

A Novel Multi-View Agglomerative Clustering Algorithm Based on Ensemble of Partitions on Different Views

Hamidreza Mirzaei

School of Computing Science, Simon Fraser University, Burnaby, Canada
hmirzaei@cs.sfu.ca

Abstract

In this paper, we propose a new algorithm for extending the hierarchical clustering methods and introduce a Multi-View Agglomerative Clustering approach to handle multi-view represented objects. Experiments on real world datasets indicate that our algorithm considering the relationship among multiple views can provide a solution with improved quality in multi-view setting. We find empirically that the multi-view version of our Agglomerative Clustering, independent of linkage method and given any number of views, greatly improves on its single-view counterparts.

1. Introduction

In traditional clustering approaches there was an assumption that data objects are independent and of identical class, and are often modeled by a fixed-length vector of feature values. In many real life problems multi-view data arise naturally. Multi-view data are instances that have multiple representations (views) from different feature spaces. Usually these multiple views are from different vector spaces or different graph spaces or a combination of vector and graph spaces [1]. The most typical examples are web pages, which can be classified based on their content as well as based on the anchor texts of inbound hyperlinks. Using the established clustering methods for clustering multi-represented data requires to restrict the consideration to a single representation or to construct a feature space combining all representations. However, the restriction to a single feature space would not consider all available information and the construction of a combined feature space demands great care when constructing a combined distance

function. Most existing clustering approaches cluster the multi-types of objects individually even when they are highly interrelated [2].

Multi-view clustering plays an important role in a wide range of classification and clustering [3,4,5]. Indeed multi-view learning has been introduced by two important papers, Yarowsky [6] and Blum et al. [3]. In [6] an algorithm for word sense disambiguation has been proposed. The co-training method proposed in [7] is also a classic multi-represented semi-supervised clustering. The multi-view version of the Expectation maximization algorithm for semi-supervised learning has been proposed in [8,9] as co-EM algorithm.

The work reported in this paper is motivated by the following observations. In [10] the multi-view versions of popular data mining methods, including partitioning methods (such as k-Means, k-Medoids, and EM) and hierarchical, agglomerative methods have been studied. Although K-Means and EM have been shown to have a great improvement on their single-view counterparts, by contrast, negative results for agglomerative hierarchical multi-view clustering have been achieved. To address this issue, in this work we have proposed a novel multi-view agglomerative clustering. This approach overcomes the shortcoming of the single-view version on clustering multi-represented objects.

The organization of the paper is as follows. The proposed approach is discussed in Section 2. Section 3 presents the experimental results and finally we conclude the work and offer some future works in Section 4.

3. Proposed Method

In this paper we extend hierarchical clustering algorithms and introduce a Multi-View Agglomerative Clustering approach to handle multi-view clustering.

We consider the problem that data is split into some subsets in which the objective is to extract a combined clustering model over these different subsets (views). We want to investigate multi-view algorithms to obtain improvements in terms of cluster entropy, i.e. based on the impurity of a cluster given the true mixture components of the data, over their single-view counterparts. In order to make a combined model, in contrast to the approach which uses all the views simultaneously and in turn [10], in our approach, given different views, we first extract a dendrogram for each view independently, then we combine these dendrograms to derive the final clustering. Combining a set of dendrograms directly is a computationally expensive problem, so we explore the usefulness of using an intermediate matrix representation of dendrograms to facilitate the combination process.

Once a dendrogram is formed, it can be represented by many different matrices. One method is the use of cophenetic distance [11,12,13]. The cophenetic description matrix for a cluster tree contains the cophenetic distances among different observations. The cophenetic distance between two observations is represented in a dendrogram by the height of the link at which those two observations are first joined where the height is the distance between the two subclusters merged by that link. Therefore, in this step, we use the concept of cophenetic distance in order to represent each dendrogram and descriptor matrices of the hierarchical cluster trees can be generated based on cophenetic matrices. Afterwards, each representation matrix is normalized to the range [0, 1] to prevent bias:

$$NDD^{(i)} = \{NDD_{k,l}^{(i)}\} \quad (1)$$

$$\text{where } DD^{(i)} = \{DD_{k,l}^{(i)}\} \text{ and } NDD_{k,l}^{(i)} = \frac{DD_{k,l}^{(i)}}{\text{Max}(DD_{m,n}^{(i)})}$$

indicate the extracted Dendrogram Descriptor from the i^{th} view and the normalized version of it, respectively. Now, it is desirable to develop a way to aggregate these separate descriptors to produce the final dendrogram for clustering the whole data. There are several aggregators for doing this and in our study we tested lots of them. Expert or prior knowledge can be used to determine the way to combine the descriptors.

Thus far, we have obtained the final dissimilarity matrix which can be used to generate the dendrogram in order to cluster the whole dataset. At the end, it is very straightforward to create a hierarchical cluster tree from the distances in this distance or dissimilarity matrix. The pseudo code of the proposed algorithm is shown in Table 1. This code is based on agglomerative

hierarchical clustering. The agglomerative algorithm begins with each element as a separate cluster and then iteratively merges the closest clusters and builds up the dendrogram. Different agglomerative clustering algorithms differ in the way they define the distances between clusters. Single linkage is a method based on the best pair of all possible pairs which means the minimum dissimilarity. On the other hand the complete linkage is based on the worst pair which means maximum dissimilarity. Finally, with the group average method, linkage is based on the average of the dissimilarities [13].

In our study we applied various aggregators such as maximum of primary matrices or average of them on different kinds of linkage methods. We noticed that changing linkage method of the base dendrograms and also final dendrogram has a negligible or even no impact on the results we obtain from combining the descriptors. In other words it's a prominent feature of our algorithm that allows different linkage methods, for example single-linkage, complete-linkage and finally average-linkage without any substantial change in the outcome. The other noticeable characteristic of our approach is that it simply handles the clustering problems with objects having more than two views. There is no difference having lots of views and the aggregation is easily done on their descriptors.

It is worth mentioning that in the phase of extracting descriptors, other definitions such as partition membership divergence matrix, cluster membership divergence and sub-tree membership divergence of the hierarchical tree were examined and in general, we found the cophenetic correlation to be much more robust. All of these methods are easy to implement and we have implemented them into our multi-view clustering algorithm.

4. Experimental Results

We implemented the proposed clustering algorithm in Matlab and ran experiments on several datasets. One good dataset for multi-view clustering can be found in [14]. This dataset contains set of features for the task of clustering the web pages into spam and non-spam. The features are split into different views and we have used direct features such as number of pages and length, as the view 1, link features such as in-degree, out-degree, PageRank and edge reciprocity as the view 2, and finally content-based features such as number of words in the home page, average word length, average length of the title, etc. as the view 3.

Table 1: Multi-view hierarchical clustering

Input: Unlabeled data: $D = \{(x_1^{(1)}, \dots, x_1^{(m)}), \dots, (x_n^{(1)}, \dots, x_n^{(m)})\}$

Distance measures: $d^1(C_i, C_j), \dots, d^m(C_i, C_j)$

- 1 Initialize $C_i = x_i, i = 1, \dots, n$.
- 2 For $v = 1, \dots, m$:
 - 3 For $t = 1, \dots, n$:
 - (a) Find pair of closest clusters (C_i, C_j)
 - (b) Merge C_i and C_j
- 4 $DD^{(v)}$ = Cophenetic matrix of v^{th} view
- 5 $NDD^{(v)} = \frac{DD^{(v)}}{\max_{k,l} (DD_{k,l}^{(v)})}$
- 6 Combined Descriptor = Appropriate aggregation of All Dendrogram Descriptors
- 7 Final Dendrogram = Linkage and Make the Tree from Combined Descriptor
- 8 Return Final Dendrogram

In practice, first some transformations must be applied on the features in order to achieve better results than the raw features. This includes numeric transformation of features mostly ratios between features such as Indegree/PageRank or TrustRank/PageRank, and $\log(\cdot)$ of several features. This is all provided in the dataset found in [14]. To measure the quality of a clustering, we use the average entropy over all clusters. We start with equation 2:

$$E = -\sum_i p_i \log p_i \quad (2)$$

where p_i is the proportion of instances in a cluster

that take the i^{th} value of the target attribute and E is defined as the entropy of that cluster. The quality of cluster hierarchy is then calculated as the sum of individual classes entropies weighted according to class size (Equation 3):

$$E = \sum_{i=1}^k \frac{m_i (-\sum_j p_{ij} \log(p_{ij}))}{m} \quad (3)$$

where p_{ij} is the proportion of the mixture component j in cluster i , m_i is the size of cluster i , k is the number of clusters, and m is the total number of examples. It is the most useful metric to score the success of the produced dendrogram at separating the groups[10]. We have compared different techniques to determine what procedure does the best job of clustering. In order to do that we have shown the

performance of our method by presentation of entropy values wrt N (number of clusters in merging procedure, starting from the root node and expanding the clusters in reversed order as they were merged).

First, we implemented our method using different base dendrograms to see if the linkage method of base dendrograms has any notable impact on the results or not. Figure 1 shows the result of our experiment for different base classifiers using *Mean* as the aggregation operator and average-linkage as the final linkage method. This is clear that the three curves don't exhibit different behavior. We did the same procedure for other aggregate operators and final linkage methods and concluded that our approach is independent of the linkage method used for base dendrograms. With the same procedure we concluded that our approach gives the same results independent of how the final linkage is done. Figure 2 illustrates three entropy curves resulted by using average-linkage, complete-linkage and single-linkage as the final linkage method on the descriptors obtained by applying *Mean* as the aggregation operator and average-linkage for the base dendrograms. The results for all other kinds of descriptors proved this independency feature of our approach.

General comparison of our multi-view clustering algorithms with the single-view counterparts for the data set is simply shown in Figure 3. The results for different single-view versions and three kinds of our multi-view algorithm (using average-linkage) for $N=1$ up to 20 confirms that although the *Min* aggregation operator doesn't provide good progress over single-view versions, the *Max* and *Mean* achieve dramatic improvements and much lower entropy values compared to the single-view clustering. It is easily seen that the improvements are going to be superior by increasing N . Figure 4 illustrates the improvements of our algorithm more clearly when N is set to 20. As it is clearly seen the multi-view method using Mean aggregator gives the best result.

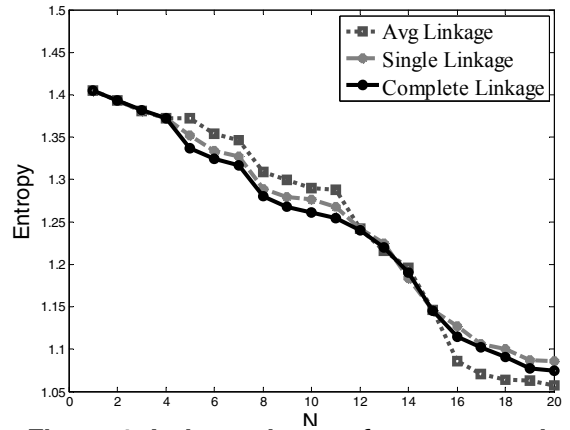


Figure 1. Independency of our approach on the base dendrograms linkage method

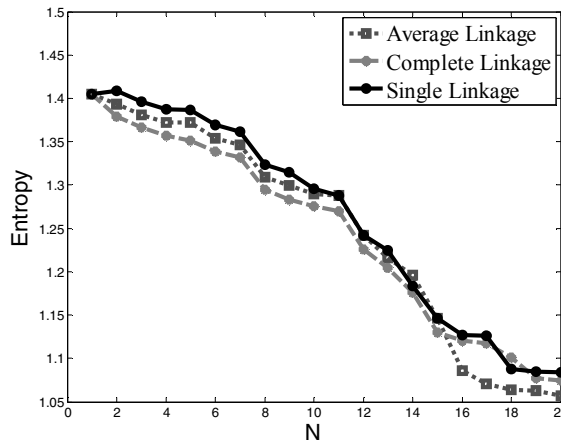


Figure 2. Independency of our approach on the final linkage method

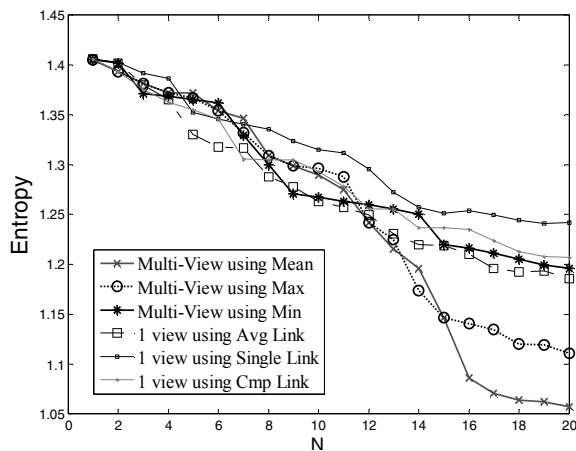


Figure 3. Comparison of multi-view algorithm with their single-view counterparts

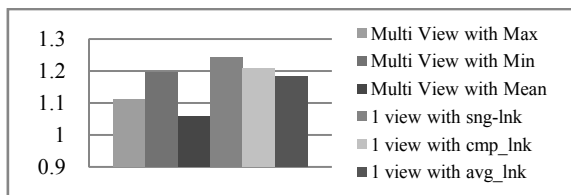


Figure 4. Comparison of single-view and multi-view entropies for N=20.

5. Conclusion

We presented the problem in which the data is split into some subsets and proposed an agglomerative multi-view approach to combine these views. The method is easy to implement, computationally cheap enough to run multiple times for a given dataset and for any number of views, and produces results which can be readily interpreted for our data sets. Testing different kinds of algorithms we gained significantly better results than the single-view variants in almost all cases. We discovered the best one which was using the mean of cophenetic matrices calculated separately

from corresponding views. We also checked single, average and complete linkage options as the distance of two clusters. The result can be used to represent a dendrogram for clustering the whole data set. Given the empirical results from the previous section, we conclude about the behavior of multi-view agglomerative clustering that independent of the linkage method, it outperforms the single view counterparts. As a future work, we aim to introduce some more optimal methods for combining descriptors in order to improve the performance of clustering.

6. References

- [1] J. Wang, H. Zeng, Z. Chen, H. Lu, L. Tao, and W. Ma. "Recom: Reinforcement clustering of multi-type interrelated data objects," In Proceedings of the ACM SIGIR Conference on Information Retrieval, 2003.
- [2] L. H. Ungar, D.P.Foster, "Clustering Methods for Collaborative Filtering", Workshop on recommendation System at the 15th National Conference on Artificial Intelligence, 1998.
- [3] A. Blum and T. Mitchell. "Combining labeled and unlabeled data with co-training", Proc. of the Conference on Computational Learning Theory, pages 92–100, 1998.
- [4] I. Muslea, C. Kloblock, and S. Minton. "Active + semisupervised learning = robust multi-view learning", Proceedings of the International Conference on Machine Learning, pages 435–442, 2002.
- [5] S. Dasgupta, M. Littman, and D. McAllester. "PAC generalization bounds for co-training", Proceedings of Neural Information Processing Systems (NIPS), 2001.
- [6] D. Yarowsky. "Unsupervised word sense disambiguation rivaling supervised methods", Proc. of the 33rd Annual Meeting of the Association for Comp. Linguistics, 1995.
- [7] J. Heer and E. H. Chi, "Identification of Web User Traffic Composition Using Multi-Modal Clustering and Information Scint", in 1st SIAM ICDM, Workshop on Web Mining, Chicago, 2001.
- [8] K. Nigam and R. Ghani. "Analyzing the effectiveness and applicability of co-training", Proc. of Information and Knowledge Management, 2000.
- [9] R. Ghani. "Combining labeled and unlabeled data for multiclass text categorization", Proc. of the International Conference on Machine Learning, 2002.
- [10] Steffen Bickel and Tobias Scheffer, "Multi-View Clustering," Proc. of the IEEE International Conference on Data Mining, 2004.
- [11] Dorthe B. Carr, Chris J. Young, Richard C. Aster, and Xiaoabing Zhang, "Cluster Analysis for CTBT Seismic Event Monitoring" (a study prepared for the U.S. Department of Energy).
- [12] H.C. Romesburg, "Cluster analysis for researchers." Belmont, CA: Lifetime Learning Publications. 1984.
- [13] Rohlf, F. J. and David L. Fisher. "Test for hierarchical structure in random data sets." Systematic Zool., 17:407-412. 1968.
- [14] Resources for Research on Web Spam, supported by the EU PASCAL Network of Excellence Challenge Program and DELIS EU- FET research project, <http://webspam.lip6.fr/wiki/pmwiki.php>.