Information Fusion for Semi-supervised Cluster Labelings

Huaying Li and Aleksandar Jeremic*
Department of Electrical and Computer Engineering, McMaster University
Hamilton, Ontario, Canada

Keywords: Clustering, Information Fusion, Cluster Ensemble and Semi-supervised Learning.

Abstract: Clustering analysis is a widely used technique to find hidden patterns of a data set. Combining multiple clustering results into a consensus clustering (cluster ensemble) is a popular and efficient method to improve the quality of clustering analysis. Many algorithms were proposed in the literature and most of which are unsupervised learning techniques. In this paper, we proposed a semi-supervised cluster ensemble algorithm. It is so-called semi-supervised because labels of some data points in the given data set are known or provided by experts. To evaluate the performance of the proposed algorithm, we compare it with other well-known algorithms, such as MCLA and BCE.

1 INTRODUCTION

Mutation is accidental changes in genomic sequence of DNA (Pickett, 2006). Studies of mutation are usually completed using fluorescence microscopy, an important tool for visualizing biochemical activity within individual cells. In the past, analysis of these images was done manually through visual inspection, which sometimes leads to a time consuming and inaccurate conclusion. Nowadays, automated image analysis techniques are developed, such as high content analysis (HCA). It is the use of automated microscopy and high end computation to understand the complex biological processes. It typically involves acquiring high resolution images and translating them into a multi-dimensional feature space, which spans hundreds of features per fluorescence channel and will be further processed to provide relevant output (Shariff et al., 2010). Cluster analysis is a widely used technique to find the hidden patterns or structure of a data set. The objective is to divide data points into distinct clusters so that data points in the same cluster are similar to each other and data points in different cluster are dissimilar. Cluster analysis is also one popular machine learning technique to further process data obtained from feature extraction step of HCA.

Although there are many clustering algorithms exist in the literature, such as hierarchical, centroid-based, distribution-based and graph theory-based algorithms, no single algorithm can correctly identify underlying structure of all data sets in practice (Xu and Wunsch, 2008). It is usually difficult to decide which algorithm should be applied for a given data set when prior information about the cluster shape and size are not provided. Furthermore, for a particular clustering algorithm, it usually generates different clustering labels for a given data set by choosing different initial start points or different parameter setting of the algorithm. Combing multiple clusterings into a consensus labeling is a hard problem because of two reasons: (1) number of clusters in each clustering could be different and the desired number of clusters is usually unknown; (2) cluster labels are symbolic so there is also a correspondence problem. In (Vega-Pons and Ruiz-Shulcloper, 2011), the authors provide a detailed review of many existing algorithms: some algorithms are based on relabeling and voting; some are based on co-association matrix; Some are based on graph and hypergraph representation of clusterings; some are based on finite mixture models and etc. All of these algorithms are unsupervised learning because input data set is unlabeled and clusters are not pre-defined. Also, most of cluster ensemble algorithms consists of two major steps: cluster ensemble generation and consensus fusion.

In this paper, we propose a semi-supervised cluster ensemble method. The term semi-supervised means labels of some data points are available and are utilized in the fusion stage. In next section we briefly describe the cluster ensemble problem and introduce the structure of multiple clustering system. In the following section, we propose a semi-supervised algorithm for combing multiple clusterings. Then we provide several numerical examples to illustrate the

*This work was supported by Natural Sciences and Engineering Research Council of Canada.

Li H. and Jeremic A..
Information Fusion for Semi-supervised Cluster Labelings.
DOI: 10.5220/0005282203180323
In *Proceedings of the International Conference on Bio-inspired Systems and Signal Processing* (BIOSIGNALS-2015), pages 318-323
ISBN: 978-989-758-069-7

algorithm and evaluate performance of the algorithm using different types of data sets. Finally, we give a conclusion in the last section.

2 CLUSTER ENSEMBLE PROBLEM

For a given data set, it depends on its characteristic to choose an appropriate clustering algorithm with some suitable parameter settings. It also depends on the purpose of the use of the clustering results. Therefore, utilizing cluster analysis techniques to obtain multiple clusterings and combining them into a consensus clustering is an efficient way to improve the quality of clustering. The objective is to obtain a consensus clustering containing the most information from each individual clustering. In this section, we first introduce some notation to describe the cluster ensemble problem and then define our multiple clustering system.

Let $X = \{x_1, x_2, \ldots, x_N\}$ denote a set of N data points and each data point x_n (for $n = 1, \ldots, N$) comes from a F-dimensional feature space. Clustering technique is used to partition the data set into some smaller sets, denoted as k clusters, in the way that data points in a cluster are more similar to each other than to those in different clusters. A clusterer Φ represents the function of generating a clustering result, which is stored in a N-dimensional label vector. The idea of cluster ensemble is represented in Fig. 1. For a given data set X, a clusterer $\Phi^{(j)}$ is used to partition the data into k_j clusters and cluster labels are stored in $\lambda^{(j)}$, where $j = 1, 2, \ldots, M$. A set of clusterings $\Lambda = \{\lambda^{(1)}, \lambda^{(2)}, \ldots, \lambda^{(M)}\}$ is generated by alternating the clustering algorithm M times in each clusterer or choosing different parameter settings (i.e. choosing different numbers of clusters for each clusterer or choosing different initial cluster centers and etc.). Multiple clusterings are fused later by a consensus function Γ in order to obtain a single label vector λ, a more reliable partition of the given data.

In Fig 1, arrows on the left represent the generation step and arrows on the right represent the consensus step. On the one hand, there is usually no constrains on how to generate the multiple clusterings. There are several possibilities in the generation process: using different clustering algorithms; using the same clustering algorithm with different initializations and/or different parameter settings; using subsets of features of data points and using random projections on different subspaces. On the other hand, the consensus step is the core step to obtain a consolidated single clustering result. In this paper, we propose an algorithm that is based on relabeling and voting. Different from the existing algorithms working with unlabeled data, our proposed algorithm deals with data set that is partially labeled. We call it semi-supervised cluster ensemble algorithm.

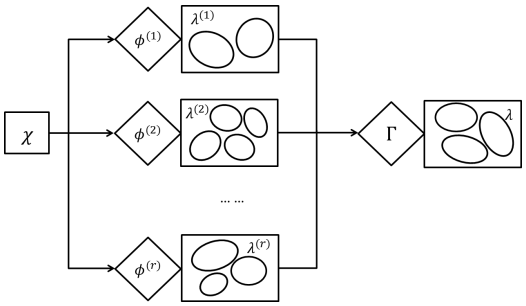

Figure 1: Multiple Clustering System.

3 PROPOSED ALGORITHM

In this section, we propose our semi-supervised algorithm on information fusion of multiple clusterings. It contains three major steps: pre-clustering, cluster ensemble generation and consensus fusion.

3.1 Pre-clustering

Usually in every field of study, opinions from experts are available but limited due to many factors. Our assumption is that for a given data set experts or resources are available to provide future cluster labels for a certain portion of all data points. In the pre-clustering step, we randomly select $p\%$ (p is a predetermined number) of data points in X to be reference points. Data set X is thus divided into two subsets: reference set and unknown set. Reference set X_r contains reference data points and unknown set X_u consists of the rest of data points called undetermined points. Experts analyze on reference set X_r and group the reference points into clusters. We call these clusters as reference clusters. These clustering labels of reference points are useful later in consensus fusion and we call them reference labels. Suppose the number of reference clusters is k_0. Reference set X_r could be divided into k_0 sub-sets: $\{X_r^1, \ldots, X_r^{k_0}\}$, where X_r^k (for $k = 1, \ldots, k_0$) contains the data points in the k-th reference cluster. Since reference points are randomly selected, we assume the number of desired clusters for the whole data set X should be consistent with the number of reference clusters k_0.

3.2 Cluster Ensemble Generation

Different clustering algorithms provide different cluster labels for the same data set since they focus on different aspects of data (Topchy et al., 2004). Due to its simplicity, k-means algorithm is a widely used algorithm to provide individual clustering in generation step of cluster ensemble algorithm. By choosing a number k quite larger than the expected number of clusters, k-means algorithm is capable to divide data points into k smaller groups. Such a small group of data pints usually is able to capture some details of the structure of the entire data set, while some of these smaller groups may need to be merged together to form a cluster because their common properties. In (Fred and Jain, 2005), cluster ensemble is generated by running k-means algorithm multiple times with random initializations. The number of clusters for each run is randomly selected from a set of integers (much greater than k_0). We used a similar generation mechanism to obtain cluster ensemble in this paper. For data set X (reference and unknown sets together), k-means algorithm is applied M times to generate M individual clusterings, which form a N-by-M label matrix Λ. Entry of Λ on the i-th row and j-th column $\Lambda_{i,j}$ is the cluster label of x_i according to j-th clustering. In previous pre-clustering step, data set X is divided into k_0+1 subsets: k_0 reference clusters and an unknown set (i.e. $X = \{X_r^1, X_r^2, \ldots, X_r^{k_0}, X_u\}$). Accordingly, matrix Λ could be segmented into k_0+1 parts: $\Lambda_r^1, \Lambda_r^2, \ldots, \Lambda_r^{k_0}, \Lambda_u$.

3.3 Consensus Fusion

Consensus fusion of multiple clusterings is the core step of the proposed algorithm. The fusion idea is stated as follow: according to an individual clustering, count the number of agreements between label of a data point in unknown set and labels of reference points in each reference clusters; assign this data point the corresponding cluster label which has the highest number of agreements; repeat the procedure for all the clusterings and determine the final cluster label based on some fusion rule. The summary of the proposed algorithm is stated in Table 1.

Suppose for $k = 1, \ldots, k_0$ R^k is the number of reference points in the k-th reference cluster and R is the total number of reference points. Thus, $R = R^1 + R^2 + \cdots + R^{k_0}$ and the total number of undetermined points is $N - R$. For the i-th undetermined data point x_i and the j-th clustering $\lambda^{(j)}$ (where $i = 1, \ldots, N-R$ and $j = 1, \ldots, M$), the association vector $\mathbf{a_{ij}}$ contains k_0 entries, each of which describes the association of x_i and a reference cluster.

Table 1: Semi-supervised clustering ensemble algorithm.

1. Pre-clustering
   (a) Choose $p\%$ of data points and obtain reference labels $(1, \ldots, k_0)$
2. Cluster Ensemble Generation
   (a) Apply clusterer $\Phi^{(j)}$ to data set X and obtain individual clustering $\lambda^{(j)}$
   (b) Repeat M times to form a label matrix $\Lambda = \{\Lambda_r^1, \Lambda_r^2, \ldots, \Lambda_r^{k_0}, \Lambda_u\}$
3. Consensus Fusion
   (a) Assign undetermined data points their most associated cluster ids (highest entry in association vector) according to label vector λ_j. Association vector is computed by
   $$\mathbf{a_{ij}}(k) = \frac{\text{occurrence of } \Lambda_u(i,j) \text{ in } \Lambda_r^k(:,j)}{\#\text{of points in } k\text{th reference cluster}}$$
   (b) Repeated M times to form new submatrix Λ_u'
   (c) Apply fusion rule to obtain consensus clustering

Recall that for the undetermined data points the corresponding segment of label matrix Λ is Λ_u. Fusion rule, such as majority voting, is difficult to apply directly to Λ_u due to the correspondence problem of cluster labels. A new matrix is necessary in order to apply fusion rule to generate the consensus labels. Based on the relabeling scheme we described above, according to a clustering $\lambda^{(j)}$, assign undetermined data points their most associated cluster labels (highest entry in the corresponding association vector) and repeat M times to form a new matrix Λ_u'. In this new label matrix, the correspondence problem is removed by utilizing the reference labels. We could apply any fusion rule to obtain the consensus clustering. In this paper, we use plurality voting scheme to generate the final consensus label.

4 ALGORITHM EXTENSION FOR LARGE DATA SETS

Our proposed algorithm requires number of reference labels is sufficient (i.e the ratio of the number of reference points and the size of data set is greater than a certain percentage $p\%$). Due to the fact that expertise or resource is usually expensive and limited, the proposed algorithm is only suitable for data set with a moderate size. For a large data set, we pro-

Table 2: Extended version of Semi-supervised clustering ensemble algorithm.

- Pre-clustering: Choose $p\%$ of data points and obtain reference labels $(1,\ldots,k_0)$
- If $\frac{R}{N} \geq p\%$, do step 2) and 3) in TABLE 1 to obtain λ_u
- If $\frac{R}{N} < p\%$, $Q = \lceil \frac{p}{1-p} \cdot \frac{N-R}{R} \rceil$
  - for $q = 1 : Q$
    $X^q = \{X_r, X_u^q\}$, where X_u^q is the q-th subset of X_u
    do step 2) and 3) in TABLE 1 on X^q to obtain λ_u^q
    end
  - $\lambda_u = \{\lambda_u^1, \lambda_u^2, \ldots, \lambda_u^Q\}$

Table 3: Data sets from UCI machine learning repository.

Data Set	Data Points	Features	Classes
Ionosphere	351	34	2
Pima	768	8	2
Balance	625	4	3
Wine	178	13	3
Segmentation	2100	19	7

pose to divide the undetermined set X_u into several smaller sub-sets and to apply the proposed algorithm on each subset combined with the reference set. Recall that N is the total number of data points in X and R is the number of reference points. If $\frac{R}{N} < p\%$, we divide the undetermined set X_u into Q sub-sets, i. e. $X_u = \{X_u^1, X_u^2, \ldots, X_u^Q\}$, where $Q = \lceil \frac{p}{1-p} \cdot \frac{N-R}{R} \rceil$. Function $\lceil x \rceil$ is the ceil function: the smallest integer not less than x.

For $q = 1, \ldots, Q$, denote the combination of reference data set and the q-th subset of X_u as a new data set $X^q = \{X_r, X_u^q\}$. Apply the semi-supervised cluster ensemble algorithm to X^q and generate consensus clustering λ_u^q and repeat it Q times. Combine the Q segments together to form the overall consolidated clustering of the whole undetermined set X_u, where $\lambda_u = \{\lambda_u^1, \lambda_u^2, \ldots, \lambda_u^Q\}$. The summary of the extended version of proposed algorithm is stated in Table 3.

5 EXPERIMENTAL RESULTS AND DISCUSSION

In this section, we provide several numerical examples to show the performance of the proposed algorithm. UCI machine learning repository website provides hundreds of data sets for machine learning researchers (Bache and Lichman, 2013). We choose five data sets from their website to evaluate the performance of our proposed algorithm. The information about the chosen data sets are listed in Table 3.

Table 4: Maximum micro-precision of proposed algorithm with selected percentages of reference.

Maximum	$p = 5\%$	$p = 10\%$	$p = 15\%$
Ionosphere	0.8034	0.8917	0.9345
Pima	0.6523	0.6914	0.7630
Balance	0.7456	0.7648	0.7872
Wine	0.4888	0.6404	0.7022
Segmentation	0.7652	0.7881	0.8262
Maximum	$p = 20\%$	$p = 25\%$	$p = 30\%$
Ionosphere	0.9345	0.9402	0.9459
Pima	0.7617	0.7773	0.7878
Balance	0.8096	0.8032	0.8240
Wine	0.6966	0.7640	0.7865
Segmentation	0.8129	0.8214	0.8367

Table 5: Average micro-precision of proposed algorithm with selected percentages of reference.

Average	$p = 5\%$	$p = 10\%$	$p = 15\%$
Ionosphere	0.7929	0.8721	0.9248
Pima	0.5997	0.6837	0.7510
Balance	0.7162	0.7304	0.7496
Wine	0.4640	0.6163	0.6702
Segmentation	0.7543	0.7719	0.8078
Average	$p = 20\%$	$p = 25\%$	$p = 30\%$
Ionosphere	0.9262	0.9319	0.9239
Pima	0.7436	0.7638	0.7809
Balance	0.7778	0.7862	0.8008
Wine	0.6478	0.7433	0.7584
Segmentation	0.8057	0.8110	0.8284

Since in our experiments testing data sets have true labels associated with them, we choose micro-precision (mp) as our metric to measure the accuracy of a clustering result with respect to the true clustering (Wang et al., 2011). Suppose there are k_t classes in truth for a given data set X with N data points. Suppose N_k is the number of data points in the k-th cluster of a clustering result that are correctly assigned to the corresponding class. Corresponding class here represents the true class that has the largest overlap with the k-cluster. The micro-precision is defined by $mp = \sum_{k=1}^{k_t} N_k/N$. As mentioned at the beginning of this paper, many cluster ensemble algorithms exist in the literature. We compared our algorithm withMCLA and CSPA proposed in (Strehl and Ghosh, 2003) and MM and BCE presented in (Wang et al., 2011). The authors of (Vega-Pons and Ruiz-Shulcloper, 2011) provide a brief review and the core idea of these algorithms. True cluster labels of the data sets listed in Table 3 are available throughout UCI website.

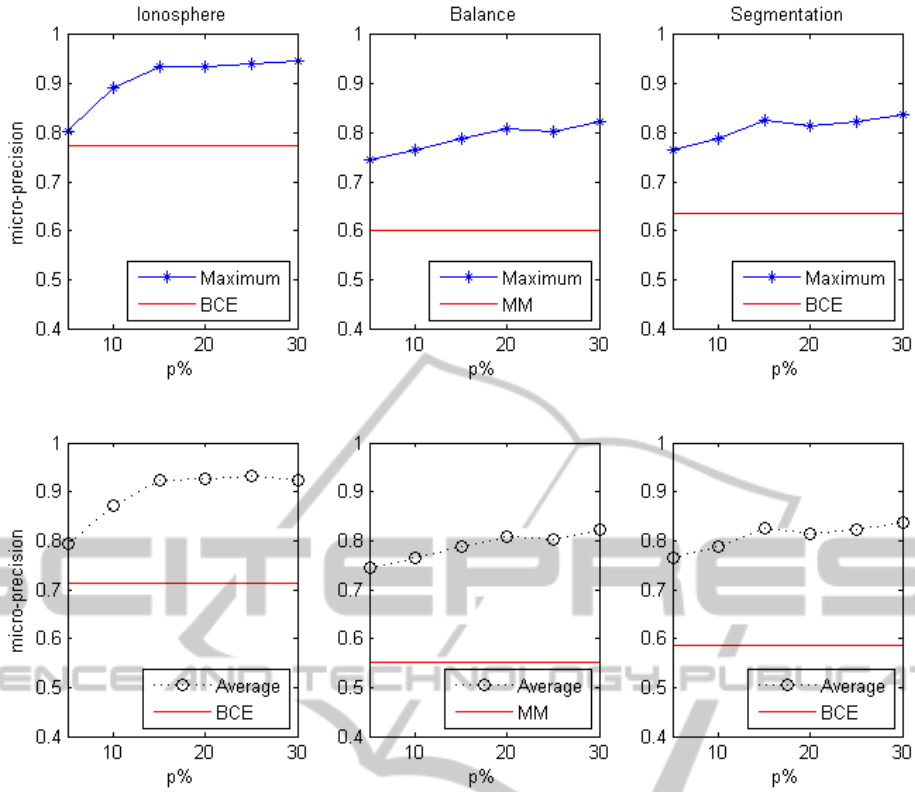

Figure 2: Micro-precision of proposed algorithm compared with other algorithms Part I.

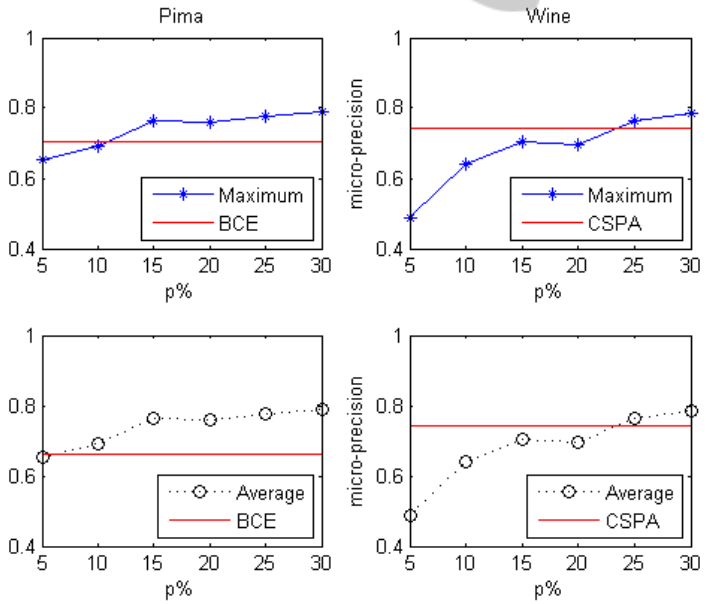

Figure 3: Micro-precision of proposed algorithm compared with other algorithms Part II.

In our experiments, we randomly select $p\%$ of data points and use the corresponding true labels of these points as reference labels where $M = 15$ is used to generate the ensemble. We start with $p = 5$ and increase the percentage of reference points with a 5% increment. The maximum and average performance of our proposed semi-supervised cluster ensemble algorithm are listed in Table 4 and Table 5 respec-

tively. Compared with experimental results reported in (Wang et al., 2011), Figure 2 and Figure 3 show that the proposed algorithm could provide higher micro-precisions in most of our experiments. In the figures each column represents one data set. The upper plots show the maximum performance while the lower plots show the average performance. The x-axis represents the percentage of reference points and y-axis displays the micro-precision of the proposed algorithm. The more reference points used, the better performance obtained. There is a horizontal line in each plot, which represents the corresponding best micro-precision reported in (Wang et al., 2011). For data sets Ionosphere, Balance and Segmentation, using only 5% of reference labels could generate a consensus clustering with much higher micro-precision. For data sets Pima and Wine, increasing the amount of reference points is able to generate a more precise consensus clustering.

We also apply the proposed algorithm to a biomedical data set which was obtained using Perkin Elmar high content imaging system. The data is used to study human breast cancer cells undergoing treatement of different drugs. In our experiment, data points are from four different treatments. As preliminary results, our proposed algorithm is able to label 75% of data points correctly by using 5% of reference data points and 86% correctly by using 20% of reference.

REFERENCES

Bache, K. and Lichman, M. (2013). UCI machine learning repository.

Fred, A. L. and Jain, A. K. (2005). Combining multiple clusterings using evidence accumulation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 27(6):835–850.

Pickett, J. P. (2006). *The American heritage dictionary of the English language*. Houghton Mifflin.

Shariff, A., Kangas, J., Coelho, L. P., Quinn, S., and Murphy, R. F. (2010). Automated image analysis for high-content screening and analysis. *Journal of biomolecular screening*, 15(7):726–734.

Strehl, A. and Ghosh, J. (2003). Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *The Journal of Machine Learning Research*, 3:583–617.

Topchy, A. P., Jain, A. K., and Punch, W. F. (2004). A mixture model for clustering ensembles. In *SDM*, pages 379–390. SIAM.

Vega-Pons, S. and Ruiz-Shulcloper, J. (2011). A survey of clustering ensemble algorithms. *International Journal of Pattern Recognition and Artificial Intelligence*, 25(03):337–372.

Wang, H., Shan, H., and Banerjee, A. (2011). Bayesian cluster ensembles. *Statistical Analysis and Data Mining*, 4(1):54–70.

Xu, R. and Wunsch, D. (2008). *Clustering*, volume 10. John Wiley & Sons.