Proceedings of the Twenty-Fourth International Florida Artificial Intelligence Research Society Conference

Consensus Clustering + Meta Clustering = Multiple Consensus Clustering

Yi Zhang and Tao Li
School of Computer Science
Florida International University
Miami, FL 33199

Abstract

Consensus clustering and meta clustering are two important extensions of the classical clustering problem. Given a set of input clusterings of a given dataset, *consensus clustering* aims to find a single final clustering which is a better fit in some sense than the existing clusterings, and *meta clustering* aims to group similar input clusterings together so that users only need to examine a small number of different clusterings. In this paper, we present a new approach, MCC (stands for multiple consensus clustering), to explore multiple clustering views of a given dataset from the input clusterings by combining consensus clustering and meta clustering. In particular, given a set of input clusterings of a particular data set, MCC employs meta clustering to cluster the input clusterings and then uses consensus clustering to generate a consensus for each cluster of the input clusterings. Extensive experimental results on 11 real world data sets demonstrate the effectiveness of our proposed method.

1 Introduction

Consensus/Ensemble clustering, also called as aggregation of clusterings or ensemble clustering, refers to the problem of finding a single (consensus) clustering from a number of different (input or base) clusterings that have been obtained for a particular dataset. Many different approaches have been developed recently to solve consensus clustering problem (Gionis, Mannila, and Tsaparas 2005; Strehl, Ghosh, and Cardie 2002; Li, Ding, and Jordan 2007; Hu et al. 2005). More recently, several approaches have also been proposed to select a subset of input clusterings to form a smaller but better performing cluster consensus than using all available solutions (Fern and Lin 2008; Azimi and Fern 2009). Typically, in these existing consensus clustering approaches, all the input clustering solutions or the selected subset of input clustering solutions are combined together to output a *single* consensus clustering of the data that is "better" than the existing clusterings, i.e., in this consensus clustering, clusters are better separated, or equivalently, the clustering objective functions are improved. There is, however, a significant drawback in generating a single consensus clustering. Recent studies have shown that in consensus clustering: 1) different input clusterings could differ significantly, and 2) subsets of input clusterings could be highly correlated (Li, Ding, and Jordan 2007; Azimi and Fern 2009; Caruana, Elhawary, and Nguyen 2006). When different input clusterings differ significantly, the consensus by simply averaging is really a brute-force voting and there is no real "consensus" in their original meaning. As a result, a single "consensus" may not be ideal in many cases and finding a single consensus clustering solution is not always the best way to explore hidden pattern structures for a given dataset (Caruana, Elhawary, and Nguyen 2006). Then, meta clustering is proposed to generate many alternative groups of good clusterings and allows the users to select the useful groups of clusterings (Caruana, Elhawary, and Nguyen 2006).

Real world datasets such as text and biology datasets are often multi-faceted with high dimensions. They can often be interpreted in many different ways and can have different clusterings that are reasonable and interesting from different perspectives (Caruana, Elhawary, and Nguyen 2006). In fact, in many datasets, clusters overlap substantially and natural clusters cannot be defined clearly. In general, a single (even the "best" if exists) clustering objective function can not effectively model the vast different types of datasets (Ding and He 2002). Therefore, it is interesting to explore multiple clustering views of a given data set. In addition, when the input clusterings differ significantly and constitute different groups, it is quite likely that the consensus formed by a certain group of input clusterings achieves better clustering performance than the consensus formed using all the input clusterings.

In this paper, we present a new approach MCC to explore multiple clustering views of a given dataset from a set of input clusterings by combining consensus clustering and meta clustering. Given a number of different (input) clusterings that have been obtained for a particular dataset, instead of generating a single consensus, our method first computes the pairwise similarities between input clusterings, and then organizes the different input clusterings into k groups where k is determined based on the spectral properties of the similarity matrix. Different from meta clustering that finds many alternate good clusterings of the data, our method generates consensus clusterings from the input clusterings of a given data set. Different from consensus clustering which finds a single consensus from the input clusterings, MCC groups the input clusterings and obtains multiple consensuses (a consensus for each group). In summary, This proposed approach brings two interrelated but distinct themes from clustering together: consensus clustering and meta clustering. Given a set of input clusterings of a particular data set, it first employs meta clustering to cluster the input clusterings and then uses consensus clustering to generate a consensus for each cluster of the input clusterings. Extensive experimental results on 11 real world data sets demonstrate the effectiveness of our proposed method.

The rest of the paper is organized as follows: Section 2 discusses the related work; Section 3 describes our proposed algorithm; Section 4 shows the experimental results on 11 real world data sets; and finally Section 5 concludes the paper.

2 Related Works

Consensus Clustering: Given multiple partitions generated by different clustering algorithms or different subsets of the dataset or different feature spaces, consensus clustering aims to ”combine” them into a single consolidated clustering that maximizes the agreement shared among all available clustering solutions and consequently obtains a better clustering solution (Gionis, Mannila, and Tsaparas 2005; Strehl, Ghosh, and Cardie 2002; Li, Ding, and Jordan 2007). Different from traditional consensus clustering, our MCC groups the input clusterings and obtains multiple consensuses (a consensus) for each group.

Meta Clustering: Meta clustering is proposed to generate many alternative good clusterings of the data and allows the users to select the useful clusterings (Caruana, Elhawary, and Nguyen 2006). In particular, meta clustering groups similar input clusterings together so that users only need to examine a small number of different clusterings. Different from meta clustering that finds many alternate good clusterings of the data, our MCC generates consensus clusterings from the input clusterings of a given data set.

Alternative Clustering: Recently, many techniques have been proposed to find alternative clusterings or multiple complementary clusterings (Cui, Fern, and Dy 2007; Qi and Davidson 2009). For example, Cui et al. presented a framework to find all non-redundant clusterings of the data where data points of one cluster can belong to different clusters in other views (Cui, Fern, and Dy 2007). Different from alternative clustering, our MCC aims to explore multiple clusterings views from the input clusterings of a given data set.

Recently, the combination of consensus clustering and meta clustering is proposed in (Zhang and Li 2011) where the different input clusterings are organized into a hierarchical tree structure and consensus clustering algorithm is applied to obtain a single consensus for the input clusterings in a subset of the hierarchical tree.

3 Methodology

3.1 An Overview of Multiple Consensus Clustering

An overview of our proposed MCC method is shown in Figure 1. MCC consists of the following 4 steps:

1. *Input Clusterings Generation* where different input (or base) clusterings are obtained by different clustering algorithms (with different parameters) on the original data set.
2. *Comparing Input Clusterings* where the pairwise similarity matrix of the input clusterings is calculated. (See Section 3.2.)
3. *Meta Clustering* where meta clustering is applied to group the input clusterings into k clusters and k is determined by the spectral model of the similarity matrix. (See Section 3.3.)
4. *Consensus Generation* where multiple consensuses can be generated by applying consensus clustering algorithms to the different groups in the flat partition. (See Section 3.4.)

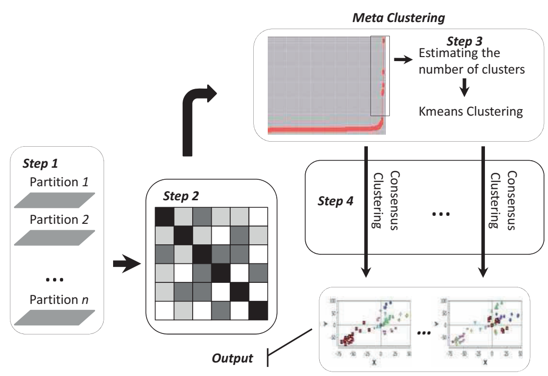

Figure 1: The general framework of multiple consensus clustering.

3.2 Similarity Matrix Calculation

In order to group the given input clustering results, the similarities between different clusterings need to be computed first. Several measures can be used for comparing different clusterings, such as pair counting (A, A, and I 2002), set matching (Dongen and Dongen 2000), and variation of information (Meila 2002). In our work, we compute the similarity based on the connectivity matrix.

Formally let $X = \{x_1, x_2, \cdots, x_n\}$ be a set of n data points. Given a partitional clustering P, consisting of a set of clusters $C = \{C_1, C_2, \cdots, C_k\}$ where k is the number of clusters and $X = \bigcup_{\ell=1}^{k} C_\ell$, we can define the following associated connectivity matrix $S(P)$:

$$S_{ij}(P) = \begin{cases} 1 & (i,j) \in C_\ell, i \neq j \\ 0 & \text{Otherwise} \end{cases}, \tag{1}$$

where $(i, j) \in C_\ell$ means that the i-th data point and the j-th data point are in the same cluster. In other words, if the i-th data point and the j-th data point are in the same cluster.

Given two clustering P^1 and P^2 of the data points in X, the similarity between them can be defined as follows

$$Sim(P^1, P^2) = \frac{2}{n \times (n-1)} \sum_{i=1}^{n} \sum_{j=i+1}^{n} \delta(S_{ij}(P^1), S_{ij}(P^2)), \tag{2}$$

where δ is Kronecker's delta function, i.e., $\delta(i,j) = 1$ if $i = j$ and $\delta(i,j) = 0$ otherwise. In other words, the similarity between P^1 and P^2 is the fraction of identical connective matrix entries.

3.3 Spectral Property of the Similarity Matrix

Given the similarity matrix S of input clusterings, how many possible groups in the input clusterings. For simplicity, suppose that the input clustering in S are ordered according to their group assignments, that is, the input clusterings in the first group appear first, the input clusterings in the second group appear next,..., and the input clusterings in the last group appear at the end. Such permutation does not change the spectral properties.

Note that input clusterings within each group are similar to each other while they are quite different from those in other groups. Hence, S can be represented by $S = W + E$ where W is a block diagonal similarity matrix with constant within-block similarities, and E is a matrix with a small value in each entry. We have the following property of the similarity matrix.

Proposition 3.1 *(Fallah, Tritchler, and Beyene 2008) Given $S = W + E$ as described above. If there are k groups in the input clusterings, then S has k large (in absolute value) eigenvalues. Specifically, there are k eigenvalues of magnitude $O(n)$ and $n - k$ eigenvalues of order $O(n^{1/2})$, where n is the number of input clusterings.*

Hence in our work, we use the spectral property of the similarity matrix to determine the number of groups for the input clusterings. After determining the number of groups, we then perform K-means clustering to obtain the individual groups of input clusterings.

3.4 Consensus Generation

After obtaining the partition of the input clusterings, a consensus clustering algorithm can be applied to obtain a single consensus for the input clusterings in the same cluster of the partition. Here we briefly describe the consensus clustering algorithms used in our MCC framework.

1. **PCA-based consensus algorithm**: This algorithm firstly performs dimensionality reduction on the input clusterings using Principal Component Analysis (PCA), then applies the recursive bisection algorithm to do consensus clustering (Asur, Ucar, and Parthasarathy 2007).
2. **CSPA** (cluster-based similarity partitioning algorithm): A clustering signifies a relationship between objects in the same cluster and can thus be used to establish a measure of pairwise similarity. This induced similarity measure is then used to recluster the objects, yielding a combined clustering (Strehl and Ghosh 2003).
3. **HGPA** (HyperGraph partition algorithm): This algorithm approximates the maximum mutual information objective with a constrained minimum cut objective. Essentially, the cluster consensus problem is posed as a partitioning problem of a suitably defined hypergraph where hyperedges represent clusters (Strehl, Ghosh, and Cardie 2002).
4. **WClustering** (Weighted ensemble clustering) (Li and Ding 2008; Zhang et al. 2009): In this algorithm, each input clustering is weighted and the weights are determined in such way that the final consensus clustering provides a better quality solution.

4 Experiments

4.1 Dataset Description

Firstly, we describe the data sets used in our experiments. The characteristics of the data sets are summarized in Table 1. The number of classes ranges from 2 to 10, the number of samples ranges from 47 to 4199, and the number of dimensions ranges from 4 to 1000. Further details are described below:

Datasets	**# sample**	**# Dimensions**	**# Class**
Glass	214	9	7
Ionosphere	351	34	2
Iris	150	4	3
Soybean	47	35	4
Wine	178	13	3
Zoo	101	18	7
CSTR	475	1000	4
Log	1367	200	8
LetterIJL	227	16	3
Reuters	2900	1000	10
WebKB4	4199	1000	4

Table 1: Descriptions of the real-world datasets.

1. Seven datasets (Glass, Ionosphere, Iris, Soybean, Wine, Zoo, LetterIJL) are from UCI data repository (Blake and Merz 1998). LetterIJL is a randomly sampled subset of three $\{$I,J,L$\}$ from Letters dataset.
2. Four datasets (CSTR, Log, Reuters, WebKB4) are standard text datasets that are often used as benchmarks for document clustering. The document datasets are preprocessed (removing the stop words and unnecessary tags and headers) using the rainbow package (McCallum 1996).
3. CSTR is the dataset of the abstracts of technical reports published in Computer Science departments between 1991 and 2002. The dataset contains 476 abstracts, which are divided into four research areas: Natural Language Processing(NLP), Robotics/Vision, Systems, and Theory.
4. The Log dataset contains 1367 text messages of system log from different desktop machines describing the status of computer components. These messages are divided into 8 different situations.
5. The Reuters dataset is a subset of the Reuters 21578 Text Categorization Test collection containing the 10 most frequent categories among the 135 topics.
6. The WebKB dataset contains webpages gathered from university computer science departments. There are about 8280 documents and they are divided into 7 categories: student, faculty, staff, course, project, department and other. The WebKB4 dataset is the subset of WebKB associating with four most populous entity-representing categories, i.e., student, faculty, course and project (Han et al. 1998).

4.2 Performance Comparison

All the above datasets come with labels. Viewing these labels as the indicators of a reasonable clustering, we define the following accuracy measure (Li and Ding 2006):

$$Accuracy = \max\Big(\sum_{C_k, L_m} T(C_k, L_m)\Big)/n, \qquad (3)$$

where n is the number of data points, C_k denotes the k-th cluster, L_m is the m-th class and $T(C_k, L_m)$ is the number of data points that belong to class m and are assigned to cluster k. Accuracy is thus computed as the maximum sum of $T(C_k, L_m)$ for all pairs of clusters and classes, and these pairs have no overlap. In the experiments, the base clusterings are obtained by running K-means 30 times.

Dataset	D1	D2
Glass	0.614	0.515
Ionosphere	0.521	0.432
Iris	0.447	0.398
Soybean	0.661	0.522
Wine	0.478	0.314
Zoo	0.411	0.401
CSTR	0.589	0.544
Log	0.651	0.526
LetterIJL	0.549	0.433
Reuters	0.681	0.376
WebKB4	0.583	0.472

Table 3: Diversity measurement.

4.3 Consensus Diversity Measurement

Since our method can generate multiple consensuses, we also measure the diversity of the different consensuses. Given a set of n consensuses $\{C_1, \cdots, C_n\}$, let $ARI(C_i, C_j)$ and $NMI((C_i, C_j)$ denote the adjusted Rand index (Rand 1971) and normalized mutual information (Fred and Jain 2003) between two consensuses P_i and P_j. We use the following two measures to compute their diversity:

$$D1 = \frac{2}{n(n-1)} \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} (1 - ARI(C_i, C_j)), \qquad (4)$$

$$D2 = \frac{2}{n(n-1)} \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} (1 - NMI(C_i, C_j)). \qquad (5)$$

Note that $D1$ and $D2$ measure the pair-wise consensus similarity using adjusted rand Index and normalized mutual information respectively. The larger the measures, the more diverse the consensuses are.

4.4 Results Analysis

We compare the performance of our method with other single consensus clustering algorithms including K-means on the consensus similarity matrix (KC) (Li and Ding 2008), the NMF-based consensus clustering (NMFC) (Li, Ding, and Jordan 2007), the cluster-based similarity partitioning algorithm (CSPA) (Strehl, Ghosh, and Cardie 2002), the HyperGraph Partitioning Algorithm (HGPA) (Strehl, Ghosh, and Cardie 2002), and the weighted consensus clustering algorithm (WC) (Li and Ding 2008). Since the proposed method generates multiple consensuses, we use HighMCC, AvgMCC and LowMCC to denote the highest accuracy, the average accuracy, and the lowest accuracy of the multiple consensuses generated by our method, respectively. The experiment comparisons are shown in Table 2. We also include the results of K-means on the original datasets as the baselines. From the comparison, we observe that 1) consensus clustering algorithms generally improve K-means clustering on all the datasets except Iris, and 2) HighMCC has the best performance on all the datasets. And it verifies that a consensus obtained from a group of clusterings might have better performance than the consensus obtained from all the input clusterings. Table 3 presents the diversity measurement of the multiple consensuses generated by our framework.

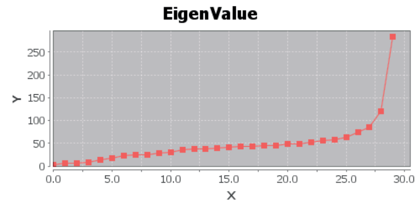

Figure 2: The spectrum of the similarity matrix for the input clusterings of the Glass dataset. X-axis is the index of 30 eigenvalues sorted in an increasing order from the left to right, and Y-axis presents the absolute value of the eigenvalues.

	C1	C2	C3	T
C1	1	0	0.255	0.647
C2	0	1	0.292	0.539
C3	0.255	0.292	1	0.551
T	0.647	0.539	0.551	1

Figure 3: Similarity comparisons of three clustering views.

4.5 Multiple Clustering Views Exploration

Then, we use the Glass dataset as an example to show how effective our approach is in presenting multiple clustering views. Firstly, from Figure 2, by examining the spectrum of the similarity matrix (i.e., three largest eigenvalues), we can obtain three distinct group of clusterings. After consensus clustering, Figure 3 displays the similarities of three final generated clustering results (C1, C2, C3) and the ground truth (T). The similarities between different clusterings are computed using Equation 2. From Figure 3, it could be observed that each input clustering has a relative small distance with the true labels, but with a higher distance among themselves. In other words, the three clustering views have good quality values and they are quite different from each other. To further verify the effectiveness of our approach in exploring meaningful multiple views, the performance of C1,

Datasets	Kmeans	KC	CSPA	HGPA	NMFC	WC	HighMCC	AvgMCC	LowMCC
Glass	38%	45%	43%	40%	49%	49%	61%	60%	58%
Ionosphere	70%	71%	68%	52%	71%	71%	72%	67%	62%
Iris	83%	72%	79.29%	86%	69%	89%	88%	83%	72%
Soybean	72%	82%	70%	81%	89%	91%	93%	86%	70%
Wine	68%	68%	69%	52%	70%	72%	74%	66%	60%
Zoo	61%	59%	56%	58%	62%	70%	70%	65%	61%
CSTR	45%	37%	50%	62%	56%	64%	69%	57%	50%
Log	61%	77%	47%	43%	71%	69%	77%	64%	61%
LetterIJL	49%	48%	48%	53%	52%	52%	52%	49%	47%
Reuters	45%	44%	43%	44%	43%	44%	45%	45%	45%
WebKB4	60%	56%	61%	62%	64%	63%	63%	62%	61%

Table 2: The experiment comparisons on 11 data sets.

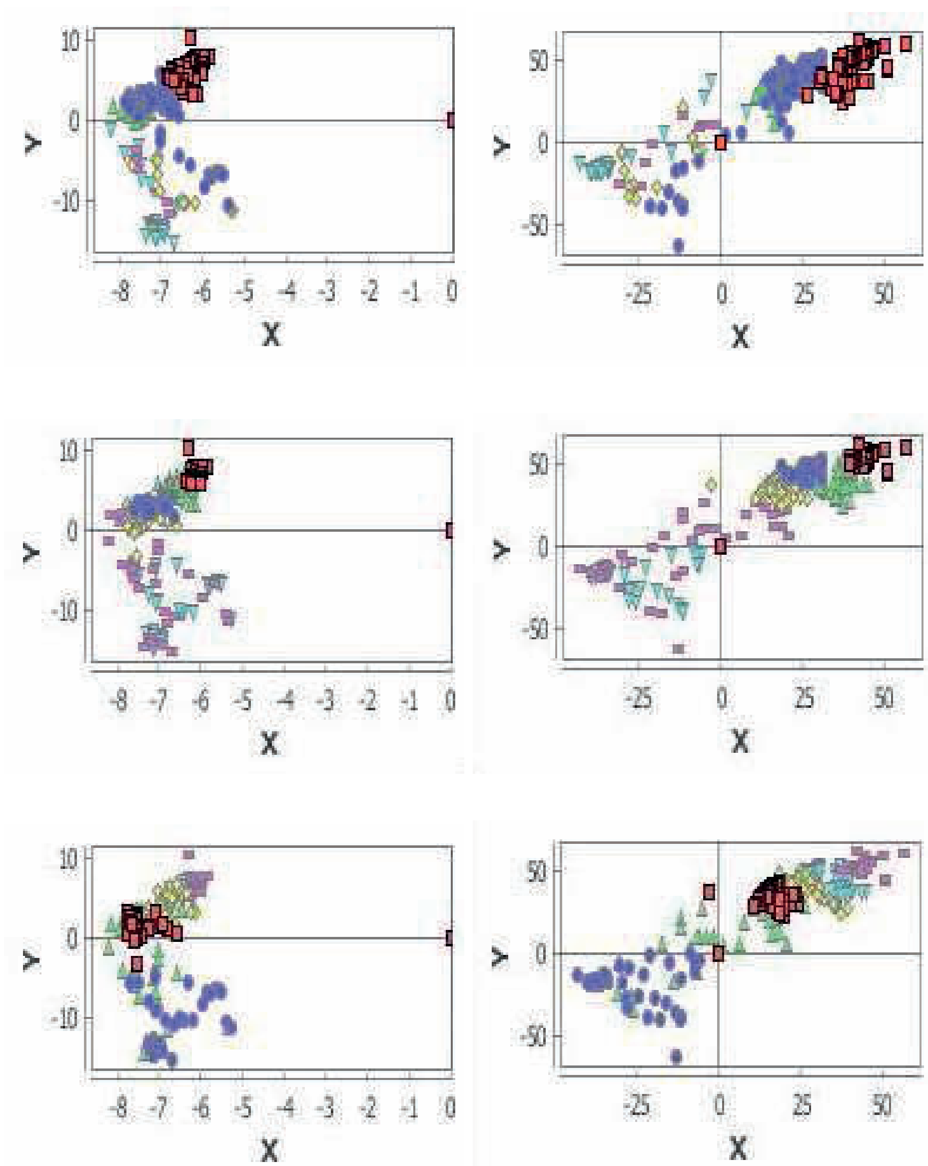

Figure 4: Multiple clustering views for the Glass dataset. The left and right views are generated by PCA and MDS respectively.

Evaluation Method	Clustering1	Clustering2	Clustering3
Accuracy	61.49%	58.28%	61.49%
NMI	51.91%	41.92%	53.95%
RandIndex	68.39%	64.32%	68.82%

Table 4: Multiple Clustering Performance Comparison

C2, C3 is verified in Table 4.4 based on accuracy, NMI, and rand index. In addition, to help users understand the clustering structures, Figure 4 shows the 2D-mapping plots of the Glass data set using Principal Component Analysis (PCA) and Multi-Dimensional Scaling (MDS), where each point is colored and shaped by cluster indicators obtained from different clustering views. We can clearly observe that the clustering views are quite different.

5 Conclusion

In this paper, we explore multiple clustering views by combining meta clustering and consensus clustering. Given a set of input clustering results, our proposed approach first applies meta clustering to group inputs into k clusters where k is determined by estimating the number of the significant eigenvalues of the similarity matrix of the input clusterings. Then, it uses consensus clustering to generate meaningful multiple clustering views. Experiments show the effectiveness of our proposed method.

6 Acknowledgment

The work is partially supported by NSF grants DBI-0850203 and HRD-0833093.

References

A, B.-H.; A, E.; and I, G. 2002. A stability based method for discovering structure in clustered data. *Pacific Symposium on Biocomputing* 7:6–17.

Asur, S.; Ucar, D.; and Parthasarathy, S. 2007. An ensemble framework for clustering protein-protein interaction networks. *Bioinformatics* 23(13):i29–i40.

Azimi, J., and Fern, X. 2009. Adaptive cluster ensemble selection. In *International Joint Conference on Artificial Intelligence*, 993–997.

Blake, C. L., and Merz, C. J. 1998. UCI Repository of Machine Learning Databases.

Caruana, R.; Elhawary, M.; and Nguyen, N. 2006. Meta clustering. In *ICDM*.

Cui, Y.; Fern, X. Z.; and Dy, J. 2007. Non-redundant multi-view clustering via orthogonalization. In *ICDM*, 133–142.

Ding, C., and He, X. 2002. Cluster merging and splitting in hierarchical clustering algorithms. In *ICDM*, 139–146.

Dongen, S. V., and Dongen, S. V. 2000. Performance criteria for graph clustering and markov cluster experiments. Technical Report INS-R0012, National Research Institute for Mathematics and Computer Science.

Fallah, S.; Tritchler, D.; and Beyene, J. 2008. Estimating number of clusters based on a general similarity matrix with application to microarray data. *Journal of Statistical Applications in Genetics and Molecular Biology* 7(1).

Fern, X. Z., and Lin, W. 2008. Cluster ensemble selection. *Journal of Statistical Analysis and Data Mining* 1(3):128–141.

Fred, A. L., and Jain, A. K. 2003. Robust data clustering. In *CVPR*, volume 2, 128–133.

Gionis, A.; Mannila, H.; and Tsaparas, P. 2005. Clustering aggregation. In *ICDE*, 341–352.

Han, E.-H.; Boley, D.; Gini, M.; Gross, R.; Hastings, K.; Karypis, G.; Kumar, V.; Mobasher, B.; and Moore, J. 1998. Webace: a web agent for document categorization and exploration. In *AGENTS*, 408–415.

Hu, X.; Yoo, I.; Zhang, X.; Nanavati, P.; and Das, D. 2005. Wavelet transformation and cluster ensemble for gene expression analysis. *International Journal of Bioinformatics Research and Applications* 1(4):447 – 460.

Li, T., and Ding, C. 2006. The relationships among various nonnegative matrix factorization methods for clustering. In *ICDM*, 362 – 371.

Li, T., and Ding, C. 2008. Weighted consensus clustering. In *SDM*, 798–809.

Li, T.; Ding, C.; and Jordan, M. I. 2007. Solving consensus and semi-supyervised clustering problems using nonnegative matrix factorization. In *ICDM*, 577–582.

McCallum, A. K. 1996. Bow: A toolkit for statistical language modeling, text retrieval, classification and clustering. http://www.cs.cmu.edu/ mccallum/bow.

Meila, M. 2002. Comparing clusterings. Technical report, Statistics, University of Washington.

Qi, Z., and Davidson, I. 2009. a principled and flexible framework for finding alternative clusterings. In *SIGKDD*, 717–726.

Rand, W. M. 1971. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association* 66(336):846–850.

Strehl, A., and Ghosh, J. 2003. Relationship-based clustering and visualization for high-dimensional data mining. *INFORMS Journal on Computing* 15(2):208–230.

Strehl, A.; Ghosh, J.; and Cardie, C. 2002. Cluster ensembles - a knowledge reuse framework for combining multiple partitions. *Journal of Machine Learning Research* 3:583–617.

Zhang, Y., and Li, T. 2011. Extending consensus clustering to explore multiple clustering views. In *SDM*.

Zhang, Y.; Zeng, E.; Li, T.; and Narasimhan, G. 2009. Weighted consensus clustering for identifying functional modules in protein-protein interaction networks. In *ICMLA*, 539–544.