it is highly efficient, due to its ability of working and voting on clusters. Second, it incorporates a locally weighted voting strategy in the meta-clustering process, which can exploit the diversity of clusters by means of local uncertainty estimation and ensemble-driven cluster validity. Experiments on eight real-world datasets demonstrate the superiority of the proposed algorithm in both clustering quality and efficiency.

Keywords: Ensemble clustering; Consensus clustering; Meta-clustering; Local weighting; Scalability

1 Introduction

Ensemble clustering is the process of fusing multiple clusterings, each referred to as a base clustering, into a probably better and more robust consensus clustering [1–12, 14–18]. It has proved to be an advantageous clustering technique in dealing with noisy data, finding clusters of arbitrary shapes, handling data from multiple sources, and constructing robust clustering result [16]. Despite its rapid development and significant success, there are still some challenging problems in ensemble clustering that remain to be tackled. In this paper, we pay attention to two of these challenging problems, i.e., the local weighting problem as well as the scalability problem, and propose a novel ensemble clustering algorithm based on locally weighted meta-clustering.

⋆ Corresponding author

In the past decade, various ensemble clustering algorithms have been designed by exploiting different techniques [1–12, 14–18]. Evidence accumulation clustering (EAC) [2] is one of the most classical ensemble clustering algorithms, which first builds a co-association matrix by considering the frequency that two objects appear in the same cluster among the multiple base clusterings, and then achieves the consensus clustering by means of hierarchical agglomerative clustering. Yi et al. [17] proposed to identify the uncertain pairs in the co-association matrix by a global threshold and recover them by the matrix completion technique. Fern and Brodley [1] formulated the ensemble clustering problem into a bipartite graph partitioning problem, where both clusters and objects are treated as graph nodes. These conventional ensemble clustering algorithms typically treat each base clsutering in the ensemble with equal weights, and fail to take into account the different reliability of the ensemble members.

In an clustering ensemble, there may be some low-quality, or even ill, ensemble members, which can significantly degrade the consensus performance. There is a need to evaluate the quality of the base clusterings and weight them accordingly. To this end, Li and Ding [10] proposed a weighted ensemble clustering method based on non-negative matrix factorization (NMF), where the weight of each base clustering is automatically determined in an optimization process. Huang et al. [4] proposed to evaluate the reliability of each base clustering by the normalized crowd agreement index (NCAI), and then presented two weighted ensemble clustering algorithms, termed weighted evidence accumulation clustering (WEAC) and graph partitioning with multi-granularity link analysis (GP-MGLA), respectively. These methods [4, 10] typically treat each base clustering as an individual and assign a global weight to each of them, but fail to explore the different reliability of the clusters inside the same base clustering. Different from the methods in [10] and [4], Huang et al. [8] proposed to estimate the uncertainty of clusters by an entropic criterion, and developed two locally weighted ensemble clustering algorithms, which are capable of evaluating and weighting the clusters in the ensemble without making specific assumption on the data distribution or access to the original data features. However, the algorithms in [8], as well as most of the existing ensemble clustering algorithms [1–4, 8–11, 14, 16–18], work at the object-level, i.e., they use the original objects as the basic operating units, which restricts their scalability for very large datasets. It remains an open problem how to tackle the local weighting issue and the scalability issue at the same time in a unified ensemble clustering framework.

In this paper, we propose a locally weighted meta-clustering (LWMC) algorithm for ensemble clustering. Each cluster is a subset of objects, and can be view as a local region in the dataset. The uncertainty of each cluster is first estimated by considering the distribution of cluster labels in the entire ensemble, and then an ensemble-driven cluster index (ECI) is computed as an indication of the reliability of this cluster. We build a cluster similarity graph (CSG) by treating each cluster as a node and deciding the edge weights with respect to the Jaccard coefficient. With the CSG partitioned into a certain number of meta-clusters, we then propose a locally weighted voting strategy to yield the final clustering

result. Different from the conventional meta-clustering algorithm (MCLA) [15], which treats all clusters equally, our LWMC algorithm is able to exploit the different reliability of clusters and weight them accordingly in the meta-clustering process. Experiments are conducted on multiple real-world datasets, which have shown the superiority of our ensemble clustering algorithm in both clustering quality and efficiency.

The rest of the paper is organized as follows. Section 2 describes the proposed ensemble clustering algorithm based on local weighting and meta-clustering. Section 3 reports the experimental results of the proposed algorithm against several baseline algorithms. Section 4 concludes the paper.

2 Proposed Algorithm

In this section, we describe the overall framework of the proposed algorithm. We first present the formulation of the ensemble clustering problem in Section 2.1, and then introduce the process of local uncertainty estimation and ensemble-driven cluster validity in Section 2.2. Finally, we present a new consensus function based on locally weighted mete-clustering to obtain the consensus result in Section 2.3.

2.1 Formulation of the Ensemble Clustering Problem

Ensemble clustering is the process of combining multiple base clusterings into a probably better and more robust consensus clustering. Let $\mathcal{X} = \{x_1, \cdots, x_N\}$ denote a dataset with N objects. The base clusterings for $\mathcal{X}$ can be generated by using different clustering algorithms or using the same algorithm with different parameter settings. Formally, an ensemble of M base clusterings can be denoted as follows:

$$\Pi = \{\pi^1, \cdots, \pi^M\}, \tag{1}$$

where π^m is the m-the base clustering in the ensemble Π. Each base clustering consists of a number of clusters. The m-th base clustering in Π can be denoted as

$$\pi^m = \{C_1^m, \cdots, C_{n^m}^m\}, \tag{2}$$

where C_i^m is the i-th cluster and n^m is the number of clusters in π^m. Further, we can denote the set of all clusters in the ensemble Π as

$$\mathcal{C} = \{C_1, \cdots, C_{N_c}\}, \tag{3}$$

where C_i is the i-th cluster and N_c is the total number of clusters in Π. It is obvious that $N_c = \sum_{m=1}^{M} n^m$. Then, given the ensemble Π, the objective of ensemble clustering is to build a better consensus clustering π^*.

2.2 From Local Uncertainty to Ensemble-Driven Cluster Validity

To deal with the potentially low-quality, or even ill, base clusterings, recently some weighted ensemble clustering approaches [4, 10] have been developed, which is able to (globally) evaluate and weight the base clusterings. However, these methods generally neglect the local diversity inside the same base clustering. A cluster can be viewed as a local region in a base clustering. Even in the same base clusterings, the quality of the clusters may be very different. To locally evaluate and weight the clusters, we follow the practice of [8] and resort to the concept of entropy, which is an important concept in information theory and indicates the uncertainty of a random variable.

Specifically, the uncertainty (entropy) of a cluster is measured by considering the distribution of cluster labels in the entire ensemble. We first consider the case of measuring the entropy of a cluster, say, $C_i \in \mathcal{C}$, w.r.t. one base clustering, say, π^m, which can be computed as follows:

$$H^m(C_i) = -\sum_{C_j^m \in \pi^m} p(C_i, C_j^m) \log_2 p(C_i, C_j^m) \tag{4}$$

with

$$p(C_i, C_j^m) = \frac{|C_i \bigcap C_j^m|}{|C_i|}, \tag{5}$$

where $\bigcap$ computes the intersection of two sets (or clusters), and $||$ outputs the number of objects in a set.

Based on the assumption that the base clusterings are independent of each other, we can further compute the entropy of cluster C_i w.r.t. the entire ensemble Π, that is

$$H^{\Pi}(C_i) = \sum_{m=1}^{M} H^m(C_i). \tag{6}$$

Note that $H^{\Pi}(C_i)$ indicates the uncertain of cluster C_i w.r.t. the ensemble Π. It is obviously that $H^{\Pi}(C_i) \in [0, +\infty)$ for any cluster $C_i \in \mathcal{C}$. When the objects in C_i belong to the same cluster in all of the base clusterings, the uncertainty of cluster C_i, i.e., $H^{\Pi}(C_i)$, reaches its minimum value 0.

With the uncertainty measure of clusters, the ensemble-driven cluster index (ECI) can be computed as follows:

$$ECI(C_i) = \mathrm{e}^{-\frac{H^{\Pi}(C_i)}{\theta \cdot M}}, \tag{7}$$

where $\theta > 0$ is a parameter to adjust the correlation between the cluster uncertainty and the ECI value. It holds that $ECI(C_i) \in (0, 1]$ for any cluster $C_i \in \mathcal{C}$. When the uncertainty of a cluster reaches its minimum 0, its ECI value reaches its maximum 1, which indicates the highest reliability (i.e., the lowest uncertainty) of it with consideration to the ensemble. In our algorithm, the ECI measure acts as a local weighting term to explore the diverse clusters in ensembles.

2.3 Finding Consensus by Locally Weighted Meta-Clustering

In this section, we devise a new consensus function based on locally weighted meta-clustering (LWMC). The cluster similarity graph (CSG) is first constructed, where the clusters in the ensemble are treated as the graph nodes and the edge weights between clusters are computed w.r.t. the Jaccard coefficient. Given two clusters C_i and C_j, the edge weight between them is computed as

$$e_{ij} = \frac{|C_i \bigcap C_j|}{|C_i \bigcup C_j|}. \tag{8}$$

With the CSG graph constructed, the normalized cut (Ncut) algorithm [13] is then adopted to partition the graph into a certain number of meta-clusters, denoted as

$$\mathcal{MC} = \{MC_1, \cdots, MC_K\}, \tag{9}$$

where MC_i denotes the i-th meta-cluster and K is the number of meta-clusters in $\mathcal{MC}$. Each meta-cluter is a set of clusters. The conventional meta-clustering algorithm [15] typically adopts a simple majority voting strategy to assign each object to one of the meta-clusters, which neglects the different reliability of clusters and may be misled by some low-quality clusters. In this paper, we propose a locally weighted voting strategy based on the ECI measure, which is able to exploit the diversity of clusters in ensembles. Given an object o_i and a meta-cluster MC_j, the locally weighted voting score of o_i w.r.t. MC_j is computed as

$$Score(o_i, MC_j) = \frac{1}{|MC_j|} \sum_{C_k \in MC_j} w(C_k) \cdot \mathbf{1}(o_i \in C_k), \tag{10}$$

with

$$w(C_k) = ECI(C_k), \tag{11}$$

$$\mathbf{1}(statement) = \begin{cases} 1, & \text{if statement is true,} \\ 0, & \text{otherwise,} \end{cases} \tag{12}$$

where $w(C_k)$ is the local weighting term, $\mathbf{1}(statement)$ is the voting term, and $|MC_j|$ is the number of clusters in the meta-cluster MC_j. Then, each object will be assigned to the meta-cluster that gives it the highest score, that is

$$MetaCls(o_i) = \arg \max_{MC_j \in \mathcal{MC}} Score(o_i, MC_j). \tag{13}$$

With each object assigned to a meta-cluster by the locally weighted voting strategy, the consensus clustering can be obtained by treating the objects assigned to the same meta-cluster as a final cluster.

3 Experiments

In this section, we conduct experiments on a variety of real-world datasets to evaluate the propoased LWMC algorithm against several other ensemble clustering algorithms.

3.1 Datasets and Evaluation Method

In our experiments, eight real-world datasets are used, namely, *Semeion*, *Steel Plates Faults (SPF)*, *Multiple Features (MF)*, *MNIST*, *Texture*, *ISOLET*, *USPS*, and *Letter Recognition (LR)*. The MNIST and USPS datasets are from Dr. S. Roweiss homepage[1], while the other six datasets are from UCI Machine Learning Repository[2]. The details of these datasets are given in Table 1.

Table 1. The benchmark datasets.

Dataset	*Semeion*	*SPF*	*MF*	*MNIST*	*Texture*	*ISOLET*	*USPS*	*LR*
#Object	1,593	1,941	2,000	5,000	5,500	7,797	11,000	20,000
#Class	10	7	10	10	11	26	10	26
#Attribute	256	27	649	784	40	617	256	16

To provide a fair comparison, in each test, we run the proposed algorithm as well as the baseline algorithms a large number of times and report their average performances. At each run, an ensemble of ten base clusterings is generated by k-means with the cluster number k randomly selected in $[2, \sqrt{N}]$ for each base clustering. To quantify the clustering performance, we adopt the normalized mutual information (NMI) [15] as the evaluation measure. Note that a larger NMI value indicates a better clustering result.

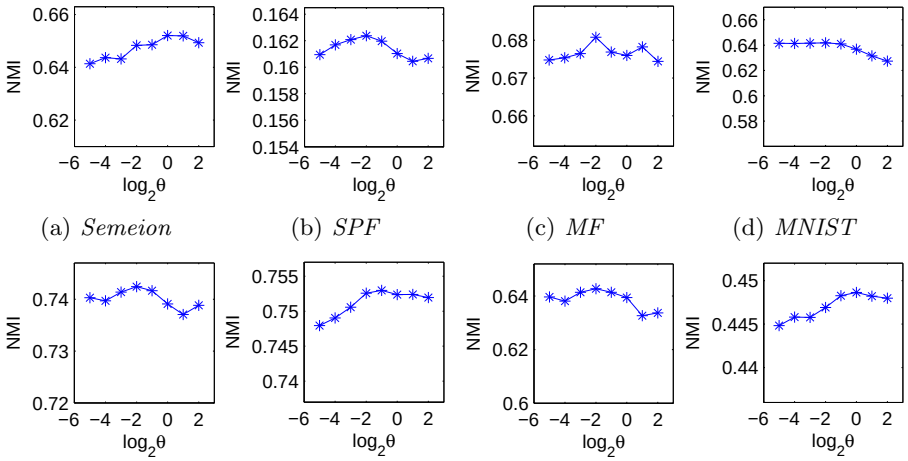

Fig. 1. The average NMI scores over 20 runs by LWMC with varying values of parameter θ. Note that the X axis corresponds to $\log_2 \theta$.

[1] http://www.cs.nyu.edu/%7eroweis/data.html
[2] http://archive.ics.uci.edu/ml

3.2 Sensitivity of Parameter θ

In this section, we test the sensitivity of parameter θ. As can be seen in Fig.1, the proposed algorithm exhibits consistent performance with varying values of parameter θ. Note that the X axis corresponds to $\log_2 \theta$. Empirically, it is suggested that the parameter θ be set to moderate values, e.g., with $\log_2 \theta \in [-2, 0]$, which corresponds to $\theta \in [0.25, 1]$. In the following, we will use $\theta = 0.5$ in the experiments for all datasets.

3.3 Comparison with Base Clusterings

The objective of ensemble clustering is to combine multiple base clusterings to build a better consensus clustering. In this section, we compare the consensus clustering produced by the proposed LWMC algorithm against the base clusterings. As shown in Fig. 2, LWMC is capable of producing significantly better clustering results than the base clusterings. In particular, despite the low quality of the base clusterings for the *SPF* dataset (according to their low NMI scores), the LWMC algorithm can still yield much better consensus clustering results.

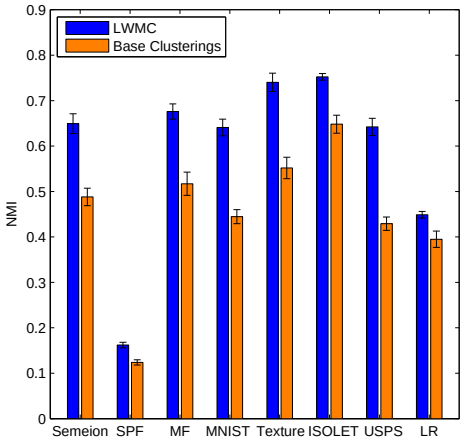

Fig. 2. Average Performances (over 20 runs) of our algorithm and the base clusterings.

3.4 Comparison with Other Ensemble Clustering Algorithms

In this section, we evaluate the performance of the proposed algorithm against eight baseline algorithms, namely, SEC [12], KCC [16], GP-MGLA [4], WEAC (with average-link) [4], EAC (with average-link) [2], CSPA [15], HGPA [15], and MCLA [15]. As shown in Table 2, the proposed LWMC algorithm shows a consistently good performance on the benchmark datasets. Although the WEAC

Table 2. Average performances (w.r.t. NMI) over 20 runs by different ensemble clustering methods (The best NMI score for each dataset is highlighted in bold).

Method	*Semeion*	*SPF*	*MF*	*MNIST*
LWMC	**0.649**$_{\pm 0.022}$	**0.162**$_{\pm 0.006}$	**0.676**$_{\pm 0.017}$	**0.641**$_{\pm 0.018}$
SEC	0.545$_{\pm 0.023}$	0.132$_{\pm 0.008}$	0.597$_{\pm 0.018}$	0.499$_{\pm 0.026}$
KCC	0.549$_{\pm 0.018}$	0.131$_{\pm 0.009}$	0.596$_{\pm 0.016}$	0.518$_{\pm 0.017}$
GP-MGLA	0.642$_{\pm 0.026}$	0.154$_{\pm 0.007}$	0.669$_{\pm 0.019}$	0.628$_{\pm 0.030}$
WEAC	0.644$_{\pm 0.025}$	0.152$_{\pm 0.009}$	0.643$_{\pm 0.022}$	0.616$_{\pm 0.028}$
EAC	0.641$_{\pm 0.027}$	0.152$_{\pm 0.009}$	0.635$_{\pm 0.023}$	0.601$_{\pm 0.032}$
CSPA	0.553$_{\pm 0.036}$	0.117$_{\pm 0.009}$	0.623$_{\pm 0.019}$	0.509$_{\pm 0.049}$
HGPA	0.491$_{\pm 0.027}$	0.116$_{\pm 0.010}$	0.535$_{\pm 0.482}$	0.409$_{\pm 0.028}$
MCLA	0.583$_{\pm 0.022}$	0.143$_{\pm 0.011}$	0.642$_{\pm 0.034}$	0.563$_{\pm 0.035}$
Method	*Texture*	*ISOLET*	*USPS*	*LR*
LWMC	**0.740**$_{\pm 0.020}$	0.752$_{\pm 0.007}$	**0.642**$_{\pm 0.019}$	**0.449**$_{\pm 0.007}$
SEC	0.644$_{\pm 0.015}$	0.694$_{\pm 0.016}$	0.473$_{\pm 0.021}$	0.412$_{\pm 0.008}$
KCC	0.644$_{\pm 0.014}$	0.690$_{\pm 0.009}$	0.501$_{\pm 0.016}$	0.409$_{\pm 0.006}$
GP-MGLA	0.725$_{\pm 0.025}$	0.749$_{\pm 0.007}$	0.615$_{\pm 0.038}$	0.440$_{\pm 0.004}$
WEAC	0.729$_{\pm 0.028}$	**0.753**$_{\pm 0.008}$	0.592$_{\pm 0.038}$	0.437$_{\pm 0.007}$
EAC	0.714$_{\pm 0.024}$	0.742$_{\pm 0.010}$	0.589$_{\pm 0.041}$	0.432$_{\pm 0.008}$
CSPA	0.653$_{\pm 0.022}$	0.700$_{\pm 0.030}$	0.509$_{\pm 0.061}$	0.262$_{\pm 0.170}$
HGPA	0.492$_{\pm 0.035}$	0.629$_{\pm 0.023}$	0.361$_{\pm 0.031}$	0.361$_{\pm 0.006}$
MCLA	0.703$_{\pm 0.014}$	0.723$_{\pm 0.020}$	0.545$_{\pm 0.030}$	0.410$_{\pm 0.013}$

method marginally outperforms our method on the *ISOLET* dataset, yet on all of the other seven datasets our method yield higher, or significantly higher, NMI scores than WEAC. To conclude, the proposed LWMC algorithm exhibits overall the best performance when compared to the eight baseline algorithms.

3.5 Execution Time

In this section, we test the execution time of different ensemble clustering algorithms with varying data sizes. In our experiments, different subsets of the *LR* datasets are used, whose sizes range from 0 to $20,000$. As shown in Fig. 3, LWMC is the fastest algorithm, whild MCLA is the second fastest one. To process the entire dataset of *LR*, the proposed LWMC method and the MCLA method consume 3.74 seconds and 5.31 seconds, respectively. Despite the extra time cost of the local weighting component, the proposed LWMC method still runs faster than the conventional MCLA method, probably due to the effeciency of the Ncut algorithm we used for graph partitioning.

The experiments are conducted in MATLAB R2014a 64-bit on a workstation with 8 Intel 2.40 GHz processors and 96 GB of RAM.

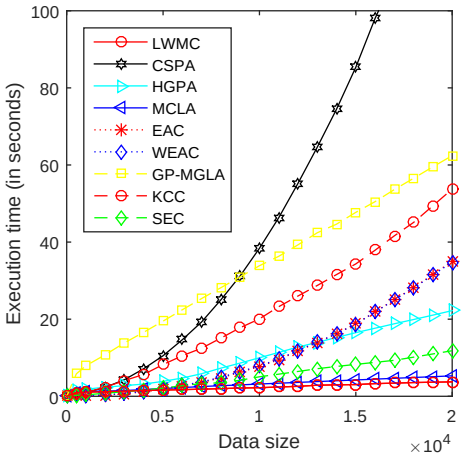

Fig. 3. Execution time of different methods with varying data sizes.

4 Conclusion

This paper proposes a locally weighted meta-clustering (LWMC) algorithm for ensemble clustering. Local uncertainty in ensembles is estimated by exploiting an entropic criterion, based on which the ensemble-driven cluster index can be obtained to evaluate and weight the clusters. Then a meta-cluster based consensus function is proposed, which exhibits two main advantages in the consensus process. First, it works at the cluster-level and is efficient for dealing with large-scale datasets. Second, it is able to exploit the diversity of clusters by incorporating a locally weighted voting strategy in the meta-clustering phase. Experiments on eight real-world datasets have shown the effectiveness and efficiency of the proposed algorithm.

5 Acknowledgement

This work was supported by NSFC (61602189, 61502543 & 61573387), the PhD Start-up Fund of Natural Science Foundation of Guangdong Province, China (2016A030310457), Guangdong Natural Science Funds for Distinguished Young Scholars (2016A030306014), and the Tip-Top Scientific and Technical Innovative Youth Talents of Guangdong Special Support Program (No. 2016TQ03X542).

References

1. Fern, X.Z., Brodley, C.E.: Solving cluster ensemble problems by bipartite graph partitioning. In: International Conference on Machine Learning (ICML'04) (2004)

2. Fred, A.L.N., Jain, A.K.: Combining multiple clusterings using evidence accumulation. IEEE Transactions on Pattern Analysis and Machine Intelligence 27(6), 835–850 (2005)
3. Huang, D., Lai, J.H., Wang, C.D.: Exploiting the wisdom of crowd: A multi-granularity approach to clustering ensemble. In: International Conference on Intelligence Science and Big Data Engineering (IScIDE'13). pp. 112–119 (2013)
4. Huang, D., Lai, J.H., Wang, C.D.: Combining multiple clusterings via crowd agreement estimation and multi-granularity link analysis. Neurocomputing 170, 240–250 (2015)
5. Huang, D., Lai, J.H., Wang, C.D.: Robust ensemble clustering using probability trajectories. IEEE Transactions on Knowledge and Data Engineering 28(5), 1312–1326 (2016)
6. Huang, D., Lai, J.H., Wang, C.D., Yuen, P.C.: Ensembling over-segmentations: From weak evidence to strong segmentation. Neurocomputing 207, 416–427 (2016)
7. Huang, D., Lai, J., Wang, C.D.: Ensemble clustering using factor graph. Pattern Recognition 50, 131–142 (2016)
8. Huang, D., Wang, C.D., Lai, J.H.: Locally weighted ensemble clustering. IEEE Transactions on Cybernetics, DOI: 10.1109/TCYB.2017.2702343 (2017)
9. Iam-On, N., Boongoen, T., Garrett, S., Price, C.: A link-based approach to the cluster ensemble problem. IEEE Transactions on Pattern Analysis and Machine Intelligence 33(12), 2396–2409 (2011)
10. Li, T., Ding, C.: Weighted consensus clustering. In: SIAM International Conference on Data Mining (SDM'08). pp. 798–809 (2008)
11. Li, Y., Yu, J., Hao, P., Li, Z.: Clustering ensembles based on normalized edges. In: Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD'07). pp. 664–671 (2007)
12. Liu, H., Wu, J., Liu, T., Tao, D., Fu, Y.: Spectral ensemble clustering via weighted k-means: Theoretical and practical evidence. IEEE Transactions on Knowledge and Data Engineering 29(5), 1129–1143 (2017)
13. Shi, J., Malik, J.: Normalized cuts and image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence 22(8), 888–905 (2000)
14. Singh, V., Mukherjee, L., Peng, J., Xu, J.: Ensemble clustering using semidefinite programming with applications. Machine Learning 79(1-2), 177–200 (2010)
15. Strehl, A., Ghosh, J.: Cluster ensembles: A knowledge reuse framework for combining multiple partitions. Journal of Machine Learning Research 3, 583–617 (2003)
16. Wu, J., Liu, H., Xiong, H., Cao, J., Chen, J.: K-means-based consensus clustering: A unified view. IEEE Transactions on Knowledge and Data Engineering 27(1), 155–169 (2015)
17. Yi, J., Yang, T., Jin, R., Jain, A.K.: Robust ensemble clustering by matrix completion. In: IEEE International Conference on Data Mining (ICDM'12). pp. 1176–1181 (2012)
18. Zhong, C., Yue, X., Zhang, Z., Lei, J.: A clustering ensemble: Two-level-refined co-association matrix with path-based transformation. Pattern Recognition 48(8), 2699– 2709 (2015)