CO-CLUSTERING OF TIME-EVOLVING NEWS STORY WITH TRANSCRIPT AND KEYFRAME

XiaoWu,Chong-WahNgoandQingLi

Department of Computer Science, City University of Hong Kong

ABSTRACT

This paper presents techniques in clustering the same-topic news stories according to event themes. We model the relationship of stories with textual and visual concepts under the representation of bipartite graph. The textual and visual concepts are extracted respectively from speech transcripts and keyframes. Co-clustering algorithm is employed to exploit the duality of stories and textual-visual concepts based on spectral graph partitioning. Experimental results on TRECVID-2004 corpus show that the co-clustering of news stories with textual-visual concepts is significantly better than the co-clustering with either textual or visual concept alone.

1. INTRODUCTION

Clustering continuously reported or time-evolving stories in news videos is a critical step for tasks like news browsing, topic tracking and summarization. Previous researches [2, 3, 4] mostly focus on assembling news stories into few coarse classes such as politic, sport and health. For certain applications, a fine granularity of classification which can put together evolving stories according to event themes such as "Indian Ocean tsunami" and "Iraq weapon problem" is more attractive. This paper describes our techniques in identifying event themes across news sources and time spans by considering textual-visual concepts under the co-clustering framework.

In news videos, time-evolving stories describe the gradual changes of an event over time. Some concepts evolve slowly while others remain intact throughout the theme. These concepts can include keywords and keyframes that may repeatedly appear or slowly change into quite different fashions. The scenario is more complicated when considering the stories not just from one source but across different channels. For story clustering, the employment of either textual or visual concept may not be enough since either concept can appear differently over time. A robust way of clustering is to take into account both textual and visual concepts while exploiting the significance of these concepts.

In this paper, instead of managing multimedia information with one-way clustering, we propose a two-way clustering algorithm to effectively fuse textual and visual concepts in news. Two-way clustering, or namely co-clustering, is originally proposed in [2, 3] to exploit the duality of documents and words. Recently, this technique is also employed in [1, 5, 7] for image clustering [5], cross-source story clustering [7] and news topic labeling [1]. In [7], the story-level clustering is achieved by the co-clustering of two news sources with transcript similarity. In [1], the co-clustering of shots and words is described. While both approaches [1, 7] consider only textual information, this paper encodes both textual and visual concepts for co-clustering. The textual concepts are a list of words extracted from speech transcripts, while the visual concepts are the clusters of similar and near-duplicate keyframes extracted from video corpus.

2. BIPARTITE GRAPH MODEL

We model the relationship of stories and textual-visual concepts with a bipartite graph model. Denote $\boldsymbol{S}$, $\boldsymbol{T}$ and $\boldsymbol{V}$ respectively as a set of stories, textual and visual concepts. A bipartite graph is described as $\boldsymbol{G} = (\boldsymbol{P}, \boldsymbol{S}, \boldsymbol{E})$, where $\boldsymbol{P} = \boldsymbol{T} \cup \boldsymbol{V}$, $\boldsymbol{T} \cap \boldsymbol{V} = \emptyset$ and $\boldsymbol{P} \cap \boldsymbol{S} = \emptyset$, $\boldsymbol{V} = \{v_1, \ldots, v_g\}$, $\boldsymbol{T} = \{t_1, \ldots, t_m\}$, $\boldsymbol{S} = \{s_1, \ldots, s_n\}$ and $\boldsymbol{E}$ is a set of edges $\{\{s_i, p_j\}: s_i \in \boldsymbol{S}, p_j \in \boldsymbol{V} \text{ or } p_j \in \boldsymbol{T}\}$. In our case, $\boldsymbol{P}$ represents the concepts which include words and keyframe clusters. An edge $\{s_i, t_j\}$ exists if news story s_i contains word t_j, while an edge $\{s_i, v_j\}$ exists if news story s_i contains a keyframe in cluster v_j (note that $\boldsymbol{T} \cap \boldsymbol{V} = \emptyset$). In this model, there is no edge between stories, between keyframe clusters, and between words. The problem of co-clustering is to partition the bipartite graph into sub-graphs by considering the co-occurrence between stories and textual-visual concepts.

The work described in this paper was partially supported by a grant from City University of Hong Kong (Project No. 7001804) and a grant from the Research Grants Council of the Hong Kong Special Administrative region, China [Project No. CityU 1072/02E].

Email: {wuxiao, cwngo}@cs.cityu.edu.hk, itqli@cityu.edu.hk

3. SPECTRAL GRAPH PARTITIONING

Graph partitioning is known as a NP-complete problem. Nevertheless, the second smallest eigenvector of the *Laplacian mat rix* $\boldsymbol{L}$ is an approximate solution for bipartitioning a graph with the minimum normalized cuts. In the bipartite case,

$$\boldsymbol{L} = \begin{bmatrix} \boldsymbol{D}_1 & -\boldsymbol{A} \\ -\boldsymbol{A}^T & \boldsymbol{D}_2 \end{bmatrix} \text{ and } \boldsymbol{D} = \begin{bmatrix} \boldsymbol{D}_1 & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{D}_2 \end{bmatrix}$$

where $\boldsymbol{A}$ is the affinity matrix, $\boldsymbol{D}_1$ and $\boldsymbol{D}_2$ are diagonal matrices such that $D_1(i,i) = \sum_j A_{ij}$, $D_2(j,j) = \sum_i A_{ij}$. Let $\boldsymbol{u} = \boldsymbol{D}_1^{1/2}\boldsymbol{x}$ and $\boldsymbol{v} = \boldsymbol{D}_2^{1/2}\boldsymbol{y}$, the general eigenvalue problem $\boldsymbol{Lz} = \lambda\boldsymbol{Dz}$ can be written as:

$$\boldsymbol{D}_1^{-1/2}\boldsymbol{A}\boldsymbol{D}_2^{-1/2}\boldsymbol{v} = (1 - \lambda)\,\boldsymbol{u}$$
$$\boldsymbol{D}_2^{-1/2}\boldsymbol{A}^T\boldsymbol{D}_1^{-1/2}\boldsymbol{u} = (1 - \lambda)\,\boldsymbol{v}$$

which define the singular value decomposition of the normalized matrix $\boldsymbol{A}_\mathbf{n} = \boldsymbol{D}_1^{-1/2}\boldsymbol{A}\boldsymbol{D}_2^{-1/2}$. $(1 - \lambda)$ is the singular value, while $\boldsymbol{u}$ and $\boldsymbol{v}$ are the corresponding left and right singular vectors respectively. Therefore, the problem of computing the eigenvector correspond to the second smallest eigenvalue can be reduced to compute the left and right singular vectors based on the second largest singular value of $\boldsymbol{A}_\mathbf{n}$,

$$\boldsymbol{A}_\mathbf{n}\boldsymbol{v}_2 = (1 - \lambda)\boldsymbol{u}_2, \quad \boldsymbol{A}_\mathbf{n}^\mathbf{T}\boldsymbol{u}_2 = (1 - \lambda)\boldsymbol{v}_2,$$

from which we can see that $\boldsymbol{u}_2$ and $\boldsymbol{v}_2$ affect each other. That is, a partitioning of features can induce a partitioning of news stories, at the same time, a partitioning of news stories can conduce a partitioning of features.

In our case, we can get a bipartitioning of news stories by the right singular vector $\boldsymbol{v}_2$, while a bipartitioning of features by the left singular vector $\boldsymbol{u}_2$. The second eigenvector of $\boldsymbol{L}$ can be obtained by

$$\boldsymbol{z}_2 = \begin{bmatrix} \boldsymbol{D}_1^{-1/2}\boldsymbol{u}_2 \\ \boldsymbol{D}_2^{-1/2}\boldsymbol{v}_2 \end{bmatrix}$$

The classical k-means algorithm can be applied on $\boldsymbol{z}_2$ to minimize the normalized cuts, which approximates the optimal bipartitioning.

In order to get the k clusters, one can recursively apply the bipartition algorithm. But we can make good use of the $l = \lceil \log_2 k \rceil$ singular vectors $\boldsymbol{u}_2, \boldsymbol{u}_3, \ldots, \boldsymbol{u}_{l+1}$, and $\boldsymbol{v}_2, \boldsymbol{v}_3, \ldots, \boldsymbol{v}_{l+1}$, and form a l-dimensional data set:

$$\boldsymbol{Z} = \begin{bmatrix} \boldsymbol{D}_1^{-1/2}\boldsymbol{U} \\ \boldsymbol{D}_2^{-1/2}\boldsymbol{V} \end{bmatrix}$$

where $\boldsymbol{U} = [\boldsymbol{u}_2, \boldsymbol{u}_3, \ldots, \boldsymbol{u}_{l+1}]$ and $\boldsymbol{V} = [\boldsymbol{v}_2, \boldsymbol{v}_3, \ldots, \boldsymbol{v}_{l+1}]$. The classical k-means algorithm can then be applied to compute the optimal k partition. In general, the spectral clustering algorithm is a procedure which combines the normalized cuts with the k-means algorithm.

4. CO-CLUSTERING ALGORITHM

We treat words and keyframe clusters which are included in the news stories as concepts. The whole news story collection is represented by a concept-by-story matrix $\boldsymbol{A}$ whose rows correspond to concepts and columns to news stories. The matrix $\boldsymbol{A}$ is represented as

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{A}_1 \\ \boldsymbol{A}_2 \end{bmatrix}$$

$\boldsymbol{A}_1$ is a word-by-story matrix whose rows correspond to words and columns to news stories. $\boldsymbol{A}_2$ is a keyframe-by-story matrix whose rows correspond to keyframe clusters and columns to news stories. In our case, $\boldsymbol{A}_1$ is an m × n matrix, where m is the number of words and n is the number of news stories. $\boldsymbol{A}_2$ is a g × n matrix, where g is the number of keyframe clusters and n is the number of news stories.

Our co-clustering algorithm which considers textual and visual concepts consists of the following steps:

1. Set up word-by-story matrix $\boldsymbol{A}_1$ and keyframe-by-story matrix $\boldsymbol{A}_2$.
2. Construct the matrix $\boldsymbol{A}$.
3. Calculate $\boldsymbol{A}_\mathbf{n} = \boldsymbol{D}_1^{-1/2}\boldsymbol{A}\boldsymbol{D}_2^{-1/2}$.
4. Compute $l = \lceil \log_2 k \rceil$ singular vectors of $\boldsymbol{A}_\mathbf{n}$, $\boldsymbol{u}_2$, $\boldsymbol{u}_3, \ldots, \boldsymbol{u}_{l+1}$, and $\boldsymbol{v}_2, \boldsymbol{v}_3, \ldots, \boldsymbol{v}_{l+1}$, and form the matrix $\boldsymbol{Z}$.
5. Run the k-means algorithm on the l-dimensional data $\boldsymbol{Z}$ to obtain the desired k-way multipartition.

5. MODELING TEXTUAL&VISUAL CONCEPTS

The concept-by-story matrix $\boldsymbol{A}$ is constructed by modeling the importance of textual and visual concepts. First, we extract the story boundaries from the segmentation ground truth data in the development set of TRECVID [4]. Then we calculate the association between a news story j and a word i by

$$W_{ij} = tf_{ij} \times idf_i$$

where tf_{ij} is the term frequency of word i in story j, and idf_i is the inverse document frequency of word i. These story-word associations form the matrix $\boldsymbol{A}_1$. For keyframes, color histograms are extracted, and k-means algorithm is applied to cluster the keyframes into g groups. We regard the keyframes in one group representing one visual concept since they are mostly similar or nearly duplicate keyframes. The matrix $\boldsymbol{A}_2$ is constructed by computing story-keyframe association

$$W_{ij} = kf_{ij} \times log_2(N/sf_i)$$

where kf_{ij} is the frequency of keyframe group i in story j; N is the number of news story; and sf_i is the story frequency of keyframe group i. Finally, the (m+g)×n concept-by-story matrix $\boldsymbol{A}$ is formed with A_{ij} equals the edge-weight W_{ij}. We order the vertices so that the

first m vertices index the words and the last g vertices index the keyframe clusters.

Notice that the concept-by-story matrix $\boldsymbol{A}$ is actually composed of $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$. To compute the co-clustering of stories and textual concepts, we only need the word-by-story matrix $\boldsymbol{A}_1$. Similarly, the co-clustering of stories and visual concepts only requires the keyframe-by-stories matrix $\boldsymbol{A}_2$. In our case, we fuse both textual and visual concepts in $\boldsymbol{A}$ while explicitly computing the significance of concepts as edge weights for co-clustering.

6. EXPERIMENTAL RESULTS

6.1. Data Set and Performance Metric

We test our performance on TRECVID-2004 [4]. We select a week's videos from 1998-03-23 to 1998-03-30 which include 14 videos from CNN and ABC news as our test set. After data preprocessing such as word stemming, stop-words and stop-frame removal[1], the data set consists of 174 new stories, 1653 keyframes and 3180 words. We build a ground-truth table by manually labeling stories according to event themes. In total, there are 22 news themes with more than one news story (See Table 1). To ensure the fairness of comparison, we omit the theme that has only one news story from experiment.

The color histogram of keyframe is represented in HSV color space. Hue is quantized into 18 bins while saturation and brightness are quantized into 3 bins respectively. Such quantization provides 162 (18 × 3 × 3) distinct color sets. In order to compute the story-keyframe association, we run the k-means algorithm based on color histogram to cluster the 1653 keyframes into g=400 groups. We attempt g=[100,1000] in the experiment and find that g is indeed not sensitive to the final results of co-clustering. This is mainly due to the use of kf_{ij} (frequency of keyframe group i in story j) and sf_i (story frequency of keyframe group i) to represent the importance of a keyframe group. When g is large, we have more groups. By co-clustering, these groups usually rearrange into few groups which are actually the groups when the value of g is small.

We use *Fmeasure* [6] as the metric for performance evaluation. F-measure assesses the quality of clusters by comparing the detected clusters with the ground-truth clusters. Let $\mathcal{G}$ be the set of ground-truth clusters and $\mathcal{D}$ be the detected ones. The F-measure FM is

$$\mathbf{FM} = \frac{1}{\boldsymbol{H}} \sum_{\boldsymbol{C}_i \in \mathcal{G}} |\boldsymbol{C}_i| \max_{\boldsymbol{C}_j \in \mathcal{D}} \{F(\boldsymbol{C}_i, \boldsymbol{C}_j)\}$$

$$F(\boldsymbol{C}_i, \boldsymbol{C}_j) = \frac{2 \times Recall(\boldsymbol{C}_i, \boldsymbol{C}_j) \times Precision(\boldsymbol{C}_i, \boldsymbol{C}_j)}{Recall(\boldsymbol{C}_i, \boldsymbol{C}_j) + Precision(\boldsymbol{C}_i, \boldsymbol{C}_j)}$$

where $Recall(\boldsymbol{C}_i, \boldsymbol{C}_j) = |\boldsymbol{C}_i \cap \boldsymbol{C}_j| / \boldsymbol{C}_i$ and $Precision(\boldsymbol{C}_i, \boldsymbol{C}_j) = |\boldsymbol{C}_i \cap \boldsymbol{C}_j| / \boldsymbol{C}_j$. The term $\boldsymbol{H} = \sum_{\boldsymbol{C}_i \in \mathcal{G}} |\boldsymbol{C}_i|$ is a normalized constant that indicates the sum of stories from each i^{th} cluster $\boldsymbol{C}_i$. The value of FM ranges from 0 to 1. The higher FM is, the better the clustering performance is.

6.2. Performance Comparison

We compare three co-clustering approaches: with texture-visual concepts (CC_WORD_KF), with textual only (CC_WORD) and with visual only (CC_KF). Figure 1 shows the performance in term of F-measure vs. the number of clusters. In overall, CC_WORD_KF has better performance than CC_WORD and CC_KF. As the number of clusters (k) increases, our algorithm improves and achieves the highest FM value when k=16. In overall, CC_WORD_KF can correctly cluster most related evolving stories into the same-theme event clusters. CC_WORD and CC_KF, in contrast, partition the related news stories into different clusters. Their performance is slightly improved when k approaches 22. The F-measure of CC_WORD_KF is 1.86 and 2.38 times higher than CC_WORD and CC_KF when the number of clusters is 16.

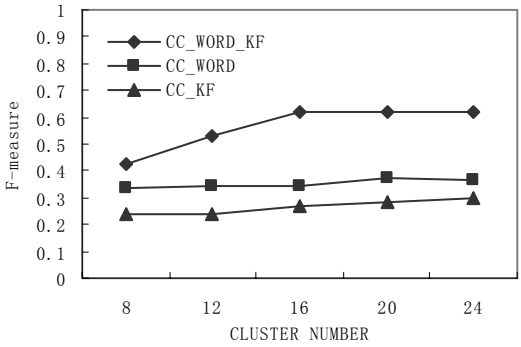

Figure 1. Overall performance comparison

Table 1 lists the twenty two ground-truth clusters along with the recall and precision of each method. As indicated in the table, CC_WORD_KF can correctly cluster the evolving events such as "Arkansas school shooting", "Clinton visited Africa" and "Yeltsin fired the whole cabinet". Furthermore, it has better precision and recall than CC_WORD and CC_KF in general. For instance, in the theme "Arkansas school shooting" that has most stories, the precision and recall of CC_WORD are 15/21 and 15/33 respectively. Though CC_KF has better recall, its precision is low (19/81), from which we can see that the "Arkansas school shooting" event is clustered with other events and form a larger cluster with 81 stories. CC_WORD_KF,

[1] Stop-frame is the keyframe with anchor person. Similar to stop-words, stop-frames usually appear in most stories but carry no information for clustering.

Table 1. Performance comparison when the number of clusters equal to 16 (#: number of stories).

Events			CC_WORD_KF		CC_WORD		CC_KF	
ID	**Event Theme**	**#**	**Prec**	**Recall**	**Prec**	**Recall**	**Prec**	**Recall**
1	Weather	6	6 / 6	6 / 6	1 / 7	1 / 6	4 / 4	4 / 6
2	Basketball	20	14 / 21	14 / 20	19 / 91	19 / 20	3 / 3	3 / 20
3	Baseball	4	3 / 21	3 / 4	3 / 4	3 / 4	1 / 1	1 / 4
4	Golf	7	2 / 3	2 / 7	7 / 91	7 / 7	2 / 6	2 / 7
5	Ice	5	4 / 5	4 / 5	5 / 91	5 / 5	4 / 25	4 / 5
6	Stock information	10	6 / 16	6 / 10	8 / 91	8 / 10	4 / 4	4 / 10
7	Finance information	15	8 / 19	8 / 15	14 / 91	14 / 15	5 / 34	5 / 15
8	Natural disasters	8	6 / 43	6 / 8	8 / 91	8 / 8	1 / 1	1 / 8
9	Air crash	3	3 / 4	3 / 3	3 / 91	3 / 3	1 / 25	1 / 3
10	Pennsylvania cabin fire	2	2 / 43	2 / 2	1 / 21	1 / 2	2 / 81	2 / 2
11	Yeltsin fired the whole cabinet	6	6 / 8	6 / 6	5 / 15	5 / 6	1 / 1	1 / 6
12	Clinton visited Africa	19	19 / 26	19 / 19	6 / 15	6 / 19	9 / 81	9 / 19
13	Africa related report	7	4 / 19	4 / 7	2 / 2	2 / 7	4 / 34	4 / 7
14	Clinton sexual scandal	7	4 / 11	4 / 7	3 / 7	3 / 7	2 / 25	2 / 7
15	Arkansas school shooting	33	32 / 43	32 / 33	15 / 21	15 / 33	19 / 81	19 / 33
16	Iraq weapon problem	5	5 / 11	5 / 5	2 / 7	2 / 5	5 / 81	5 / 5
17	Hospital employee killed patients	5	5 / 6	5 / 5	4 / 21	4 / 5	2 / 34	2 / 5
18	Florida Judy was found guilty	2	2 / 2	2 / 2	2 / 91	2 / 2	1 / 25	1 / 2
19	Nicholas sentence	2	2 / 2	2 / 2	2 / 91	2 / 2	1 / 25	1 / 2
20	Homosexual	3	2 / 8	2 / 3	1 / 7	1 / 3	1 / 25	1 / 3
21	Dentist refused HIV patients	2	2 / 11	2 / 2	1 / 7	1 / 2	2 / 81	2 / 2
22	Transportation bill	3	3 / 26	3 / 3	2 / 2	2 / 3	1 / 3	1 / 3
F-measure			**0.643**		**0.346**		**0.270**	

Group 1 Group 2 Group 3 Group 4 Group 5 Group 6 Group 7

Figure 2. Keyframe cluster of event ID-11 (This figure shows the centroids of the keyframe groups)

nevertheless, improves the precision (32/43) and recall (32/33) significantly by combining both textual and visual concepts.

In addition to the superior performance in clustering, CC_WORD_KF can also group words and keyframe clusters that are mostly meaningful and representative to the event themes. Table 2 lists the top 10 words for the theme "Yeltsin fired the whole cabinet". Figure 2 shows the corresponding visual concepts represented by keyframe clusters.

Table 2. Word cluster of event ID-11

Minister	Russian	Boris	Project	Yeltsin
Cabinet	Fire	Parliament	Post	Prime

7. CONCLUSIONS

We have presented our approach in integrating visual and textual concepts for grouping time-evolving news stories by co-clustering framework. Experimental results on TRECVID 2004 data set indicate that our approach outperforms the co-clustering algorithm with either textual or visual information. Though encouraging, further analysis such as how to select the best number of clusters and how to improve precision-recall by other concepts (e.g., audio, motion) is still required.

REFERENCES

[1] P. Duygulu, J-Y, Pan and D and A. Forsyth, "Towards Auto-Documentary: Tracking the Evolution of News Stories", in *ACMMM'04* , Oct. 2004, pp. 820-827.

[2] I. S. Dhillon, "Co-clustering Documents and Words Using Bipartite Spectral Graph Partitioning", in *ACM SIGKDD'01*, Aug. 26-29, 2001, pp. 269-274.

[3] I. S. Dhillon, S. Mallela, and D. S. Modha, "Information-theoretic Co-clustering", in *ACM SIGKDD'03*, Washington, USA, Aug 24-27, 2003.

[4] W. Kraaij, A. F. Smeaton, and P. Over, "TRECVID 2004 - An Introduction", 2004.

[5] G. Qiu, "Image and Feature Co-clustering", in *ICPR'04*, 2004, pp. 991-994.

[6] M. Steinbach, G. Karypis and V. Kumar, "A Comparison of Document Clustering Techniques", in *Proc.ofKDDWorkshoponTextMining* , 2000.

[7] D-Q. Zhang, C-Y. Lin, S-F. Chang and J. R. Smith, "Semantic Video Clustering Across Sources Using Bipartite Spectral Clustering", in *ICME'04*, Jun. 2004.