SEMANTIC VIDEO CLUSTERING ACROSS SOURCES USING BIPARTITE SPECTRAL CLUSTERING

Dong-Qing Zhang, Ching-Yung Lin**, Shi-Fu Chang* and John R. Smith***
** Dept. of Electrical Engineering, Columbia Univ., {dqzhang, sfchang}@ee.columbia.edu*
*** IBM T.J. Watson Research Center, {cylin, jrsmith}@watson.ibm.com*

ABSTRACT

Data clustering is an important technique for visual data management. Most previous work focuses on clustering video data within single sources. In this paper, we address the problem of clustering across sources, and propose novel spectral clustering algorithms for multi-source clustering problems. Spectral clustering is a new discriminative method realizing clustering by partitioning data graphs. We represent multi-source data as bipartite or K-partite graphs, and investigate the spectral clustering algorithm under these representations. The algorithms are evaluated using TRECVID-2003 corpus with semantic features extracted from speech transcripts and visual concept recognition results from videos. The experiments show that the proposed bipartite clustering algorithm significantly outperforms the regular spectral clustering algorithm to capture cross-source associations.

1. INTRODUCTION

Mining multimedia data has become an important research topic due to the increasing demand for managing vast amount of multimedia content. Data clustering is an important technique toward automatic multimedia content management and multimedia data mining.

Clustering techniques are intended to group data with similar attributes into clusters that exhibit certain high-level semantics. Much previous effort has been focusing on data clustering problem in single video sources, for example [1][2]. However, in many practical applications, it is often more interesting to discover the events co-occurring in multiple sources, for example same-topic news stories reported by multiple broadcast channels. These applications raise new research problems of finding cross-source clusters or associations.

One of the closely related research projects is the Topic Detection and Tracking (TDT) [3], an effort aiming to detect and track novel or retrospective events in information streams. However, in TDT, although the data sources are assumed to come from multiple sources, the problem of finding cross-source topics or events has not been explicitly addressed in the literatures.

Spectral clustering [4] is a clustering algorithm recently developed in the machine learning community. Comparing with many prior clustering techniques, spectral clustering has the following advantages: it is a pairwise distance based approach, allowing the method working on non-metric space; it is a discriminative approach, which does not assume the data in each cluster having convex distribution; it is free of singularity problem caused by high dimensionality of feature vectors. These properties make the spectral clustering algorithm a favorable choice for visual data clustering, since visual features are often high dimensional and the distribution of each cluster is not necessarily convex or a Gaussian function.

Data mining on bipartite graphs has been studied in some previous work. For example, in document clustering, [5] proposes to use an algorithm called co-clustering to find the document clusters and word clusters by constructing a bipartite graph where document nodes and word nodes are separated in the two parties of bipartite graph. In [6], a bipartite normalized cuts algorithm is proposed to realize normalized cuts partitioning in bipartite graphs.

In this paper, we deal with the cross-source clustering problem by using a bipartite or K-partite graph representation. The regular spectral clustering algorithm in [4] is extended to the K-partite graph and bipartite graph, where an efficient algorithm can be found because of the particular structure of bipartite graph. The developed techniques are applied to semantic video clustering for evaluation. Specific evaluation metrics are proposed for evaluating the performance of cross-source clustering. The experiments show that the proposed bipartite spectral clustering algorithm outperforms the regular spectral clustering to discover cross-source associations. Moreover, spectral clustering is shown to outperform K-means clustering algorithm in clustering videos using visual features.

2. SPECTRAL CLUSTERING ALGORITHM

Spectral clustering (SC) is a discriminative clustering algorithm differing from many previous clustering algorithms such as K-means or EM. In SC, the data points

(for example, documents or images) are represented as vertices in a graph, whose edge weights signify the pairwise similarities of data points. Clustering is realized by partitioning the graph into disjoint sub-graphs. Prior to SC, several graph partitioning methods have been studied for clustering, for example, *min-cut* and *normalized cuts* algorithm. However, spectral clustering is more robust than *min-cut* algorithm and *normalized cuts* partitioning by using multiple eigenvectors and embedded K-means algorithm to achieve superior accuracy and robustness [7].

To be complete, the spectral clustering algorithm is briefly described as follows. Suppose there are *n* data points, which need to be grouped into k clusters. The spectral clustering consists of the following steps:

1. Form the affinity matrix $A \in R^{n \times n}$, where

$$A_{ij} = \begin{cases} S(i,j), & i \neq j \\ 0, & i = j \end{cases}$$

where $S(i,j)$ is the similarity between the point *i* and point *j*. The similarity can be the inner product of two feature vectors or as in [7] set to $S(i,j) = \exp\left(-d_{ij}^2 / 2s^2\right)$ Where d_{ij} is the distance of point *i* and *j*, and s is the control parameter to adjust the sensitivity of clustering to the distance measure.

2. Construct the degree matrix *D* as a diagonal matrix, where

$D_{ii} = \sum_{i=1}^{n} A_{ij}$ and the Laplacian matrix $L = D^{-1/2} A D^{-1/2}$

3. Find $x_1, x_2, ..., x_k$ the k largest eigenvectors of *L*, and form the matrix $X = [x_1, x_2, ..., x_k]$ by stacking the vectors in columns.

4. Normalize each row of *X* so that each row has unit length.

5. Treating each row of *X* as a feature vector, cluster them into *k* clusters via K-means algorithm

3. INTERPRETATION OF SECTRAL CLUSTERING

Partitioning a graph into two disjoint subgraphs can be thought of as a process to assign the nodes with label 0 and 1, denoting two non-overlapping clusters. Each eigenvector from the step 3 in the above SC algorithm can be considered a function that assigns each vertex a real number representing the confidence to assign label 1 to a vertex (confidence ranging from –1 to 1). Whereas, calculating eigenvectors yields such functions orthogonal to one other. These eigenvectors are illustrated as one-dimensional functions in the Figure 1. According to the Figure 1, if we take 0 as a threshold, thresholding the second eigenvector would lead to the bipartition of all data; Likewise, thresholding the third eigenvector would further partition the clusters yielded by the second eigenvector and so on.

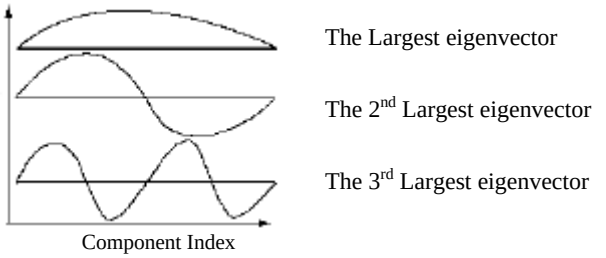

Figure 1. The one-dimensional illustration of spectral clustering. The *n*th largest eigenvector abbreviates for the eigenvector corresponding to the *n*th largest eigenvalue.

Therefore, the normalized row vectors of *X* can be thought of as the normalized meta-feature vectors by stacking all confidence values from each round of partitioning. K-means algorithm is therefore intended to re-cluster these meta-feature vectors.

Furthermore, it is well-known that the second largest eigenvector of the matrix *L* is an approximate solution for bipartitioning the graph with the *normalized cuts* principle, whose exact solution is NP hard. Namely, the 2nd largest eigenvector of the matrix *L* is near optimal to bipartition the graph in a discriminative fashion (*normalized cuts*). The rationale is likewise for the rest of the eigenvectors.

Hence, roughly speaking, the SC algorithm is a procedure that combines the discriminative partitioning algorithm (*normalized cuts*) with the generative clustering algorithm (K-means).

4. SPECTRAL CLUSTERING ACROSS SOURCES

Clustering across sources has different objectives comparing with the regular clustering. For the multi-source data, we may be more interested in the data associations across sources than those within individual sources. This requires additional strategies to handle situations that are not covered by standard approaches. For example, in multi-source data, the sizes of within-source clusters may be much larger than those of cross-source clusters, resulting in the dominance of within-source clusters and loss of cross-source clusters. Furthermore, the distances of data points within one source may be much larger than those of the data points across sources, resulting in the difficulty to form cross-source clusters.

To solve these problems, we represent the multi-source data as a K-partite graph [10], a graph whose vertices can be partitioned into K disjoint sets so that no two vertices within the sets are linked by arcs. In this representation, the video data from a particular source are represented by the vertices in one of the disjoint sets. In particular, if there are only two sources, the constructed graph is the familiar bipartite graph, as shown in the Figure 2.

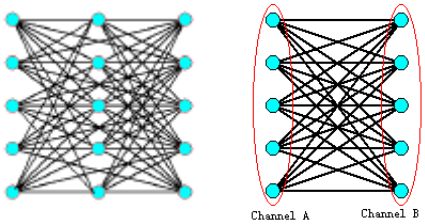

Figure 2. 3-partite graph and bipartite graph

Partitioning K-partite graph is straightforward by using the regular spectral clustering algorithm. However, for bipartite graph, we can take the advantage of its particular structure to increase efficiency and reduce the space requirement. This is strongly desirable in computation since spectral clustering algorithm takes $O(n^2)$ space complexity and $O(n^3)$ time complexity.

In the bipartite case, the matrix A and D can be written as the following block matrix

$$A = \begin{bmatrix} 0 & B \\ B^T & 0 \end{bmatrix}, \text{ and } D = \begin{bmatrix} D_u & 0 \\ 0 & D_d \end{bmatrix}$$

where $B \in R^{m \times l}, D_u \in R^{m \times m}, D_d \in R^{l \times l}, m + l = n$. Then the eigenvalue problem of $D^{-1/2} A D^{-1/2}$ can be converted to the following singular value problem

$$B_s x_d = l x_u \text{ where } B_s = D_u^{-1/2} B D_d^{-1/2}$$

the eigenvector matrix is then formed by

$$X = \begin{bmatrix} x_{u1}, x_{u2}, \ldots x_{uk} \\ x_{d1}, x_{d2}, \ldots x_{dk} \end{bmatrix}$$

The rest of procedure for bipartite clustering is the same as the spectral clustering in section 2. The singular value conversion equations are derived in the appendix for reference. This simplification reduces the computation of a large $n \times n$ matrix to a smaller $m \times l$ matrix, leading to less expensive computation.

5. SEMATNIC FEATURES FOR CLUSTERING

The methodology of semantic feature detection and recognition is out of the scope of this paper. Here, we only utilize the existing feature detection results from the corpus and our collaborators. Two sets of semantic features are employed in clustering: textual features from speech transcripts and visual semantic features from the visual concept detection results [9].

For story-level clustering, term vectors are extracted from speech transcripts to represent the semantics of video stories. In order to take into account the relative frequencies of different terms, term vectors are calculated using TF*IDF measure.

For shot-level visual clustering, we use a visual semantic representation system called *model vector* [9]. The *model vector* system associates each shot with a vector, each of whose components is the confidence score of finding the corresponding visual concept in the given shot. The concept detection results are yielded by multiple discriminative classifiers. Ensemble fusion algorithms are followed to combine the output from multiple classifiers to boost the performance. The entire procedure may result in unusual non-convex or even skewed distributions of feature vectors.

We adopt inner product of two feature vectors as the similarity measure for both speech term vectors and visual *model vectors*.

6. EVLAUTION METRICS AND EXPERIMENTS

We use TREC-VID 2003 video data for performance evaluation. TREC-VID 2003 is an open benchmark for the performance evaluation of concept detection and search in videos. The corpus includes news videos from two broadcast channels: ABC and CNN. Furthermore, since these data are a subset of TDT2 [3] corpus, the ground truth of topic detection can be found in the Linguistic Data Consortium web site. The entire TREC-VID 2003 data set is partitioned into two disjoint subsets: development set and test set. We only use the development set, since there are well-labeled story boundary data in the development set.

The speech transcripts are available in the TREC-VID 2003 corpus. For story level clustering, the story boundaries are extracted from the segmentation ground truth data in the development set. For visual semantic clustering. The recognition results are from the IBM TREC-VID 2003 team. Our test set consists of 2128 stories for story clustering, and 2434 shots (sampled subset) for shot level clustering.

Two evaluation metrics are developed. As to story level clustering, we use a subjective metric, where the ground truth data are labeled by human annotators. The metric aims to gauge the capability of capturing the cross-source associations. Two metrics are defined for subjective evaluation: *Recall* is the capability of capturing the cross-source associations, defined as

$$\frac{\#(GT \text{ associations captured by clustering})}{\#(\text{overall } GT \text{ associations})}$$

where GT associations means ground truth pairwise cross-source links. # is the shorthand for "number of". *Precision* is the accuracy of finding pairwise associations, written as

$$\frac{\#(\text{correct associations by clustering})}{\#(\text{all associations by clustering})}$$

In order to calculate the precision, each detected cluster needs to be aligned to a ground truth cluster. The principle of alignment is that the assigned ground truth cluster should have maximum overlap with the evaluated cluster.

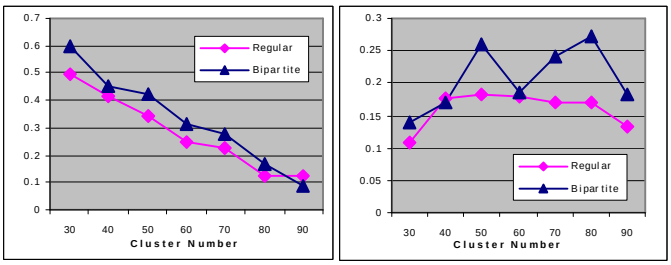

Figure 3. The recall (left) and precision (right) of regular spectral clustering and bipartite spectral clustering

The ground truth data come from the topic annotation data in the TDT2 corpus. We convert the topic annotation data in TDT2 to 28 ground truth clusters covering 499 stories. The result comparison is shown in the Figure 3.

For visual concept based clustering, it is difficult to evaluate using subjective metric, since most clusters do not have semantic meaning. Hence, we use the averaged homogeneity of clusters as a metric, defined as

$$hom = \frac{1}{|E|}\sum_{e_i \in E} w(e_i)$$

where E is the set consisting of all edges whose two end-vertices residing in the same cluster. $w(e_i)$ is the weight of the edge e_i. Intuitively, *hom* is the average similarity of two data points in clusters.

We compare the performance of K-means clustering and spectral clustering. Results are shown in the figure 4.

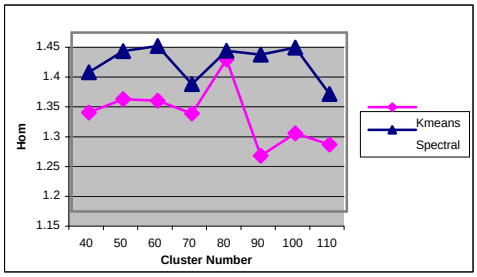

Figure 4. Comparison of K-means and spectral clustering

7. CONCLUSION

We have addressed the problem of clustering across sources and proposed new bipartite and k-partite spectral clustering algorithms to solve them. Our experiments showed that the bipartite spectral clustering outperforms the regular spectral clustering to capture cross-source associations. Future work is to extend the method to hierarchical case by aggregating shot-level clustering to story level, and seek approaches to reduce the space requirement.

Acknowledgements. We thank Belle Tseng, Apostol Natsev, Milind Naphade, and Lexing Xie for valuable comments and discussions.

8. REFERENCES

[1] J.-M. Odobez, D. Gatica-Perez, and M. Guillemot, " Spectral Structuring of Home Videos," in Proc. Int. Conf. on Image and Video Retrieval (CIVR), Urbana-Champaign, Jul. 2003.

[2]L. Xie,S.-F. Chang, A. Divakaran and H. Sun, "Unsupervised Discovery of Multilevel Statistical Video Structures Using Hierarchical Hidden Markov Models", in ICME 2003.

[3] J. Allan, J.G. Carbonell, G. Doddington, J. Yamron and Y. Yang. "Topic Detection and Tracking Pilot Study Final Report", Proceedings of the Broadcast News Transcription and Understanding Workshop, Feb. 1998.

[4] A. Y. Ng, M. Jordan, and Y. Weiss, "On Spectral Clustering: Analysis and an algorithm", In NIPS 14,, 2002.

[5] I. S. Dhillon, "Co-clustering documents and words using bipartite spectral graph partitioning", in ACM SIGKDD Conference, August 26 - 29, 2001.

[6] H. Zha, X. He, C. Ding, M. Gu and H. Simon. "Bipartite Graph Partitioning and Data Clustering", Proceedings of ACM CIKM 2001, pp. 25-32, Nov. 5-10, 2001, Atlanta, Georgia.

[7] A.Y. Ng, A.X. Zheng and M. Jordan, "Stable algorithms for link analysis", In Proceedings of the Twenty-fourth Annual International ACM SIGIR, 2001.

[8] Y.Weiss, "Segmentation using eigenvectors: a unifying view", Proceedings of IEEE International Conference on Computer Vision p. 975-982 (1999)

[9] C-Y. Lin, B. L. Tseng, M. Naphade, A. Natsev and J.R. Smith, "VideoAL: A Novel End-to-End MPEG-7 Automatic Labeling System,"IEEE Intl. Conf. on Image Processing (ICIP), Barcelona, Spain, September, 2003.

[10] F. Harary, "Graph Theory", Reading, MA: Addison-Wesley, p. 23, 1994.

9. APPENDIX

The derivation of bipartite spectral clustering algorithm

Let $A = \begin{bmatrix} 0 & B \\ B^T & 0 \end{bmatrix}$, and $D = \begin{bmatrix} D_u & 0 \\ 0 & D_d \end{bmatrix}$

where $B \in R^{m \times l}, D_u \in R^{m \times m}, D_d \in R^{l \times l}, m + l = n$. Let $x = [x_u, x_d]^T$, and use the change-of-variable $x = D^{1/2} y$, then we can convert the eigenvalue problem to the generalized eigenvalue problem:

$$D^{-1/2} A D^{-1/2} x = lx \Rightarrow Ay = lDy$$

Let $y = [y_u, y_d]^T$ then $y_u = D_u^{-1/2} x_u, y_d = D_d^{-1/2} x_d$, we have

$$\begin{bmatrix} 0 & B \\ B^T & 0 \end{bmatrix} \begin{bmatrix} y_u \\ y_d \end{bmatrix} = l \begin{bmatrix} D_u & 0 \\ 0 & D_d \end{bmatrix} \begin{bmatrix} y_u \\ y_d \end{bmatrix}$$

which is equivalent to $By_d = lD_u y_u, B^T y_u = lD_d y_d$

therefore we have

$$D_u^{-1/2} B D_d^{-1/2} x_2 = l x_1 \text{ and } D_d^{-1/2} B^T D_u^{-1/2} x_1 = l x_2$$

Let $B_s = D_u^{-1/2} B D_d^{-1/2}$, then the equations correspond to the singular value problem $B_s x_1 = l x_2, B_s^T x_2 = l x_1$, where x_1 is called the left singular vector of B_s and x_2 is called the right singular vector. Note that B_s may not be square or symmetric matrix.