ICIC Express Letters
Volume **10**, Number **6**, June **2016**

 ISSN 1881-803X
pp. **1469–1474**

SEMI-SUPERVISED SPECTRAL CLUSTERING ENSEMBLE

Jingya Yang and Linfu Sun

School of Information Science and Technology
Southwest Jiaotong University
No. 111, Erhuan Road, North Sec. 1, Chengdu 610031, P. R. China
yangjingya@my.swjtu.edu.cn

Received November 2015; accepted February 2016

Abstract. *Cluster ensemble is a hot research topic in machine learning field. It combines different base clustering results to get a consensus one. In this paper, we design a semi-supervised spectral clustering ensemble (SSSCE) model, based on both semi-supervised learning and spectral clustering algorithm. There are three contributions. The first is that the base clustering results are represented as a similarity matrix, and pairwise constraints are used to modify the similarity matrix, which is guided by the semi-supervised learning. The second is that spectral clustering algorithm is applied to processing the similarity matrix for a consensus cluster result. Finally, some standard UCI and Microsoft datasets are used for experiments and the experimental results show that SSSCE outperforms other cluster ensemble algorithms as well as spectral clustering ensemble.*

Keywords: Semi-supervised learning, Semi-supervised spectral clustering ensemble, Pairwise constraints, Similarity matrix

1. **Introduction.** Cluster ensembles address the problem of combining multiple clusterings of a set of objects into a single consolidated clustering [1], and there are three primary motivations for developing cluster ensembles. First, cluster ensembles can exploit and reuse existing knowledge implicit in legacy clusterings. Moreover, cluster ensembles can enable clustering over distributed datasets in cases where the raw data cannot be shared or pooled together because of restrictions due to ownership, privacy, storage, etc. Also, cluster ensembles can improve the quality and robustness of results. Therefore, many algorithms have been proposed for cluster ensembles [2, 3], especially for the consensus function [1, 4, 5, 6].

Note that the choice of consensus functions is very important for ensembles, and inappropriate consensus function will seriously affect the accuracy of the results. A family of spectral clustering (SC) algorithm [7] emerging in recent years have shown great promise. Compared with traditional clustering algorithms, SC has some obvious advantages. It can recognize the clusters of unusual shapes and obtain the globally optimal solutions in a relaxed continuous domain by eigen decomposition. In the existing cluster ensemble algorithms, SC is often used as the clusterer to generate base clustering results [8, 9, 10]. However, SC is computationally expensive, and the overall efficiency of the algorithm will be largely reduced if SC is used as the clusterer, whereas if SC is used as the consensus function, better final results can be obtained.

Furthermore, semi-supervised learning can not only make the model or model parameters more accurate, but also increase the model stability and robustness by utilizing some labeled data, of which a relatively common method is pairwise constraints [11]. Yet, current algorithms for spectral clustering ensemble (SCE) are mostly unsupervised algorithms [12, 13, 14], which cannot take advantages of the known information of datasets. As a result, the precision, robustness, and stability of the algorithms are degraded.

As to the analysis above, this paper presents a semi-supervised spectral clustering ensemble model. It uses pairwise constraints derived from the labeled data to modify the similarity matrix generated from multiple base clusterings, and reclusters the similarity matrix with SC algorithm, which takes the advantages of both semi-supervised learning and SC algorithm.

The remainder of this paper is organized as follows. We state the concept of semi-supervised cluster ensemble and implementation scheme in Section 2. In Section 3, generation of diverse base clusterings is introduced. In Section 4, the formulation and algorithm of SSSCE are presented. Experimental results in comparison with other methods are provided in Section 5. Conclusions are drawn in Section 6.

2. **The Semi-Supervised Cluster Ensemble Problem.** Suppose we are given a set of objects $O = \{o_1, o_2, \ldots, o_n\}$, with r being the number of clusterings of these n objects, a consensus function Γ is defined as a function $N^{n\times r} \to N^n$ mapping a set of clusterings to an integrated clustering $\Gamma : \{\lambda^{(q)}|q \in \{1, 2, \ldots, r\}\} \to \lambda$, in which $\lambda^{(q)} = \{\lambda_l^{(q)}|l \in \{1, 2, \ldots, k\}\}$ denotes a partitioning of objects set O into K clusters. Moreover, there are two sets of pairwise constraints including must-links: $M = \{(o_i, o_j)|o_i$ and o_j are in the same cluster$\}$, and cannot-links: $C = \{(o_i, o_j)|o_i$ and o_j are in two different clusters$\}$.

A reasonable goal for cluster ensembles is to seek a clustering that shares the most information with the base clusterings. Besides, for semi-supervised cluster ensembles, another goal is to learn a similarity matrix S such that the distances of point pairs in M are as small as possible, while those in C are as large as possible.

In addition, semi-supervised cluster ensembles consist of two phases: the generation of base clusterings and the combination of multiple base clustering results.

3. **Generation of Diverse Base Clusterings.** The first step of cluster ensembles is to produce multiple clustering results with differences, which reflect the structure of data sets from different aspects in favor of integration. At this stage, the different clustering algorithms [15] and the same clustering algorithm with different initialization or parameters are both useful for improving the diversity of the components.

Among the existing clustering algorithms, k-means algorithm is the simplest and efficient one. However, it is well known that k-means clustering is sensitive to the initialization. Obviously, such a fact is not desirable for applications but useful for constructing an ensemble. Therefore, we randomly select the parameter k_i of the ith base k-means from the preestablished interval $[k_{\min}, k_{\max}]$ with uniform distribution in our method. Using random initialization in base clustering not only avoids the cost of accurate initialization for each individual k-means, but also provides the required diversity for cluster ensembles.

4. **Spectral Clustering Ensemble with Pairwise Constraints.** In this section, we introduce how to combine the base clusterings, namely the r label vectors $\left\{\lambda^{(q)}|q \in \{1, 2, \ldots, r\}\right\}$ generated from k-means with random initialization, with pairwise constraints and SC algorithm. The detailed steps are as follows.

4.1. **Similarity matrix construction.** The first step of combination is to transform the label vectors into a suitable hypergraph H representation. The concatenated block matrix $H = H^{(1,2,\ldots,r)} = \left(H^{(1)} \ldots H^{(r)}\right)$ defines the adjacency matrix of a hypergraph with n vertices and $\sum_{q=1}^{r} K^{(q)}$ hyperedges, where $H^{(q)} = \left\{h_a|a \in \left\{1, 2, \ldots, K^{(q)}\right\}\right\}$. Each column vector h_a specifies a hyperedge h_a, where 1 indicates that the vertex corresponding to the row is part of that hyperedge and 0 indicates that it is not.

Then, we construct the similarity matrix according to cluster-based similarity partitioning algorithm (CSPA) [1], which can be computed in one sparse matrix multiplication:

$$S = \frac{1}{r} H H^T \tag{1}$$

where matrix H^T is the transposition of matrix H, and S is $n \times n$ sparse matrix.

4.2. **Similarity matrix modified with pairwise constraints.** The second step of combination is to modify the similarity matrix with pairwise constraints. S_{ij}, any one element of the similarity matrix S, represents the similarity of the point pair o_i and o_j. If o_i and o_j are labeled in the same class, S_{ij} equals 1. On the other hand, if o_i and o_j are labeled in two diffierent classes, S_{ij} equals 0.

So we make use of these limited degrees of supervision to modify the similarity matrix S to improve the accuracy of cluster ensembles.

i) if the point pair (o_i, o_j) belongs to must-link constraint, $S_{ij} = 1$;

ii) if the point pair (o_i, o_j) belongs to cannot-link constraint, $S_{ij} = 0$.

4.3. **Semi-supervised spectral cluster ensembles algorithm.** In this subsection, we introduce the third step of combination, in which the similarity matrix is reclustered with SC algorithm described as below.

In general, SC first maps data sets into a new space by the eigenvectors of an affinity matrix, which defines the similarities of data sets. Moreover, a Gaussian function was often used as the similarity function with form

$$W_{ij} = \exp\left(-\|o_i - o_j\|^2 / 2\sigma^2\right) \tag{2}$$

where σ is the scaling parameter, and W_{ij} is the similarity of samples o_i and o_j. The top k eigenvectors are used as k-dimension indicator vectors for samples. Then, a simple clustering algorithm such as k-means clustering is used to get K clusters.

In conclusion, we design SSSCE algorithm by incorporating semi-supervised learning and SC algorithm into the cluster ensemble process. SSSCE algorithm is summarized as follows.

Algorithm: SSSCE (Semi-Supervised Spectral Cluster Ensembles)

Input: objects set $O = \{o_1, o_2, \ldots, o_n\}$, number of clusters K, number of component clustering r, M denotes the set of must-link data points, and C denotes the set of cannot-link data points

1. Generation of component clustering for ensemble
   for $q = 1 : r$
   k_q: randomly selected from $[k_{\min}, k_{\max}]$
   $\lambda^{(q)} = k\text{-}means(O, k_q)$.
   end for
2. Transform the given cluster label vectors $\{\lambda^{(q)} | q \in \{1, 2, \ldots, r\}\}$ into hypergraph H.
3. Construct the similarity matrix $S = \frac{1}{r} H H^T$, and $S \in R^{n \times n}$, $H \in R^{n \times d}$.
4. Amend the similarity matrix S with pairwise constraint information: if the point pair (o_i, o_j) belongs to M, $S_{ij} = 1$; if the point pair (o_i, o_j) belongs to C, $S_{ij} = 0$.
5. Form the affinity matrix $W \in R^{n \times n}$ defined by $W_{ij} = \exp\left(-\|s^{(i)} - s^{(j)}\|^2 / 2\sigma^2\right)$ if $i \neq j$, and $W_{ii} = 0$; $s^{(i)}$ and $s^{(j)}$ denote the ith and jth row of S.
6. Define D to be the diagonal matrix where $D(i, i)$ is the sum of W's ith row, and construct the matrix $L = D^{-1/2} W D^{-1/2}$.
7. Find the k largest eigenvectors $x_1, x_2, \ldots, x_k$ of L, and form the matrix $X = \{x_1, x_2, \ldots, x_k\} \in R^{n \times k}$ by stacking the eigenvectors in columns.
8. Form the matrix Y from X by renormalizing each of X's rows to have unit length, i.e., $Y_{ij} = X_{ij} / \left(\sum_j X_{ij}^2\right)^{1/2}$.
9. Treating each row of Y as a point in R^k, cluster them into K clusters via k-means.
10. Finally, assign the base point o_i to cluster j if and only if row i of the matrix Y was assigned to cluster j.

Output: K clusters of O.

---

5. Experimental Study.

5.1. Datasets and evaluation criteria.

In this section, we run experiments on datasets from UCI machine learning repository, and the number of instances, features and classes in each data set are listed in Table 1.

TABLE 1. The number of the instances, features, and classes in each dataset

Dataset	Instances	Features	Classes
beer	870	892	3
congressEW	435	16	2
aerosol	905	892	3
alph	814	892	3
alphabet	814	892	3
amber	880	892	3
ambulances	930	892	3
americanflag	873	892	3
anonovo	732	892	3
apple	871	899	3
seed	210	7	3
aquarium	922	892	3
arrow	834	892	3
balance	625	4	3
banana	840	892	3
baobab	900	892	3

For all datasets, there are two steps leading to the final consensus clustering. First, we run k-means algorithms with random initialization to get a set of base clustering results. Second, various cluster ensemble algorithms, including CSPA, HyperGraph Partitioning Algorithm (HGPA), Meta-CLustering Algorithm (MCLA), SSSCE, Expectation Maximization consensus Number (EMcN), Quadratic Mutual Information consensus (QMIc), and SCE, are applied to combining the base clustering results into a consensus clustering.

For evaluation, we use micro-precision [16] to measure accuracy of the consensus cluster with respect to the true labels. The micro-precision is defined as:

$$MP = \frac{1}{n}\sum_{h=1}^{K} a_h \tag{3}$$

where K is the number of clusters, n is the number of objects, and a_h denotes the number of objects in consensus cluster h that are correctly assigned to the corresponding class. Then, $0 \leq MP \leq 1$ with 1 indicating the best possible consensus clustering, which has to be in full agreement with the class labels.

The adjusted rand index (ARI) [17] is another measure of the accuracy between two clusterings. Let U and V be two partitions on dataset O and suppose that U is our external criterion and V is a clustering result. Let n_{ij} be the number of objects that are in both class u_i and cluster v_j. Let $n_{i\cdot}$ and $n_{\cdot j}$ be the number of objects in class u_i and cluster v_j respectively. Then the ARI between these two partitions can be defined as:

$$ARI(U,V) = \frac{\sum_{ij}\binom{n_{ij}}{2} - \left[\sum_i \binom{n_{i\cdot}}{2}\sum_j \binom{n_{\cdot j}}{2}\right]\Big/\binom{n}{2}}{\frac{1}{2}\left[\sum_i \binom{n_{i\cdot}}{2} + \sum_j \binom{n_{\cdot j}}{2}\right] - \left[\sum_i \binom{n_{i\cdot}}{2}\sum_j \binom{n_{\cdot j}}{2}\right]\Big/\binom{n}{2}} \tag{4}$$

If the partitions U and V are completely independent, the value of the adjusted rand index $ARI(U, V)$ is 0, which means that nothing can be predicted about V by observing U and vice versa. If the two partitions are the same, the value of the ARI is 1.

5.2. **Experimental results.** Given n objects, we first use k-means on 16 datasets to generate 10 base clustering results for each dataset with 10 random initializations. Cluster ensemble algorithms are then applied on the 10 results. The average MPs and standard deviations over all datasets are reported in Table 2, and the ARIs are in Table 3. In addition, the first column both in two tables is the average value of 10 k-means clustering results, the maximum value of which is the second one.

The key observations from Table 2 can be summarized as follows: (i) SSSCE almost always has a higher average MP than base clustering results, which means the consensus clustering from SSSCE is indeed better than the base clusterings in quality; (ii) SSSCE outperforms other cluster ensemble algorithms for most of the datasets in terms of average MP, whereas the performance of SCE is not much better than other cluster ensemble algorithms, which means that semi-supervised learning has a very important practical significance to improve the performance of clustering.

TABLE 2. k-means with different initializations are used as base clustering algorithms, and the results are average MPs and standard deviations for different cluster ensemble algorithms. The highest MP among different algorithms on each data set is bolded.

	AK-means	MK-means	CSPA	HGPA	MCLA	SSSCE	EMcN	QMIc	SCE
beer	.448±0.022	.543±0.000	.417±0.010	.427±0.044	.462±0.059	**.682±0.199**	.404±0.032	.439±0.065	.429±0.024
congressEW	.853±0.005	**.871±0.000**	.828±0.000	.517±0.005	.867±0.008	.867±0.008	**.871±0.000**	.766±0.129	**.871±0.000**
aerosol	.386±0.030	.493±0.000	.380±0.007	.394±0.013	.387±0.037	**.598±0.079**	.381±0.027	.414±0.069	.367±0.042
alph	.358±0.012	.442±0.000	.426±0.004	.421±0.007	.342±0.008	**.666±0.136**	.372±0.037	.407±0.037	.340±0.007
alphabet	.343±0.017	.442±0.000	.422±0.003	.427±0.009	.347±0.021	**.603±0.085**	.364±0.029	.394±0.036	.345±0.009
amber	.638±0.026	.752±0.000	.523±0.001	.491±0.077	.577±0.062	**.826±0.047**	.645±0.062	.615±0.043	.596±0.055
ambulances	.503±0.006	.520±0.000	.387±0.004	.414±0.002	.449±0.082	**.726±0.000**	.439±0.074	.410±0.082	.343±0.007
americanflag	.450±0.019	.530±0.000	.430±0.004	.359±0.004	.455±0.031	**.580±0.004**	.463±0.055	.537±0.007	.443±0.045
anonovo	.419±0.037	.518±0.000	.509±0.001	.434±0.032	.427±0.001	**.662±0.124**	.469±0.052	.472±0.052	.452±0.049
apple	.640±0.007	.658±0.000	.617±0.000	.656±0.000	.612±0.002	**.782±0.123**	.610±0.003	.721±0.089	.612±0.002
seed	.637±0.029	.895±0.000	.884±0.008	.614±0.040	.883±0.015	.886±0.031	.793±0.113	.753±0.121	**.900±0.004**
aquarium	.497±0.041	.701±0.000	.401±0.018	.422±0.026	.550±0.079	**.580±0.116**	.407±0.046	.422±0.056	.387±0.051
arrow	.465±0.017	.558±0.000	.478±0.006	.384±0.004	.500±0.002	**.664±0.086**	.494±0.009	.523±0.036	.500±0.000
balance	.532±0.005	.594±0.000	.506±0.006	.422±0.002	.522±0.004	**.715±0.094**	.547±0.031	.583±0.047	.534±0.012
banana	.460±0.008	.482±0.000	.450±0.003	.398±0.013	.483±0.005	**.660±0.048**	.473±0.013	.476±0.017	.484±0.003
baobab	.531±0.024	.617±0.000	.438±0.003	.440±0.015	.484±0.015	**.706±0.092**	.518±0.048	.500±0.014	.500±0.006

TABLE 3. The results are ARIs for different cluster ensemble algorithms, and the highest ARI among different algorithms on each data set is bolded.

	AK-means	MK-means	CSPA	HGPA	MCLA	SSSCE	EMcN	QMIc	SCE
beer	0.0007	0.0070	0.0333	0.0209	0.0458	**0.5977**	0.0336	0.0309	0.0267
congressEW	0.0532	0.5316	0.4280	0.0000	0.5382	0.5355	0.5447	0.3301	**0.5501**
aerosol	0.0005	0.0054	0.0062	0.0044	0.0014	**0.3523**	−0.0027	0.0000	0.0035
alph	0.0005	0.0055	0.0182	0.0087	0.0099	**0.4615**	0.0177	0.0070	0.0120
alphabet	0.0009	0.0087	0.0170	0.0133	0.0055	**0.3671**	0.0199	0.0206	0.0100
amber	0.0292	0.2916	0.1860	0.1656	0.2061	**0.8208**	0.25486	0.2439	0.2214
ambulances	0.0022	0.0215	0.0137	−0.0003	0.0608	**0.5083**	0.02679	0.0294	0.0569
americanflag	0.0056	0.0560	0.0576	−0.0047	0.0306	**0.2944**	0.0513	0.0408	0.0246
anonovo	0.0085	0.0853	0.0858	0.0107	0.0776	**0.3831**	0.0697	0.0872	0.0905
apple	0.0005	0.0051	0.0046	−0.0020	0.0067	**0.3652**	0.0055	0.0047	0.0040
seed	0.0617	0.6173	0.7039	0.2166	**0.7171**	0.6838	0.6210	0.5517	0.7166
aquarium	0.0000	0.0000	0.0160	−0.0014	−0.0200	**0.4258**	0.0053	0.0010	0.0420
arrow	0.0064	0.0639	0.0577	0.0095	0.0580	**0.3614**	0.0772	0.0648	0.0552
balance	0.0126	0.1259	0.1188	0.0389	0.1337	**0.4519**	0.1316	0.1132	0.1548
banana	0.0068	0.0681	0.0561	0.0152	0.0741	**0.5917**	0.0722	0.0766	0.0804
baobab	0.0111	0.1114	0.0510	0.0258	0.0739	**0.5311**	0.1085	0.1059	0.0848

Then we evaluate and calculate the accuracy of SSSCE by using ARI. Table 3 shows the results. Compared with other cluster ensemble algorithms, SSSCE can perform just as good or even better, while the ARI value of SCE is not higher, which shows the importance of selection of consensus function and semi-supervised learning.

6. **Conclusions.** In this paper we introduced the cluster ensemble problem. We applied SC algorithm to combining multiple base clusterings into a single consolidated clustering and took advantage of semi-supervised thought to modify the similarity matrix generated from the base clusterings. Experimental results on a very wide range of data sets show that SSSCE outperforms SCE, as well as other cluster ensemble algorithms in terms of accuracy. The direction of future work is how to place additional domain constraints to yield consensus solutions that are useful and actionable in diverse applications, for example, image segmentation.

Acknowledgment. This research was supported by the research subject of state science and technology support program under Grant 2015BAF32B05.

REFERENCES

[1] A. Strehl and J. Ghosh, Cluster ensembles – A knowledge reuse framework for combining multiple partitions, *Journal of Machine Learning Research*, vol.3, pp.583-617, 2002.

[2] J. Ghosh and A. Acharya, Cluster ensembles, *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol.1, no.4, pp.305-315, 2011.

[3] S. Vega-Pons and J. Ruiz-Shulcloper, A survey of clustering ensemble algorithms, *International Journal of Pattern Recognition and Artificial Intelligence*, vol.25, no.3, pp.337-372, 2011.

[4] H. G. Ayad and M. S. Kamel, Cumulative voting consensus method for partitions with variable number of clusters, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol.30, no.1, pp.160-173, 2008.

[5] H. Wang, H. Shan and A. Banerjee, Bayesian cluster ensembles, *Statistical Analysis and Data Mining*, vol.4, no.1, pp.54-70, 2011.

[6] J. Yi, T. Yang, R. Jin et al., Robust ensemble clustering by matrix completion, *IEEE the 12th International Conference on Data Mining*, pp.1176-1181, 2012.

[7] Y. N. Andrew, M. I. Jordan and Y. Weiss, On spectral clustering: Analysis and an algorithm, *Proc. of Conference on Neural Information Processing Systems*, Vancouver, Canada, pp.849-856, 2002.

[8] J. Jia, X. Xiao, B. Liu et al., Bagging-based spectral clustering ensemble selection, *Pattern Recognition Letters*, vol.32, no.10, pp.1456-1467, 2011.

[9] X. Z. Deng, L. C. Jiao and S. Lu, Spectral clustering ensemble applied to SAR image segmentation using nonnegative matrix factorization, *Acta Electronica Sinica*, vol.12, 2011.

[10] J. Jia, X. Xiao and B. Liu, Similarity-based spectral clustering ensemble selection, *The 9th International Conference on Fuzzy Systems and Knowledge Discovery*, pp.1071-1074, 2012.

[11] O. Chapelle, B. Schölkopf and A. Zien, *Semi-Supervised Learning (Adaptive Computation and Machine Learning)*, 2006.

[12] Z. Yu, L. Li, J. You et al., SC3: Triple spectral clustering-based consensus clustering framework for class discovery from cancer gene expression profiles, *IEEE/ACM Trans. Computational Biology and Bioinformatics*, vol.9, no.6, pp.1751-1765, 2012.

[13] J. R. Barr, K. W. Bowyer and P. J. Flynn, Framework for active clustering with ensembles, *IEEE Trans. Information Forensics and Security*, vol.9, no.11, pp.1986-2001, 2014.

[14] Y. Moazzen, B. Yalcin and K. Tasdemir, Sampling based approximate spectral clustering ensemble for unsupervised land cover identification, *IEEE International Geoscience and Remote Sensing Symposium*, 2015.

[15] M. H. C. Law, A. P. Topchy and A. K. Jain, Multi-objective data clustering, *Proc. of IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recog.*, vol.2, pp.424-430, 2004.

[16] F. Torres and J. Griffin, Control with micro precision in abrasive machining through the use of acoustic emission signals, *International Journal of Precision Engineering and Manufacturing*, vol.16, no.3, pp.441-449, 2015.

[17] M. Hoffman, D. Steinley and M. J. Brusco, A note on using the adjusted rand index for link prediction in networks, *Social Networks*, vol.42, pp.72-79, 2015.