Single Data, Multiple Clusterings

Sajib Dasgupta and Vincent Ng
Human Language Technology Research Institute
University of Texas at Dallas
Richardson, TX 75083-0688
{sajib,vince}@hlt.utdallas.edu

Abstract

There has been extensive research in the clustering community on formalizing the definition of the quality of a given data clustering. However, is it possible to measure the quality of a clustering unless human judgment is taken into consideration? The notion of quality is subjective: for example, given the task of clustering a set of movie reviews, some users might want to cluster them according to sentiment, while others might want to cluster them according to genre. If the clustering algorithm is passive (i.e., it does not have the ability to produce multiple clusterings by actively taking user intent into account), it is hard to justify the algorithm to be qualitatively best across different domains. There has been a recent surge of interest in quantifying how *clusterable* a dataset is [2]. Can we similarly define *multi-clusterability*? In this paper, we present a (really) simple *active clustering* architecture that can help understand the multi-clusterability of a dataset.

1 Introduction

Although it is common to produce only a single clustering of a dataset, in many cases data points can be clustered along different dimensions. For example, while traditional work on text clustering has largely focused on grouping documents by topic, it is conceivable that a user may want to cluster documents along other dimensions, such as the author's mood, gender, or age. Similarly, a set of movie reviews can be clustered according to genre (e.g., action, romantic, or documentary) or sentiment (positive, negative, or neutral). A traditional clustering algorithm produces a clustering along only one dimension, which is typically the dimension for which the objective function employed by the clustering algorithm achieves optimality. A clustering algorithm fails if a user wants to cluster along a dimension that is not the optimal dimension produced by the algorithm. The question, then, is: can a clustering algorithm produce a clustering according to the user-specified dimension that is suboptimal with respect to the objective function?

There has been extensive research in the clustering community to measure the quality of a given data clustering. Typically, these qualitative scores depend on intra-cluster similarity, inter-cluster dissimilarity, and the size of the clusters. Recently, Ackerman and Ben-David [1] have proposed a set of axiomatic properties that they believe are prerequisites of any good clustering-quality measures. However, one important notion that is commonly left out of a qualitative measure is the human factor. As mentioned before, different subsets of features might lead to different kind of clusterings of a dataset. Even though existing clustering qualitative scores help to identify a particular clustering as the best one, it might not be deemed fit by an end user.

One may argue that it is possible to design the feature space in such a way that helps induce the user-desired clustering. Given enough *side* information, one may even learn the similarity metric that meets user demand [8]. However, these approaches are typically knowledge-dependent or domain-specific. We desire a *knowledge-lean* approach that can generate the user-desired clustering without reliance on prior knowledge of features or similarity function.

One possible solution is to incorporate user feedback into the clustering process to ensure that the points are clustered according to the dimension that the user wants. Recent work on *active clustering* have involved the development of algorithms where the user is asked to *construct* the feature space in an interactive manner [4] or *identify* the clusters that need to be merged or split [3]. However, these systems require considerable human feedback, thus making the system semi-supervised.

In this paper, we propose a (really) simple active spectral clustering architecture that is capable of producing the user-desired clustering. In comparison to previous work on feedback-oriented clustering, the amount of user feedback required by our algorithm is minimal. In fact, the feedback turns out to be as simple as a cursory look at a list of features and is required only once. Experimental results are very promising: our system is able to generate the user-specified clustering with reasonable accuracies on several challenging text classification tasks, thus providing suggestive evidence that our approach is viable. One distinguishing feature of our approach is that it can give users a rough idea of the visible clustering dimensions of the data, in case the user does not know how she wants the data points to be clustered (due to the lack of knowledge of the data, for instance).

2 Active Spectral Clustering

We start with some notations. Let $X = x_1, ..., x_n$ be a set of n data points to be clustered[1], $s : X \times X \rightarrow \Re$ be a similarity function over X, and S be the similarity matrix that captures pairwise similarities (i.e., $S_{i,j} = s(X_i, X_j)$). We want to learn a partitioning function f that takes S as input and outputs a 2-way partition $C = \{C_1, C_2\}$ such that $C_1 \cup C_2 = X$ and $C_1 \cap C_2 = \phi$. Given a clustering algorithm with a predefined objective function, the *optimal partitioning function* produces a clustering that optimizes the objective. All other partitioning functions are *suboptimal*. Below we show how we learn optimal and suboptimal partitioning functions using spectral clustering.

Second eigenvector as optimal partitioning function: Normalized cut is one of the most widely used objective function in spectral clustering. The real-valued partitioning function f that captures the optimal 2-way normalized cut partition of X is the solution to the following constrained optimization problem: $\text{argmin}_{f \in \Re^n} \quad \sum_{i,j} S_{i,j} (\frac{f(i)}{\sqrt{d_i}} - \frac{f(j)}{\sqrt{d_j}})^2$ subject to $f^T D f = 1$ and $Df \perp \mathbf{1}$, where D is a diagonal matrix with $D_{i,i} = \sum_j S_{i,j}$, and $d_i = D_{i,i}$. The closed form solution to this optimization problem is the eigenvector corresponding to the second largest eigenvalue of the Laplacian matrix, $D^{-1/2} S D^{1/2}$ [7]. Clustering using the second eigenvector is trivial: we can just apply 2-means to the n data points represented by the second eigenvector.

Deriving suboptimal partitioning functions: As mentioned above, suboptimal partitioning functions are useful if they can produce a clustering according to user interest. Each eigenvector (except for the first one) with a non-zero eigenvalue captures a distinct dimension of the data. Hence, we can take the top $(m + 2)$ eigenvectors corresponding to the $(m + 2)$ largest eigenvalues, and apply 2-means to the third through the $(m + 2)$th eigenvectors separately to produce m suboptimal clusterings.

Given that we have learned one optimal and m suboptimal partitioning functions, the next question is how we determine which one captures the user interest. One way to do this is to have the user inspect the partitions and decide which corresponds most closely to the desired clustering. The main drawback associated with this kind of user feedback is that the user may have to inspect a large number of data points in order to make a decision. Hence, to reduce human effort, we employ an alternative procedure: we (1) identify the most informative features characterizing each partition, and (2) have the user inspect just the features rather than the data points.

To select these informative features, we rank them by their weighted log-likelihood ratio (WLLR): $P(w_i \mid c_j) \cdot \log \frac{P(w_i \mid c_j)}{P(w_i \mid \neg c_j)}$, where w_i and c_j denote the ith feature and the jth cluster respectively. Informally, feature w will have a high rank with respect to cluster c if it appears frequently in c and infrequently in $\neg c$. This correlates reasonably well with what we think an informative feature should be. Now, for each of the $(m+1)$ partitions, we (1) derive top 100 features for each cluster according to the WLLR, and (2) present the ranked lists to the user. The user will select one of the partitions as most relevant to her interest by inspecting as many features in the ranked lists as needed.

[1] We present our system for 2-way clustering task, but it can be easily extended for n-way ($n > 2$) clustering.

3 Evaluation

Experimental setup. We evaluate our system on several text datasets.

Sentiment: Three datasets contain customer reviews of three different types of products from Amazon [books (BOO), DVDs (DVD), and electronics (ELE)] [5]. The goal is to cluster them according to *sentiment* (i.e., positive or negative).

2 Newsgroups: To illustrate the difference between topic-based clustering and sentiment-based clustering, we will also show results on POL, a dataset created by taking all the documents from two sections of 20 Newsgroups, namely, `sci.crypt` and `talks.politics`. The goal is to cluster them according to *topic* (i.e., politics or science).

Artificial datasets: Finally, we create two artificial datasets that possess multiple clustering dimensions, namely BOO-DVD and ELE-KIT. For example, the BOO-DVD dataset consists of all of the reviews taken from the BOO and DVD domains, and ELE-KIT contains all the reviews from the ELE and kitchen (KIT) domains. The goal is to see whether they can be clustered according to either *topic* (e.g., book vs. DVD) or *sentiment*.

To preprocess a document, we follow Dasgupta and Ng [6]: we first tokenize and downcase it, and then represent it as a vector of unigrams, using frequency as presence. We use dot product as the similarity function in spectral clustering. Finally, we learn one optimal and three suboptimal partitioning functions (i.e, $m = 3$), assuming that they are enough to capture the desired clusterings.

Baseline. We use Non-negative Matrix Factorization (NMF) as our baseline, which has recently shown to be effective for document clustering. Note that NMF produces only one clustering for a given dataset. Hence, while evaluating NMF on the two artificial datasets, we compare its output against the correct topic-based clustering and the correct sentiment-based clustering separately and report the better result in terms of both accuracy and Adjusted Random Index (ARI) in Table 2.

Human experiments. Note that our approach requires a user to specify which of the four partitioning functions (defined by the second through fifth eigenvectors) are relevant by inspecting a set of features derived from each partition (see Table 1 for a snippet). To better understand how easy it is for a human to select the desired dimension, we performed the experiment independently with five humans and computed the agreement rate. Interestingly, the human judges achieved perfect agreement rate on all but the DVD-ELE artificial dataset, where near-perfect agreement rate (4/5) was achieved. These results, together with the fact that it took less than five minutes to identify the relevant dimension for each dataset, indicate that asking a human to determine the desired clustering based on solely the informative features is a viable task.

Clustering results. Next, we evaluate the clustering of each dataset using the partitioning functions selected by the majority of the human judges (see Table 2). As we can see, our active clustering algorithm outperforms the baseline by a large margin, which is not unexpected, as the baseline does not (and cannot) take into account human feedback. Note that the sentiment-based clustering accuracies are much lower than those of topic-based clustering. This can potentially be attributed to the fact most reviews are sentimentally ambiguous, as reviewers typically discuss both the positive and negative aspects of a product before making a final decision. Interestingly, for both artificial datasets, at least one eigenvector corresponds to either topic or sentiment, which seems to suggest that spectral learning is effective enough to unearth both the topic and sentiment dimensions when both of them are present in a dataset.

4 From Practice to Theory

As practitioners, we can significantly benefit from answers to the following theoretical questions:

Can we quantify multi-clusterability? There has been a surge of recent interest in quantifying how *clusterable* a dataset is, which Ackerman and Ben-David [2] rightly define to be how "strong" or "conclusive" the clustering structure of a given dataset is. Can we similarly define the *multi-clusterability* of a dataset? Not all datasets are multi-clusterable: for example, the 2 Newsgroups dataset is not multi-clusterable, whereas the remaining datasets are. We desire some axiomatic properties that can define and quantify multi-clusterability with theoretical justifications.

Can we quantify clusterability along a particular dimension? A dataset is not clusterable along all dimensions that are linguistically plausible. For example, we could not find any partitioning function that clusters along gender for any of our evaluation datasets. It is possible that all the documents in

DVD				BOO-DVD			
$\mathbf{e}_2$	$\mathbf{e}_3$	$\mathbf{e}_4$	$\mathbf{e}_5$	$\mathbf{e}_2$	$\mathbf{e}_3$	$\mathbf{e}_4$	$\mathbf{e}_5$
$\mathbf{C}_1$	$\mathbf{C}_1$	$\mathbf{C}_1$	$\mathbf{C}_1$	$\mathbf{C}_1$	$\mathbf{C}_1$	$\mathbf{C}_1$	$\mathbf{C}_1$
fan	music	music	saw	reader	young	wonderful	loved
bought	wonderful	video	watched	information	men	excellent	children
video	collection	found	fan	research	scene	music	novel
series	cast	workout	loved	subject	cast	highly	enjoyed
money	quality	bought	series	important	role	collection	wonderful
workout	excellent	videos	comedy	text	films	features	bought
$\mathbf{C}_2$	$\mathbf{C}_2$	$\mathbf{C}_2$	$\mathbf{C}_2$	$\mathbf{C}_2$	$\mathbf{C}_2$	$\mathbf{C}_2$	$\mathbf{C}_2$
role	money	series	money	music	bought	boring	waste
between	waste	cast	quality	actors	workout	waste	original
young	thought	fan	video	script	recipes	novel	version
cast	worst	stars	director	films	information	worst	quality
world	nothing	original	found	comedy	disaapointed	pages	review
actors	saw	comedy	version	scene	waste	ending	sound

Table 1: Top six features induced for each dimension for the DVD and BOO-DVD domains. The lightly and darkly shaded columns correspond to the topic and sentiment dimensions respectively as selected by the human judges. $\mathbf{e}_2, \ldots, \mathbf{e}_5$ are the top eigenvectors; $\mathbf{C}_1$ and $\mathbf{C}_2$ are the clusters.

	NMF		Topic		Sentiment	
	Acc	ARI	Acc	ARI	Acc	ARI
POL	85.2	0.53	93.7 (2)	0.76 (2)	–	–
DVD	50.3	0.01	–	–	70.3 (3)	0.17 (3)
BOO	52.1	0.01	–	–	69.5 (4)	0.15 (4)
ELE	63.7	0.07	–	–	66.3 (3)	0.10 (3)
BOO-DVD	70.2	0.18	77.1 (2)	0.29 (2)	68.8 (3)	0.14 (3)
DVD-ELE	82.5	0.51	95.9 (2)	0.78 (2)	62.6 (3)	0.06 (3)

Table 2: Topic- and sentiment-based clustering results. The human-selected eigenvector is in parentheses.

a given dataset were written entirely by men or by women, or the gender distribution is too skewed to learn any gender-wise partitioning function. It would be interesting to quantify how clusterable a dataset is along a given dimension.

Can we quantify ambiguity? As we can see from Table 2, for all three sentiment datasets, the second eigenvector fails to capture the sentiment-wise partitioning. The reason is that many reviews are *sentimentally ambiguous*, as a reviewer may have negative opinions on the actors but at the same time talk enthusiastically about how much she enjoyed the plot, for instance. The presence of both positive and negative sentiment-bearing words in these reviews renders the sentiment dimension *hidden* as far as clustering is concerned. There seems to be a nice correlation between the degree of ambiguity and and the quality of the clustering produced by spectral techniques (at least observed empirically). Is it possible to prove it, or at least quantify the extent to which a dataset is ambiguous?

References

[1] M. Ackerman and S. Ben-David. Measures of clustering quality: A working set of axioms for clustering. In *Advances in NIPS*, pages 121–128, 2008.

[2] M. Ackerman and S. Ben-David. Clusterability: A theoretical study. In *Proceedings of AISTATS*, 2009.

[3] M.-F. Balcan and A. Blum. Clustering with interactive feedback. In *Proceedings of ALT*, 2008.

[4] R. Bekkerman, H. Raghavan, J. Allan, and K. Eguchi. Interactive clustering of text collections according to a user-specified criterion. In *Proceedings of IJCAI*, pages 684–689, 2007.

[5] J. Blitzer, M. Dredze, and F. Pereira. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *Proceedings of the ACL*, pages 440–447, 2007.

[6] S. Dasgupta and V. Ng. Topic-wise, sentiment-wise, or otherwise: Identifying the hidden dimension for unsupervised text classification. In *Proceedings of EMNLP*, 2009.

[7] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

[8] E. P. Xing, A. Y. Ng, M. I. Jordan, and S. J. Russell. Distance metric learning with application to clustering with side-information. In *Advances in NIPS*, pages 505–512, 2002.