Co-regularized Spectral Clustering with Multiple Kernels

Abhishek Kumar*
Dept. of Computer Science
University of Maryland
abhishek@cs.umd.edu

Piyush Rai*
School of Computing
University of Utah
piyush@cs.utah.edu

Hal Daumé III
Dept. of Computer Science
University of Maryland
hal@umiac.umd.edu

Abstract

We propose a co-regularization based multiview spectral clustering algorithm which enforces the clusterings across multiple views to agree with each-other. Since each view can be used to define a similarity graph over the data, our algorithm can also be considered as learning with multiple similarity graphs, or equivalently with multiple kernels. We propose an objective function that *implicitly* combines two (or more) kernels, and leads to an improved clustering performance. Experimental comparisons with a number of baselines on several datasets establish the efficacy of our proposed approach.

1 Introduction

Many real-world datasets have representations in form of multiple views [4, 5]. For example, webpages usually consist of both the page-text and hyperlink information; images on the web have associated captions with them; in multi-lingual information retrieval, the same document has multiple representations in different languages, and so on. Although these individual views might be sufficient on their own for a given learning task, they can often provide complementary information to each-other which can lead to improved performance on the learning task at hand.

Clustering seeks a partition of the data based on some similarity measure between the examples. In many cases, we have access to multiple similarity graphs (or kernels), constructed from multiple views of the data. Although one could use just one similarity graph in some graph based clustering algorithm [12], it makes more sense to combine the information from the multiple similarity graphs, and do clustering using the combined representation of similarities. Since the true underlying clustering would assign a point to the same cluster irrespective of the similarity graph being used, we can approach the multiview clustering problem by looking for clusterings that are consistent across the graphs defined over each of the views: corresponding nodes in each graph should have the same cluster membership. We use *views*, *graphs* and *kernels* interchangeably in the subsequent text. In this paper, we propose a spectral clustering algorithm that attempts to achieve this goal by *co-regularizing* the clustering hypotheses across views. We propose a spectral clustering objective function that *implicitly* combines multiple kernels to achieve a better clustering. Our approach is in contrast with several other existing works on multiple kernel learning [6] that try to learn an optimal kernel matrix, given a number of base kernel matrices. We focus on the two-kernel case for the simplicity of exposition, but the objective can be extended to more than two kernels in a straightforward manner.

2 Spectral Clustering

Spectral clustering [12] is a technique that exploits the properties of the Laplacian of the graph whose edges denote the similarities between the data points. The top k eigenvectors of the normalized graph Laplacian are relaxations of the indicator vectors that assign each node in the graph to one of the k clusters. Apart from being theoretically well-motivated, spectral clustering has the advantage of performing well on arbitrary shaped clusters, which is otherwise a shortcoming with several other clustering algorithms such as the k-means algorithm. Here we briefly outline the spectral clustering algorithm due to Ng et al. [8]:

*Authors contributed equally

- Construct an $n \times n$ positive semi-definite similarity matrix (or kernel) $\mathbf{K}$, where $\mathbf{K}_{ij}$ quantifies the similarity between samples i and j.
- Compute the normalized graph Laplacian $\mathcal{L} = \mathbf{D}^{-1/2}\mathbf{K}\mathbf{D}^{-1/2}$, where $\mathbf{D}$ is a diagonal matrix with $\mathbf{D}_{ii} = \sum_j \mathbf{K}_{ij}$.
- Let $\mathbf{U}$ denote a $n \times k$ matrix with columns as the top k eigenvectors of $\mathcal{L}$
- Normalize each row of $\mathbf{U}$ to obtain $\mathbf{V}$.
- Run the k-means algorithm to cluster the row vectors of $\mathbf{V}$.
- Assign example i to cluster m if the i-th row of $\mathbf{V}$ is assigned to cluster m by the k-means algorithm.

3 Co-regularized Spectral Clustering

Let $\mathbf{X} = \{\mathbf{x}_1^{(v)}, \mathbf{x}_2^{(v)}, \ldots, \mathbf{x}_n^{(v)}\}$ denote the examples in view v and $\mathbf{K}^{(v)}$ denote the similarity or kernel matrix of $\mathbf{X}$ in this view. We write the normalized graph Laplacian for this view as: $\mathcal{L}^{(v)} = \mathbf{D}^{(v)^{-1/2}}\mathbf{K}^{(v)}\mathbf{D}^{(v)^{-1/2}}$. The spectral clustering algorithm of Ng et al. [8] solves the following optimization problem for the normalized graph Laplacian $\mathcal{L}^{(v)}$:

$$\max_{\mathbf{U}^{(v)} \in \mathbb{R}^{n \times k}} tr(\mathbf{U}^{(v)^T}\mathcal{L}^{(v)}\mathbf{U}^{(v)}), \qquad \text{s.t.} \quad \mathbf{U}^{(v)^T}\mathbf{U}^{(v)} = I \tag{1}$$

where tr denotes the matrix trace. The matrix $\mathbf{U}^{(v)}$ can then be used in the algorithm outlined in Sec. 2 to get the final clustering. Our multi-kernel spectral clustering framework builds on the standard spectral clustering with a single kernel, by appealing to the co-regularization framework typically used in the semi-supervised learning literature [4].

Co-regularization essentially works by making the hypotheses learned from different views of the data agree with each other on unlabeled data [10]. The framework employs two main assumptions for its success: (a) the true target functions in each view should agree on labels for the unlabeled data (compatibility), and (b) the views are independent given the class label (conditional independence). The compatibility assumption is of particular importance since it allows us to shrink the space of possible target hypotheses by searching only over the compatible functions. Standard PAC-style analysis [4] shows that this also leads to reductions in the number of examples needed to learn the target function, since this number depends on the size of the hypothesis class.

For the clustering setting, we propose a co-regularization based approach to make the clustering hypotheses on different graphs (i.e., views) agree with each other. The effectiveness of spectral clustering hinges crucially on the construction of the graph Laplacian and the resulting eigenvectors that reflect the cluster structure in the data. Therefore, we construct an objective function that combines of the graph Laplacians from all the views of the data, and regularize the eigenvectors of each Laplacian such that the cluster structures resulting from each Laplacian look consistent across all the views.

Note from Section 2 that the matrix $\mathbf{U}^{(v)}$ is the data representation for the subsequent clustering step, (with i'th row mapping to the original i'th sample). In our proposed objective function, we encourage the row-wise similarities of $\mathbf{U}^{(v)}$ to agree with those of other views. Agreement in similarities of $\mathbf{U}^{(v)}$'s will more likely produce clusterings consistent with each other across views.

We will work with two-view case for the ease of exposition. We propose the following cost function as a measure of disagreement between clusterings of two views:

$$D(\mathbf{U}^{(v)}, \mathbf{U}^{(w)}) = \left\| \frac{\mathbf{K}_{\mathbf{U}^{(v)}}}{||\mathbf{K}_{\mathbf{U}^{(v)}}||_F^2} - \frac{\mathbf{K}_{\mathbf{U}^{(w)}}}{||\mathbf{K}_{\mathbf{U}^{(w)}}||_F^2} \right\|_F^2. \tag{2}$$

$\mathbf{K}_{\mathbf{U}^{(v)}}$ is the similarity matrix for $\mathbf{U}^{(v)}$, and $||\cdot||_F$ denotes the Frobenius norm of the matrix. The similarity matrices are normalized by their Frobenius norms to make them comparable across views. We choose linear kernel, i.e. $k(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^T\mathbf{x}_j$ as our similarity measure in Equation 2. This implies that we have $\mathbf{K}_{\mathbf{U}^{(v)}} = \mathbf{U}^{(v)}\mathbf{U}^{(v)^T}$. A linear kernel for $\mathbf{U}^{(\cdot)}$ is reasonable here because the Laplacian for spectral clustering has already taken care of the non-linearities present in the data (if any) and moreover, as we shall see, we get a nice optimization problem by using linear kernel for $\mathbf{U}^{(\cdot)}$. We also note that $||\mathbf{K}_{\mathbf{U}^{(v)}}||_F^2 = k$, where k is the number of clusters. Substituting this

in Equation 2 and ignoring the constant additive and scaling terms that depend on the number of clusters, we get

$$D(\mathbf{U}^{(v)}, \mathbf{U}^{(w)}) = -tr(\mathbf{U}^{(v)}\mathbf{U}^{(v)^T}\mathbf{U}^{(w)}\mathbf{U}^{(w)^T})$$

We want to minimize the above disagreement between the clusterings of views v and w. Combining this with the spectral clustering objectives of individual views, we get the following joint *maximization* problem for two graphs:

$$\max_{\mathbf{U}^{(v)}\in\mathbb{R}^{n\times k}, \mathbf{U}^{(w)}\in\mathbb{R}^{n\times k}} tr(\mathbf{U}^{(v)^T}\mathcal{L}^{(v)}\mathbf{U}^{(v)}) + tr(\mathbf{U}^{(w)^T}\mathcal{L}^{(w)}\mathbf{U}^{(w)}) + \lambda tr(\mathbf{U}^{(v)}\mathbf{U}^{(v)^T}\mathbf{U}^{(w)}\mathbf{U}^{(w)^T}) \quad \text{s.t.} \quad \mathbf{U}^{(v)^T}\mathbf{U}^{(v)} = I,\ \mathbf{U}^{(w)^T}\mathbf{U}^{(w)} = I \tag{3}$$

The hyperparameter λ trades-off the spectral clustering objectives and the spectral embedding (dis)agreement term. The joint optimization problem given by Equation 3 can be solved using alternating maximization w.r.t. $\mathbf{U}^{(v)}$ and $\mathbf{U}^{(w)}$. For a given $\mathbf{U}^{(w)}$, we get the following optimization problem in $\mathbf{U}^{(v)}$:

$$\max_{\mathbf{U}^{(v)}\in\mathbb{R}^{n\times k}} tr\{\mathbf{U}^{(v)^T}(\mathcal{L}^{(v)} + \lambda\mathbf{U}^{(w)}\mathbf{U}^{(w)^T})\mathbf{U}^{(v)}\}, \qquad \text{s.t.} \quad \mathbf{U}^{(v)^T}\mathbf{U}^{(v)} = I \tag{4}$$

This is a standard spectral clustering objective on view v with graph Laplacian $\mathcal{L}^{(v)} + \lambda\mathbf{U}^{(w)}\mathbf{U}^{(w)^T}$. The solution $\mathbf{U}^{(v)}$ is given by the top-k eigenvectors of this modified Laplacian. Since the alternating maximization can make the algorithm stuck in a local maximum [9], it is important to have a sensible initialization. We start with the graph Laplacian $\mathcal{L}^{(w)}$ of the more informative view and initialize $\mathbf{U}^{(w)}$. The alternating maximization is carried out after this until convergence. For fixed λ and n, the joint objective can be shown to be bounded from above by a constant. Since the objective is non-decreasing with the iterations, the algorithm is guaranteed to converge. In practice, we monitor the convergence by the difference in the value of the objective between consecutive iterations, and stop when the difference falls below a minimum threshold of $\epsilon = 10^{-4}$. In all our experiments, we converge within less than 10 iterations. Note that we can use either $\mathbf{U}^{(v)}$ or $\mathbf{U}^{(w)}$ in the final k-means step of the spectral clustering algorithm, depending on which of the views is more informative. If both views are believed to be equally informative, a column-wise concatenation of the two matrices could be used. The objective of Eq. 3 can be extended to more than two views by employing co-regularizers for each pair of the views. We leave the details for a longer version.

4 Experiments

We compare our co-regularization based multiple kernel spectral clustering approach with a number of baselines. In particular, we compare with:

- **Single View:** Using the most informative view, i.e., one that achieves the best spectral clustering performance using a single view of the data.
- **Feature Concatenation:** Concatenating the features of each view, and then running spectral clustering using the Laplacian derived from this new representation of the data.
- **Kernel Combination:** Combining different kernels by adding them, and then running standard spectral clustering on the corresponding Laplacian. As suggested in earlier findings [6], even this seemingly simple approach often leads to near optimal results as compared to more sophisticated approaches.
- **CCA based Feature Extraction:** Applying CCA for feature fusion from multiple views of the data [3], and then running spectral clustering using these extracted features.
- **Minimizing-Disagreement Spectral Clustering:** Our last baseline is the *minimizing-disagreement* approach to spectral clustering [7], and is perhaps most closely related to our co-regularization based approach to spectral clustering.

We report experimental results on one synthetic and two real-world datasets. Our synthetic data consists of two views and is generated in a manner akin to [13] which first chooses the cluster c_i each sample belongs to, and then generates each of the views $x_i^{(1)}$ and $x_i^{(2)}$ from a two-component Gaussian mixture model. These views are combined to form the sample $(x_i^{(1)}, x_i^{(2)}, c_i)$. Our first real-world dataset is taken from the handwritten digits (0-9) data from the UCI repository. The dataset consists of 2000 examples, with view-1 being the 76 Fourier coefficients, and view-2 being the 216 profile correlations of each example image. Our second real-world dataset is a subset of the Caltech-101 data from the Multiple Kernel Learning repository [1] from which we chose 450

examples having 30 underlying clusters. For this data, we chose the bio-inspired "Sparse Localized Features" as the first view and the 4x4 "Pyramid Histogram Of visual Words" as the second view. We compare all the approaches on a number of evaluation measures. Here we report: (1) F-score which is the harmonic mean of precision and recall scores, and (2) Cluster Entropy. We also experimented with the Normalized-Mutual-Information and Rand-Index but do not include those results due to space limitation. All the results are reported with the best choice of the kernel and the hyperparameters. The results are shown in Table 1. As we can see, on all the datasets experimented with, the co-regularization approach outperforms all the other baselines.

	Synthetic Dataset		**Handwritten Digits Data**		**Caltech-101 Data**	
	F1	Avg. Entropy	F1	Avg. Entropy	F1	Avg. Entropy
SV	0.69(±0.00)	0.73(±0.00)	0.58(±0.02)	1.20(±0.03)	0.21(±0.01)	2.42(±0.04)
FC	0.68(±0.00)	0.69(±0.00)	0.54(±0.03)	1.28(±0.05)	-	-
KC	0.70(±0.00)	0.65(±0.00)	0.71(±0.05)	0.86(±0.11)	0.09(±0.01)	3.06(±0.04)
CCA	0.71(±0.00)	0.65(±0.00)	0.63(±0.03)	1.08(±0.07)	0.17(±0.01)	2.63(±0.04)
MD	0.69(±0.00)	0.67(±0.00)	0.69(±0.04)	0.87(±0.09)	0.10(±0.01)	3.00(±0.04)
CS	**0.75(±0.00)**	**0.62(±0.00)**	**0.72(±0.05)**	**0.84(±0.12)**	**0.23(±0.01)**	**2.34(±0.04)**

Table 1: The various approaches: **SV:** Single View, **FC:** Feature Concatenation, **KC:** Kernel Combination, **MD:** Minimizing Disagreement spectral clustering, **CS:** co-regularized spectral clustering. The base k-means clustering was run with 10 different initializations; mean and standard deviations are reported. Note 1: Since the Caltech data views are given in form of the kernel matrices, we did not try **FC** on this. Note 2: The std. dev. of all algorithms is zero on the synthetic data; different initializations lead to the same clustering in this case.

5 Related Work and Conclusion

A number of clustering algorithms have been proposed in the past to learn with multiple views of the data. Some of them first extract a set of shared features from the multiple views and then apply any off-the-shelf clustering algorithm such as k-means on these features. The Canonical Correlation Analysis [5, 3] (CCA) based approach is an example of this. Alternatively, some other approaches exploit the multiple views of the data as part of the clustering algorithm itself. For example, [2] proposed an EM based framework for multi-view clustering in mixture models. Multi-view clustering algorithms have also been proposed in the framework of spectral clustering [15, 7]. In [11], the information from multiple graphs are fused using Linked Matrix Factorization. [14] uses maximum margin clustering (MMC) with multiple kernels, and simultaneously finds the best cluster labeling and the optimal linear combination of base kernels. In contrast, our approach, although uses multiple kernels, does not require explicitly combining the kernels. Furthermore, each step leads to a simple eigenvalue problem which is efficiently solvable using state-of-the-art eigensolvers.

References

[1] The UCSD Multiple Kernel Learning Repository. http://mkl.ucsd.edu.

[2] S. Bickel and T. Scheffer. Multi-View Clustering. In *ICDM*, 2004.

[3] M. B. Blaschko and C. H. Lampert. Correlational Spectral Clustering. In *CVPR*, 2008.

[4] A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. In *COLT*, 1998.

[5] K. Chaudhuri, S. M. Kakade, K. Livescu, and K. Sridharan. Multi-view Clustering via Canonical Correlation Analysis. In *ICML*, 2009.

[6] C. Cortes, M. Mohri, and A. Rostamizadeh. Learning non-linear combination of kernels. In *NIPS*, 2009.

[7] V. R. de Sa. Spectral Clustering with two views. In *Proceedings of the Workshop on Learning with Multiple Views, ICML*, 2005.

[8] A. Ng, M. Jordan, and Y. Weiss. On spectral clustering: analysis and an algorithm. In *NIPS*, 2002.

[9] D. Niu, J. G. Dy, and M. I. Jordan. Multiple non-redundant spectral clustering views. In *ICML*, 2010.

[10] V. Sindhwani, P. Niyogi, and M. Belkin. A Co-regularization approach to semi-supervised learning with multiple views. In *Proceedings of the Workshop on Learning with Multiple Views, ICML*, 2005.

[11] W. Tang, Z. Lu, and I. S. Dhillon. Clustering with Multiple Graphs. In *ICDM*, 2009.

[12] U. von Luxburg. A Tutorial on Spectral Clustering. *Statistics and Computing*, 2007.

[13] X. Yi, Y. Xu, and C. Zhang. Multi-view em algorithm for finite mixture models. In *ICAPR, Lecture Notes in Computer Science, Springer-Verlag*, 2005.

[14] B. Zhao, J. T. Kwok, and C. Zhang. Multiple Kernel Clustering. In *SDM*, 2009.

[15] D. Zhou and C. J. C. Burges. Spectral Clustering and Transductive Learning with Multiple Views. In *ICML*, 2007.