Clustering Learning Objects Collections Using Cluster Ensembles

Hanan Ayad and Mohamed Kamel

Pattern Analysis and Machine Intelligence Group
University of Waterloo, Waterloo, Ontario, Canada
hanan@pami.uwaterloo.ca and mkamel@pami.uwaterloo.ca

Abstract

Learning Object Repositories are increasingly being used in learning systems to provide high-quality, reusable educational materials. A relevant data mining problem associated with the automatic categorization of learning objects is the discovery of intrinsic classes based on the textual contents of the meta-data records. In this paper, we present a cluster ensemble method, that is applicable for distributed clustering of learning objects, and for overcoming issues of large-sized collections and very high dimensionality of the vocabulary space. First, an initial significant reduction of the vocabulary space is proposed. Subsequently, a cluster ensemble method, consisting of three stages is proposed. Base clusterings for multiple random subsets of the data are generated, followed by the application of an adaptive voting algorithm for resolving the cluster label mapping problem. Finally, a consensus clustering is extracted by merging similar clusters that may be produced by fine-resolution base clusterings, and maximum likelihood principle is applied on a resulting assignment probability matrix. The experimental analysis shows improvement in the consensus clustering quality over the base clusterings, and a competitive performance compared to clustering the original collection, as measured using normalized mutual information. The analysis validates the applicability of the proposed ensemble method.

1. Introduction

Data clustering produces a segmentation of a collection of objects into a set of homogeneous clusters, enabling the discovery of intrinsic classes, the extraction of meaningful summary of data (e.g. topics), and efficient access of information in large collections of objects. In real-world data, particularly large collections of text-based objects, it is difficult to accurately identify clusters that match a meaningful classification determined by domain-experts. A fundamental difficulty stems from the complexity of the desired cluster structures, which may include elaborate classification hierarchies, highly unbalanced cluster sizes, varying cluster densities, and degrees of overlaps among classes. Another major difficulty is due to the very high-dimensionality of the data, which is usually an order of magnitude larger than the size of the data, leading to a significant problem in *machine learning* known as the *curse of dimensionality*, where objects represents isolated points in a vast empty space.

Cluster ensembles [14, 6, 7, 8, 9, 4, 5, 2, 1, 3, 10, 16, 15, 12, 17] have recently emerged as a method to improve clustering performance by combining multiple clusterings, also known as *base clusterings or partitions*, using a combining algorithm, which is referred to as a *consensus function*. Cluster ensembles have been proposed for finding arbitrarily-shaped clusters [6, 7, 8, 9, 2, 1], for reducing clustering instability due to noise or outliers [5], to improve the robustness of randomized clustering algorithms [6], for reusing pre-existing clusterings (knowledge reuse) [14], for exploring random feature subspaces [14, 15] and random projections [15] in high dimensional data, for estimating confidence in various cluster assignments [4], and for clustering in distributed environments using feature or object distributed clustering [14]. Several applications have been investigated including DNA micro-array data [4], distributed data mining [14, 11], and image segmentation [5].

Here, we explore the performance of a proposed cluster ensemble method for clustering learning objects collections. Based on a preliminary analysis, we first propose a filtering of the dimensionality that significantly reduces the vocabulary size while slightly improves the clustering quality. Subsequently, We apply the proposed cluster ensemble method, whereby multiple base clusterings are generated on random subsets of the data collection. This cluster ensemble generation method is referred to in the literature as object-distributed clustering and is a useful approach for distributed data mining applications [14], where a data collection is distributed among multiple sites. It is also noted that clustering a subset of the data instead of the entire col-

lection, at a time, automatically reduces the dimensionality (in this case, the vocabulary size) required to represent the data. The second stage of the proposed cluster ensemble method addresses the problem of cluster label mismatch caused by the lack of correspondence between the cluster labels of each of the base clusterings. A mapping is performed based on a "winner-takes-all" voting technique. Finally, an aggregation technique, in conjunction with the maximum likelihood principle, are applied for the extraction of the final consensus clusters.

Section 2 describes the learning objects dataset used for testing the proposed cluster ensemble method. Section 3 presents the details of the proposed cluster ensemble method. Section 4 presents the experimental study and Section 5 discuss the conclusions.

2. Learning Objects Collection

The proposed cluster ensemble method is applied on the Canada SchoolNet Metadata, which consists of 2371 metadata records collected from the"Curriculum Area" of the Canada's SchoolNet website (http://www.schoolnet.ca) [1]. SchoolNet uses an extended set of the Dublin Core metadata element set. The words in the "title", "description" and "keywords" metadata fields are used to represent the records in a vector space representation, which is suitable for the application of general pattern recognition techniques. Stopword removal and stemming are performed, and the *Term Frequency - Inverse Document Frequency* (TFxIDF) scheme [13] for term weighting is applied on the unique words occurring in combined fields (title, description and keywords) from all metadata records.

2.1. Data Representation

The dataset is represented in a record $\times$ terms matrix format, denoted $\mathbf{X}$, and of size 2371×7166. Each record, denoted $\mathbf{x}_i$ is represented as a vector in a d-dimensional feature space, $\mathbf{x_i} \in \Re^d$. The formulation of the clustering problem consider $(X1, \ldots, X_d)$ denote a $1 \times d$ as a random vector of d variables, or features, and $Y \in \{1, ..., k\}$ as the unknown cluster label. Given a sample of Xs, the goal is to determine the cluster label for each object.

A clustering of the records is represented by a labeling vector $\mathbf{y} \in \mathcal{C}^n$, and $\mathcal{C} = \{c_1, \ldots, c_k\}$ where each clustered record is assigned a cluster label $y_i \in \{c_1, \ldots, c_k\}$, where k is a pre-specified number of clusters, such that $1 < k < n$. Records that may not have been clustered (due to sub-sampling), are assigned a cluster label 0. A hard clustering or a *partition*, where each clustered object is assigned to one and only one cluster, can be represented by a binary assignment indicator matrix [14]. Such matrix, denoted as $\mathbf{U}$, has a row for each cluster, and a column for each object, such that, $u_{ij} \in \{0, 1\}$, and $\sum_{i=1}^{k} u_{ij} = 1$, for each clustered object $\mathbf{x_j}$.

A soft clustering of the data assigns each object to each cluster by an assignment probability or a degree of membership. The representation of a soft clustering is a generalization of the binary assignment matrix $\mathbf{U}$, consisting of the estimated posterior probabilities of cluster membership (cluster assignment probabilities) $\hat{P}(c_i|x_j)$. That is, $u_{ij} \in [0, 1]$, $\sum_{i=1}^{k} u_{ij} = 1$. Maximum likelihood principle can be applied to extract a hard clustering from a probabilistic clustering by assigning each observation to the cluster with the maximum posterior probability, and an estimated maximum likelihood probability vector $\hat{\mathbf{p}}$ can be generated.

2.2. Data Classification

The Canada SchoolNet Metadata records are categorized into a finite set of classes designated by the "subject" metadata field. The classification structure is a *forest*, i.e., a collection of trees. Each tree represents a hierarchical classification of a set of subjects. The are 15 root subjects (i.e. trees), and a total of 150 individual subjects (nodes) in the entire classification structure. In the experimental analysis presented here, we use the root subjects as the desired clusters. Considering the 150 subjects is a more challenging problem since, on one hand, many of these subjects have significant similarities among them, and many have very few records. On the other hand, the relationships between subjects and their ancestors require the development of techniques that can capture the complexity of the hierarchy. While this is an interesting research direction in our future work, it is outside the scope of the presented work.

3. Proposed Cluster Ensemble Method

The cluster ensemble method has three stages, the generation of the base clusterings, the mapping of cluster labels for resolving the label mismatch problem, and the extraction of the consensus clusters. The details of each of these stages are presented below.

3.1. Generation of Base Clusterings

Multiple subsets of the collection of learning objects are randomly sampled. A portion $\alpha \in [0, 1]$ of the size of the collection ($n = 2371$) is used as the sample size. We experiment with various portion values. An efficient center-based algorithm (the k-means), with the cosine measure of similarity, is applied on each of the random data samples to produce the base clusterings. We experiment with various

[1] The data was collected and pre-processed by K. Hammouda from the integration team of the LORNET Theme 4, PAMI Group.

number of clusters k, that is larger than or equal to the final desired number K of consensus clusters.

Let $\mathcal{U} = \{\mathbf{U}^1, \ldots, \mathbf{U}^b\}$, denote a cluster ensemble consisting of a set of b base clusterings. A consensus function $\Phi(\mathcal{U}, K) = \hat{\mathbf{y}}$ combines the cluster ensemble, into one consensus clustering denoted $\hat{\mathbf{y}}$, and consisting of K consensus clusters. The proposed consensus function estimate a probabilistic consensus clustering denoted by $\hat{\mathbf{U}}$, which is used to compute the maximum likelihood clustering $\hat{\mathbf{y}}$, and the corresponding probability vector $\hat{\mathbf{p}}$.

3.2. Adaptive Mapping of Cluster Labels

An adaptive voting algorithm is developed for mapping cluster labels. The algorithm, referred to here as `Ada-MaxVote`, adopts a "winner-takes-all" voting strategy. Algorithm 1 shows the details of the proposed `Ada-MaxVote` algorithm. The algorithm takes as input an ensemble $\mathcal{U}$, and processes it sequentially. It uses a reference clustering denoted $\mathbf{U}^0$. The algorithm generates an empirical cluster assignment probability matrix, denoted by $\bar{\mathbf{U}}$, with k columns and n rows. The matrix $\bar{\mathbf{U}}$ is seen as the "averaged" assignment probabilities of the objects, for the matched clusters. In order to match clusters, a measure of similarity among clusters is required. We use the Jaccard measure, which was previously proposed in [14] for similarity computation between pairs of clusters. The first base clustering is used to initialize the reference clustering matrix. Each cluster, in each subsequent base clustering, "votes" for the cluster label of its most similar reference cluster, and the reference clustering is updated by incrementing the the corresponding members of the matched clusters. The output matrix $\bar{\mathbf{U}}$ is computed by normalizing the last reference clustering, so that the sum of each column is equal to 1.

Algorithm 1 The `Ada-MaxVote` algorithm

Function $\bar{\mathbf{U}} =$ `Ada-MaxVote` $(\{\mathbf{U}^1, \ldots, \mathbf{U}^b\})$
1: $\mathbf{U}^0 \leftarrow \mathbf{U}^1$ {Initialize the reference clustering}
2: **for** $i = 2$ to b **do**
3: $\quad \mathbf{W}^i \leftarrow \mathbf{0}$ {Initialize a $k \times k$ binary mapping matrix $\mathbf{W}$}
4: $\quad$ **for** $j = 1$ to k **do**
5: $\quad\quad$ **for** $l = 1$ to k **do**
6: $\quad\quad\quad s_{jl} =$ `Jaccard`$(\mathbf{u}^i_j, \mathbf{u}^0_l)$
7: $\quad\quad$ **end for**
8: $\quad\quad m = \arg_l \max(s_{jl})$
9: $\quad\quad w^i_{mj} = 1$ {Set values in mapping matrix $\mathbf{W}$}
10: $\quad$ **end for**
11: $\quad \mathbf{U}^0 = \mathbf{V} + \mathbf{W}^i \times \mathbf{U}^i$ {Update votes in the voting matrix}
12: **end for**
13: $\bar{\mathbf{U}} = \tilde{\mathbf{U}}^0$ {Compute cluster assignment probability matrix}

3.3. Extracting the Consensus Clusters

If the number of clusters k in the base clusterings is equal to the desired number of consensus clusters K, the maximum likelihood principle can be directly applied on the cluster assignment probability matrix $\bar{\mathbf{U}}$ to extract the consensus clustering, as given in Equation 1.

$$\hat{y}_j = \arg_{1 \le i \le k} \max\{\hat{u}_{ij}\}, \quad \hat{p}_j = \max_{1 \le i \le k} \{\hat{u}_{ij}\}, \qquad (1)$$

where $\hat{\mathbf{U}} = \bar{\mathbf{U}}$, and $1 \le j \le n$.

On the other hand, when $k > K$, we apply the hierarchical average-link clustering algorithm to merge the k clusters into K consensus clusters. The similarity measure used is again the Jaccard measure. The estimated assignment probability matrix $\hat{\mathbf{U}}$ is computed by averaging the assignment probabilities corresponding to the merged clusters, for each object. The consensus clusters are also extracted using the maximum likelihood principle described in Equation 1.

4. Experimental Analysis

4.1. Clustering Quality

For Measuring the quality of the consensus clustering, we used the normalized mutual information (NMI) proposed in [14]. The NMI criterion provides a measure of agreement between two different clusterings $\mathbf{y}^{(1)}$, and $\mathbf{y}^{(2)}$, and is given as.

$$NMI(\mathbf{y}^{(1)}, \mathbf{y}^{(2)}) = \frac{\sum_{i=1}^{k}\sum_{j=1}^{k} n_{i,j} \log\left(\frac{n \times n_{i,j}}{n_i^{(1)} \times n_j^{(2)}}\right)}{\sqrt{\left(\sum_{i=1}^{k} n_i^{(1)} \log\frac{n_i^{(1)}}{n}\right)\left(\sum_{j=1}^{k} n_{(j)}^{2} \log\frac{n_j^{(2)}}{n}\right)}} \qquad (2)$$

where $n_i^{(1)}$, is the number of objects assigned the cluster label c_i in $\mathbf{y}^{(1)}$, $n_j^{(2)}$ is the number of objects assigned the cluster label c_j in $\mathbf{y}^{(2)}$, and $n_{i,j}$ is the number of objects assigned the cluster label c_i in $\mathbf{y}^{(1)}$, and c_j in $\mathbf{y}^{(2)}$. The NMI measure takes value between 0 and 1, with higher values indicating better clustering quality.

4.2. Vocabulary Space

A preliminary cluster analysis using the kmeans algorithm on the original matrix of 2371 records $\times$ 7166 terms, with $k = 15$, reveals a clustering with a normalized mutual information (NMI) of 0.40 with respect to the true classification. Since the data has a very large vocabulary space, we conduct an initial analysis of the relationship between the

term weights and the vocabulary size. The sum of weights of each terms over all the records are computed, which we refer to here as, the sum-of-weights criterion. We compute the vocabulary size if a threshold based on this criterion is used to filter out terms from the vocabulary space. Figure 1 shows a plot of the vocabulary size over the range of the sum-of-weights criterion. The curve indicates a rapid decrease in the vocabulary size at a very low and narrow interval of threshold values. Notably, the vocabulary size drops from 7166 terms less than 2000 in approx $1/100$ of the range of the sum-of-weights. That is, the sum-of-weights criterion can be used for aggressive reduction of the dimensionality since a significant number of terms appears to have a very low overall weight. Furthermore, when the rate of change of the curve starts to slow, this indicates that fewer terms seem to exhibit higher overall weight values. Hence, we choose a threshold at such a transition stage from a sharp to a relatively slower rate of change. The first such occurrence is close to threshold value of 0.85, corresponding to a vocabulary size of 1883. A clustering of the 2371×1883 matrix using the kmeans algorithm has a slightly higher NMI of 0.42, compared to the clustering of the original matrix. Hence, the analysis based on sum-of-weight criterion led to an initial significant reduction of the dimensionality while preserving the clustering quality (with slight improvement).

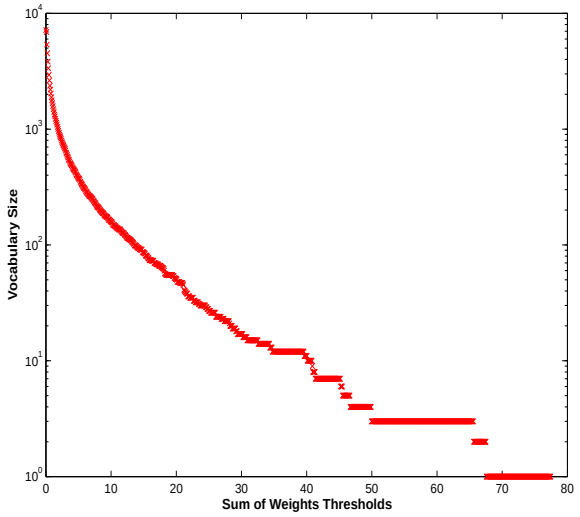

Figure 1. Sum of terms versus number of terms in Vocabulary. Y-axis is log-scale

4.3. Experimental Results

Table 1 shows the results of the experiments. The ensemble parameters are the ratio of the sample size to total number of records, denoted α, the number of clusters in each of the base clusterings k (in the experiments reported in this paper, a fixed value of k is use within an ensemble), and the third parameter is the size of the ensemble size b. We select a variety of arbitrary values for each of those parameters. As a general guideline, as α decreases, it would be desirable to increase b so as to reduce the chance of records not occurring in any of the samples. It would also be desirable, as α decrease, to also decrease k so as to avoid an over fragmentation of the clusters of records.

Table 1. Clustering quality using NMI

Parameters			Base Clusterings			Consensus
α	k	b	Max	Average	Min	
0.5	15	10	0.25	0.23	0.22	0.39
0.5	20	20	0.28	0.25	0.23	0.40
0.5	25	20	0.28	0.26	0.23	0.40
0.5	25	30	0.28	0.26	0.23	0.43
0.5	25	40	0.27	0.26	0.23	0.45
0.5	25	50	0.28	0.26	0.23	0.42
0.5	30	10	0.28	0.27	0.26	0.40
0.5	30	20	0.27	0.26	0.24	0.39
0.5	45	10	0.29	0.28	0.26	0.37
0.3	15	50	0.19	0.17	0.15	0.42
0.3	15	100	0.19	0.17	0.15	0.44
0.3	20	50	0.20	0.18	0.16	0.38
0.25	15	50	0.17	0.15	0.13	0.42
0.25	15	60	0.18	0.15	0.13	0.40
0.25	15	100	0.17	0.15	0.12	0.43
0.25	20	40	0.18	0.16	0.15	0.40
0.25	20	50	0.19	0.16	0.14	0.40
0.2	15	50	0.15	0.13	0.11	0.37
0.2	15	100	0.15	0.13	0.11	0.42
0.15	15	50	0.13	0.11	0.09	0.36
0.15	15	100	0.13	0.11	0.09	0.42

The results in Table 1 show very low NMI values for the base clusterings with respect to the true classification of the entire collection. This is due to the fact that only a portion of the records are clustered. While each base clustering contributes a partial clustering due to "restricted accessibility", the consensus clustering aggregates multiple partial views and constructs a clustering for the entire collection improving the quality which is indicated by the higher NMI values appearing in the last column of Table 1.

5. Conclusion

The effective dissemination of learning object repositories requires an accompaniment by advances in related content analysis tools. Cluster ensembles provide a new paradigm for addressing issues related to data distribution, large collection sizes, very high-dimensional feature spaces, as well as complex class structures.

In this paper, we demonstrated the applicability of a proposed cluster ensemble method for addressing some of the issues related to content analysis for learning object collections, by testing it on the Canada SchoolNet meta-data records. In the proposed cluster ensemble method, base clusterings were generated efficiently on multiple random subsets of the collection. While an initial significant reduction of the vocabulary space was performed, the dimensionality is further reduced simultaneously as each base clustering "focuses" on a small portion of the data, thus dividing the overall size of the original clustering problem.

The experimental results showed that the proposed cluster ensemble method successfully aggregates multiple partial clusterings into a single combined clustering for the entire collection, that is of competitive quality compared to clustering the original substantially larger data. In addition, the combined clustering significantly improves the quality of the partial base clusterings.

Furthermore, the adaptive nature and low computational complexity of the proposed voting algorithm ($O(k^2bn)$) allows for on-line aggregation of newly generated clusterings into the previously estimated cluster assignments probabilities. The integration of newly generated clusterings into the extracted consensus clustering is facilitated by the low computational complexity of the extraction method, which is linear in the number of objects n and quadratic in the number of clusters k, where k is normally a small integer such that $k \ll n$.

The consequences of the findings of this work for the application of clustering techniques on learning object repositories can be summarized as follows. First, it is possible through the proposed aggressive reduction of the vocabulary space to achieve a gain in the computation time without sacrificing the quality of the clustering (a slight gain in quality was actually achieved). Secondly, instead of clustering an entire collection of learning objects at once, competitive results can be achieved by clustering multiple random subsets of the collection and combining them into a single consensus clustering. A further advantage of this approach is the simpler representation (low dimensionality and small size) of each subset compared to the entire set of learning objects. Finally, the proposed combining method can be applied if a collection of learning objects is distributed among multiple sites.

As a future direction, we believe that an interesting, yet challenging problem, which needs to be addressed is the development of cluster ensemble methods that model complex classification structures such as the collection of subject hierarchies underlying the Canada SchoolNet collection. It seems reasonable to assume that such hierarchies are natural structures for the classification of learning object repositories, which motivates the development of methods that enable the discovery of these types of structures.

References

[1] H. Ayad, O. Basir, and M. Kamel. A probabilistic model using information theoretic measures for cluster ensembles. In *Multiple Classifier Systems: Fifth International Workshop, MCS 2004, Cagliari, Italy, Proceedings.*, pages 144–153, 2004.

[2] H. Ayad and M. Kamel. Finding natural clusters using multi-clusterer combiner based on shared nearest neighbors. In *Multiple Classifier Systems: Fourth International Workshop, MCS 2003, UK, Proceedings.*, pages 166–175, 2003.

[3] H. Ayad and M. Kamel. Cluster-based cumulative ensembles. In *Multiple Classifier Systems: Sixth International Workshop, MCS 2005, Seaside, CA, USA, Proceedings.*, pages 236–245, 2005.

[4] S. Dudoit and J. Fridlyand. Bagging to improve the accuracy of a clustering procedure. *Bioinformatics*, 19(9):1090–1099, 2003.

[5] B. Fischer and J. Buhmann. Bagging for path-based clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(11):1411–1415, 2003.

[6] A. Fred. Finding consistent clusters in data partitions. In J. Kittler and F. Roli, editors, *Multiple Classifier Systems, 3rd International Workshop on Muliple Classifier Systems MCS 2001, LNCS 2096*, pages 309–318. Springer, 2001.

[7] A. Fred and A. Jain. Data clustering using evidence accumulation. In *Proceedings of the 16th International Conference on Pattern Recognition. ICPR 2002*, volume 4, pages 276–280, Quebec City, Quebec, Canada, August 2002.

[8] A. Fred and A. Jain. Evidence accumulation clustering based on the k-means algorithm. In T.Caelli, A. Amin, R. Duin, M. Kamel, and D. de Ridder, editors, *Structural, Syntactic, and Statistical Pattern Recognition, volume LNCS 2396*, pages 442–451. Springer-Verlag, 2002.

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

[10] L. I. Kuncheva and S. Hadjitodorov. Using diversity in cluster ensembles. In *IEEE International Conference on Systems, Man and Cybernetics, Proceedings*, pages 1214–1219, The Hague, The Netherlands., 2004.

[11] S. Merugu and J. Ghosh. Privacy-preserving distributed clustering using generative models. In *IEEE Int'l Conf. on Data Mining (ICDM03)*, pages 211–218, Melbourne, FL, Nov 2003. AAAI/MIT Press.

[12] B. Minaei, A. Topchy, and W. Punch. Ensembles of partitions via data resampling. In *IEEE Intl. Conf. on Information Technology: Coding and Computing, ITCC04, Proceedings*, volume 2, pages 188–192, Las Vegas, April 2004.

[13] G. Salton and M. J. McGill. *Introduction to modern information retrieval.* McGraw-Hill, 1983.

[14] A. Strehl and J. Ghosh. Cluster ensembles - a knowledge reuse framework for combining multiple partitions. *Journal of Machine Learning Research (JMLR)*, 3:583–617, December 2002.

[15] A. Topchy, A. Jain, and W. Punch. Combining multiple weak clusterings. In *IEEE Intl. Conf. on Data Mining 2003, Proceedings*, pages 331–338, Melbourne, Fl., November 2003.

[16] A. Topchy, A. Jain, and W. Punch. A mixture model of clustering ensembles. In *SIAM Conf. on Data Mining*, pages 379–390, April 2004.

[17] A. Topchy, A. Jain, and W. Punch. Clustering ensembles: Models of consensus and weak partitions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(12):1866–1881, 2005.