A Lexicon-Guided LSI Method for Semantic News Video Retrieval*

Juan Cao[1, 2], Sheng Tang[1], Jintao Li[1], Yongdong Zhang[1], and Xuefeng Pan[1,2]

[1] Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100080, China
[2] Graduate University of the Chinese Academy of Sciences, Beijing 100039, China
{caojuan, ts, jtli, zhyd, xfpan }@ict.ac.cn

Abstract. Many researchers try to utilize the semantic information extracted from visual feature to directly realize the semantic video retrieval or to supplement the automated speech recognition (ASR) text retrieval. But bridging the gap between the low-level visual feature and semantic content is still a challenging task. In this paper, we study how to effectively use Latent Semantic Indexing (LSI) to improve the semantic video retrieval through the ASR texts. The basic LSI method has been shown effective in the traditional text retrieval and the noisy ASR text retrieval. In this paper, we further use the lexicon-guided semantic clustering to effectively remove the noise introduced by news video's additional contents, and use the cluster-based LSI to automatically mine the semantic structure underlying the terms expression. Tests on the TRECVID 2005 dataset show that the above two enhancements achieve 21.3% and 6.9% improvements in performance over the traditional vector-space model(VSM) and the basic LSI separately.

Keywords: ASR text, LSI, Semantic video retrieval.

1 Introduction

In video retrieval, the users' need is not only the visually similar content, but the semantic similar content. So low-level features are now becoming insufficient to build efficient news video retrieval systems. Many works have done to bridge the gap between low-level features and semantic content, but this is still a challenging task in the future. In this paper, we try an alternative way to mine the video's semantic information from its automatic speech recognition (ASR) text. One reason is that ASR text is the direct semantic description about video content. The other is that the technologies of text retrieval are more mature than the ones of visual features process. But news video's ASR texts have their own characteristics, and we should adopt the traditional information retrieval methods, LSI, for it.

* This paper is supported by National Basic Research Program of China (973 Program, 2007CB311100), Beijing Science and Technology Planning Program of China (D01060-08040291) and National High Technology and Research Development Program of China (863 Program, 2007AA01Z416).

H.H.S. Ip et al. (Eds.): PCM 2007, LNCS 4810, pp. 187–195, 2007.

Many works have done to prove the effectiveness of LSI in common text retrieval [5][9][13]. Moreover, L.Hollink et al. demonstrated the feasibility of the basic LSI method in the noisy ASR texts[1]. In TRECVID 2005 ASR texts, the word error rate(WER) is 33.8%[14], and the average length of shot document is about 14.5, while the counterpart in TREC collection is 84.6(after pre-processing) [13]. These data are a challenge to the basic LSI. By enhancing the basic LSI to fit ASR texts, we further affirmed the validity of the LSI approach in video's ASR texts retrieval.

1.1 Related Work

In semantic video retrieval field, much of prior work focused on extracting semantic information from videos' multi-modality resources to supplement the video's ASR text. Neo[2] integrates event-related high-level features (HLFs) to provide the additional context and knowledge. The event-related HLFs are relevant visual features detected by a machine learning approach. The improvement of the performance is significant. But the HLF detectors' training need a large scale annotated corpus, and the detection confidence is unstable when the test collection is inconsistent with the training collection. Many researchers try to use the data mining technology to automatically mine the semantic information directly from the video's low-level visual features. In [3] and [4], authors perform LSI separately on the region-level and low-level visual features, and try to automatically bridge the gap from low-level visual features to the semantic content. But these approaches need a rigorous experimental environment: [3] assume that a video shot is well represented by its key frames. In [4]'s experiment, the corpus must be standard and contain limited semantic categories. However, all these assumptions are unpractical.

1.2 Our Work

We parse a corpus in three level hierarchical structures, initially keywords(term-document matrix), topics(LSI semantic space) and semantic clustering, and retrieve under this structure.

Generally, ASR texts include two main errors: one is the word error imported by speech recognition step, and the other is the matching error induced when mapping the ASR text to the video's shot. To reduce the negative influence imported by errors, we retrieve the text by using LSI approach instead of using the common term-matching algorithm. LSI can map the keywords space to a reduced-dimension semantic concept space and ignore the unimportant details. These features make the LSI very compelling and useful in semantic retrieval.

Besides, the video's shot ASR text is very short (in our experiment, the shot text's average length is 14.5). Much important information may appear as rarely as the real noises do. Furthermore, news videos not only include the real-life events shots (valuable news content), but also include some additional shots (redundant content), such as led-in/out, advertisement, special graphic effects, etc. So we mine semantic structure in this impure corpus is unadvisable. We utilize lexicon–guided semantic clustering to remove the redundant contents in news video, and perform LSI in the semantic class instead of the whole document collection. As confirmed by our experiments, the introduction of lexicon can get an improvement of 6.9%.

The rest of this paper is organized as follows: In section 2, we give a brief review of LSI, and describe its importance to ASR texts retrieval. Next we present the lexicon-guided semantic clustering and the cluster-based LSI in the section 3. Then we set up the experimental framework to compare the adaptive LSI to traditional methods in section 4 and discuss two open questions in section 5. Finally, conclusion and future work are given in the last section.

2 Latent Semantic Indexing

2.1 LSI

In natural langue analyzing field, a certain word can be interpreted in different ways within different contexts (polysemy); while the same concept can be described using different terms(synonymy). LSI was proposed to solve these problems. The key idea is clustering the **co-occurring keywords**, and mapping documents and queries into a lower dimensional space(latent semantic space)[5]. The advantage of retrieval in this space is that a query can be similar to a document even when they share no words.

The LSI technique makes use of the singular value decomposition(SVD) to mine the total collection's semantic structure. Firstly, we represent the document collection as a term-document matrix $M_{t\times d}$, then the SVD decompose $M_{t\times d}$ into the product of three matrices:

$$M_{t\times d} = T_{t\times r} S_{r\times r} \left(D_{\mathrm{d}\times \mathrm{r}}\right)^T \tag{1}$$

where t is the number of terms, d is the number of documents. r is the rank of M. T and D are the matrices with orthonormal columns. S is a diagonal matrix of M's singular values sorted in decreasing. The singular value is larger, the corresponding dimension is more important. By restricting the matrixes T,S,D to their first k rows , we can get a approximate matrix M':

$$M'_{t\times d} = T'_{t\times k} S'_{k\times k} \left(D'_{\mathrm{d}\times k}\right)^T \tag{2}$$

Where k< r **.**

Discarding the less important dimensions can remove much of the noise, and transform the term space to the reduced-dimension latent semantic space.

In retrieval period, through the formula(3) we transform the q to latent semantic space:

$$q' = q^T T'_{t\times k} S^{-1}{}_{k\times k} \tag{3}$$

We use the standard cosine measure(4) to compute the similarity between the query and document:

$$LSI\left(q, doc_i\right) = \frac{q' \bullet \left(D'\right)_i}{\left|q'\right|\left|\left(D'\right)_i\right|} \tag{4}$$

$\left(D'\right)_i$ is the i-th column of matrix D'.

2.2 LSI in ASR Texts

In the ASR text retrieval, the contribution of LSI is not limited in polysemy and synonymy problems, it can reduce the negative effect brought by speech recognition errors. LSI reduces the importance of the individual terms, and pays more attention to the latent structure underlying the whole document collection. So the individually term recognition error can not influence the retrieval precision in LSI.

Furthermore, LSI is a completely automatic unsupervised statistical learning approach, and is free of hand-labeled training data. This factor overcomes the drawback that the confidence depends on the training collection. This problem lies in most of the traditional machine learning approaches used to extract HLFs from visual feature, and hinders the HLFs' application in semantic video retrieval.

The other problem in practical application is the efficiency. [5] shows that decomposing a matrix with 70,000 documents and 90,000 terms requires about 18 hours of CPU time on a SUN SPARCstation 10 workstation. This cost is a too horrendous to bear. Note that the term-document matrix is quite sparse. In our experiment, the matrix contains only 0.17% non-zero entries. So we select the SVDPACK[6] to quickly compute the SVD. This software package implements Lanczos and subspace iteration-based methods for determining several of the largest singular triplets (singular values and corresponding left- and right-singular vectors) for large sparse matrices. For a matrix with the size of 20932×8410, only require about 1 minute of CPU time on a P4 PC. Meanwhile our semantic clustering in next section will be used to reduce the dimension of the matrix, and to further increase the performance efficiency.

3 Lexicon Guided Semantic Cluster

Based on the characteristics of news videos, we predefined four semantic classes: politics, sports, finance, and general [10]. The general class is designed to remove the redundant shots described in the above section. The initial cluster centroids are pre-constructed pseudo samples based on the HowNet's relevance calculator[7][11]. Then we use the k-means approach to continually adjust the initial cluster centers to really reflect the current collection's characteristics. The detail process is as follows:

- In step1, we utilize the HowNet's relevance calculator to separately produce three relevant word lists for politics, sports and finance. Then we transform these lists to term vector space of the Term-Document Matrix $M_{txd.}$. These vectors are the initial cluster centers $Center = \{C_0, C_1, C_2, C_3\}$. Where C_0 is the initial center of general class, and we set it's values with zero.
- In step2, we use the iterative computation to adjust the pseudo centers to the real corpus, and produce the final cluster centers : $Center' = \{C_0', C_1', C_2', C_3'\}$.
- In step3, we classify the corpus to four classes based on $Center'$.
- In step4, by measuring the distance between the query and the three new cluster centers C_1', C_2', C_3', we get the scores of the query belonging to the corresponding semantic classes: α_1, α_2, α_3.

– In step5, we separately perform the LSI analysis in three semantic classes, and fuse the results based on the scores in step4:

$$Sim(q, D) = \sum_{k=1,2,3} \alpha_k LSI_k(q, D_k) \tag{5}$$

By deeply analyzing the original data and the classified results, we found that the general class can effectively remove the additional shots. In the 20932 documents, only 2228 shot documents are clustered into general class, and these snippets are very short, or their contents are irrelevant to any news classes.

The comparison experiments tell us that classifying the document collection only based on the HowNet's cluster centers or the k-means centers performs worse than the above approach. Because the HowNet's cluster centroids are produced by its corpus and rules, and the terms' distribution is somewhat different from the current collection. So the strategy of using lexicon to guide the k-means clustering is reasonable and hence effective.

4 Empirical Validation

We choose the TRECVID 2005's English test collection as our experimental data. After pre-processing, the total number of the shot documents is 20932, the number of the unique terms is 8410, and 305529 non-zero entries in the term-document matrix.

We select 12 topics from the 2005 search task to test our methods, and the choice criterion is the number of the relevant documents in the test collection bigger than 50. Table.1 is the topics' information:

149="Find shots of Condoleeza Rice";
160="Find shots of something (e.g., vehicle, aircraft, building, etc) on fire with flames and smoke visible";
161="Find shots of people with banners or signs";
162"Find shots of one or more people entering or leaving a building";
163="Find shots of a meeting with a large table and more than two people";
164="Find shots of a ship or boat";
165="Find shots of basketball players on the court";
166="Find shots of one or more palm trees";
168="Find shots of a road with one or more cars";
169="Find shots of one or more tanks or other military vehicles";
170="Find shots of a tall building (with more than 5 floors above the ground)";
172="Find shots of an office setting, i.e., one or more desks/tables and one or more computers and one or more people";

Table 1. Statistics of topics

topic	149	160	161	162	163	164	165	166	168	169	170	172
Rel-Num	72	90	255	162	95	64	130	159	548	286	135	487
Pre-Rank	9	17	12	22	14	5	13	18	7	11	19	16

Where the Rel-Num is the topics' relevant documents number in the test collection; and the Pre-Rank is the topics' mean average precision ranking among the 24 topics, the average ranking is 13.5.

The version of HowNet we used in our experiments includes 158849 words.

4.1 Evaluation Method

To determine the accuracy of the methods' retrieval results, we use *average precision*, following the standard in TRECVID evaluations. Let $L^k = \{l_1, l_2, ..., l_k\}$ be a ranked version of the retrieval results set S. At any given rank k let R_k be the number of relevant shots in the top k of L, where R is the total number of relevant shots in a set of size S. Then average precision is defined as:

$$\text{Average Precision} = \frac{1}{R}\sum_{k=1}^{|S|}\frac{R_k}{k} f(l_k) \tag{6}$$

Where function $f(l_k) = 1$ if $l_k \in$ R and 0 otherwise. The average precision favors highly ranked relevant shots.

We use the truths provided by NIST to evaluate our methods. But we found that the truths are only subsets of the whole test collection, and many of the documents returned by our system were not judged for relevance. The same obstacle of evaluation is also encountered by[9]. So we only evaluate the performance of submitted results with relevance judgments.

4.2 Pre-processing

Firstly, we matched the ASR texts with the shots based on the shot temporal boundaries, and expanded the shot boundaries to include up to 3 immediate neighbors on either side to compensate for speech and visual misalignment.

Then we remove the stopwords using the SMART's English stoplist. Additionally, we computed all the terms' document frequency (DF) occurred in the whole collection, and add the terms with highest DF to the user-defined stop word list.

To decrease the matrix total size and to enhance the efficiency, we extract only nouns and verbs. This step can decrease the term dimension from 8410 to 7159.

We also used the Porter's well-established stemming algorithm [8] to unify terms, which allows several forms of a word to be considered as a unique term.

4.3 Primary Results

In our experiment, we realized three runs to compare our method's effect. Where:

Run1: the vector space model(VSM)
Run2: LSI
Run3: lexicon-guided cluster + LSI

The three methods have the same pre-processing. In Run2 the dimension of LSI is 306, and in Run3 the dimensions of three LSI respectively are 206, 206 and 214.

Fig. 1 displays the performance curves for the retrieval average precisions of three methods for 12 topics.

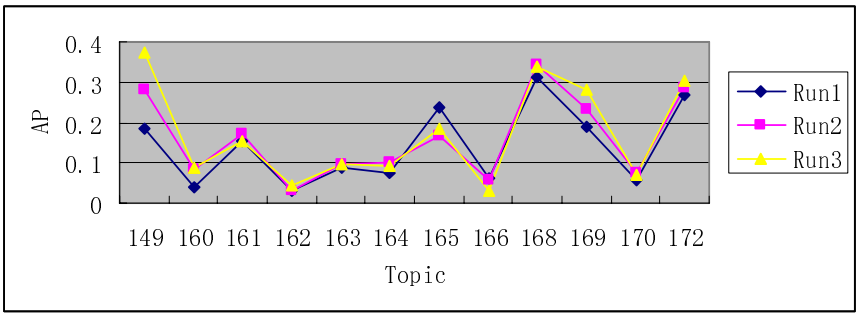

Fig. 1. Average precision of three methods

Table.2 displays the mean average precision(MAP) of the three runs:

Table 2. Evaluation Results in MAP

Runs	Run1	Run2	Run3
MAP	0.141	0.160	0.171

In Run2, we noted that applying LSI in semantic retrieval can increase the performance by about 13.5% than VSM, and Run3 shows that the lexicon-guided semantic cluster can increase the performance by about 6.9% than LSI.

5 Discussion

On the deeper analysis, we find that LSI is designed as a recall enhancing method by expanding the terms in retrieving process. The enhancement can easily find in the second row of Table.3. But the other result of expansion is that there are highly ranked but irrelevant documents in the LSI's result list(Run2 in Top10 is 1.08, and Run1 in Top10 is 1.16). The lexicon-guided semantic cluster can partially alleviate this problem(the Run3 in Top10 is 1.75). But the hits in the top ranking is still too low, and we'll seek some more effective approach to distinguish what should be expanded from what should not be expanded, such as the syntactic or statistically-derived phrases, part of speech parsing, etc.

Table 3. The average hits at depths 10,50,100,300 and 500

	Top10	Top50	Top100	Top300	Top500
Run1	1.16	6.66	11.33	35.83	58.58
Run2	1.08	7.33	15.75	42.91	69.91
Run3	1.75	8.25	16	33.83	36.67

Where the scores are the average hits of 12 topics at corresponding depths.

Besides, choosing the appropriate number of dimensions for the LSI representation is an open research question. In our experiments, we set k with different values such as 100, 200, 300, 500, 800 and 1000. We find that the performance improves as k increases for a while, and then decreases, and reach the best results on from 200 to 300.

6 Conclusion

In this paper, we proposed a method of using lexicon knowledge to guide the semantic clustering, and using semantic cluster to restrict the LSI's expansion. The experiments show that the proposed approach can improve the retrieval performance significantly. Prior works have proved the validity of the basic LSI in common texts retrieval, and our works further affirmed the validity of the enhanced LSI approach in video's ASR texts retrieval. In the future we'll try to fuse multi-modality video features to a uniform representation space, and to learn the semantic in this space.

References

1. Hollink, L., Nguyen, G.P., Koelma, D.C., Schreiber, A.T., Worring, M.: Assessing user behaviour in news video retrieval. IEE proceedings on Vision, Image and Signal Processing, 911–918 (2005)
2. Neo, S-Y., Zhao, J., Kan, M-Y., Chua, T-S.: Video Retrieval Using High-level features: Exploiting Query-matching and Confidence-based Weighting. In: Sundaram, H., Naphade, M., Smith, J.R., Rui, Y. (eds.) CIVR 2006. LNCS, vol. 4071, Springer, Heidelberg (2006)
3. Souvannavong, F., Merialdo, B., Huet, B.: Latent Semantic Indexing for Semantic Content Dectection of Video Shots. In: ICME 2004, Taipei, Taiwan (June 27th - 30th, 2004)
4. Zhao, R., Grosky, W.I.: From Features to Semantics: Some Preliminary Results. In: ICME 2000, New York, USA (July 30th - August 30th, 2000)
5. Rosario, B.: Latent Semantic Indexing: An overview [A]. INFOSYS 240 (Spring 2000)
6. Berry, M.W.: SVDPACK: A Fortran-77 software libray for the sparse singular value decomposition. Technical Report CS-92-159, University of Tennessee, Knoxville, TN (June 1992)
7. Dong, Z., Dong, Q.: HowNet, http://www.keenage.com/
8. Porter, M.F.: An Algorithm for Suffix Stripping Program, vol. 14, pp. 130–137 (1980)

9. Dumains, S.T.: Latent Semantic Indexing(LSI)and TREC-2[C]. In: Harman, D. (ed.) The Second Text Retrieval Conference(TREC2).National Institute of Standards and Technology Special Publication, pp. 105–116 (1994)
10. Chua, T.-S., Neo, S.-Y., Goh, H.-K., Zhao, M., Xiao, Y., Wang, G.: TRECVID 2005 by NUS PRIS. In: TRECVID 2005, NIST, Gaithersburg, Maryland, USA (November 14-15, 2005)
11. Qun, L., Su-Jian, L.: Computing word similarities based on IqowNet? Computational Linguistics and Chinese Language Processing? 7(2), 59–76 (2002)
12. Cao, J., Li, J., Zhang, Y., Tang, S.: A Novel Method for Spoken Textual Feature Extraction in Semantic Video Retrieval. In: Zhuang, Y., Yang, S., Rui, Y., He, Q. (eds.) PCM 2006. LNCS, vol. 4261, Springer, Heidelberg (2006)
13. Dumais, S.: Latent Semantic Indexing (LSI): TREC-3 Report. In: Harman, D. (ed.) The Third Text Retrieval Conference, National Institute of Standards and Technology Special Publication 500–226, pp. 219–230 (March 1994)
14. Garafolo, A., Voorhees, E.M.: The TREC Spoken Document Retrieval Track: A Success Story. In: Proceedings of the RIAO 2000 Conference: Content-Based Multimedia Information Access, Paris, pp. 1–20 (2000)