CMC-FZU @ TRECVID 2010: Semantic Indexing

Jianjun Huang, Liao Youqiang, Ji Changyu, Chen Shu, Zheng Qibiao, Lin Yulian, Wu Shencheng

College of Mathematics & Computer Science, Fuzhou University
Fuzhou, Fujian 350108, P.R. China

Abstract

This paper summarizes our approaches to the semantic indexing task in TRECVID 2010. Several fusion strategies are employed in our detection system, including a low-level feature concatenation, text retrieval fusion, and ontology relation fusion. It can be drawn that the fusion of text retrieval results with a baseline in a proper way can contribute greatly to the system performance. Our approach to ontology relation fusion increases the number of the inferred true shots returned. Two of our runs using fusion schemes have the best infAP for a feature among the 30 features evaluated. We describe our text and ontology relation fusion schemes in detail in this paper. The description of our submission runs for the semantic indexing is as follows:

ID	Description
F_A_Fuzhou_Run1_1	The fusion outcome of the meta data text retrieval result with the svm prediction.
F_A_Fuzhou_Run2_2	This run is the result of the ontology relation fusion using Run1 as its baseline result.
F_A_Fuzhou_Run3__3	This is the direct result of svm prediction with a mixed feature.
L_A_Fuzhou_Run4_4	The fusion outcome of the meta data retrieval result and the svm result with a mixed feature.

1. Introduction

In TRECVID 2010 semantic indexing task, 130 concepts had been selected. Each team was expected to submit a maximum of 4 runs. Two types of submissions would be considered: "full" in which participants were required to provide a result for all the proposed 130 concepts and "light" which participants were required to provide a result only for a predefined list of 10 concepts. Our group participated in the semantic indexing task, and submitted 4 runs including three runs of “full” type and one run of “light” type. We present an overview of our participation and algorithms in the semantic indexing task, including annotation, key frame extraction, feature representation, learning, fusion and final results.

2. Key Frame Extraction and Annotation

Key frame extraction is a preliminary step for video content analysis and retrieval. We use the key frames provided by LIG (Laboratoire d'Informatique de Grenoble) on the development data set (IACC.1.tv10 training). On the test set (IACC.1.A), key frames were to be extracted by each participant. Since a number of videos had been removed from the TV 2010 test collection, we derived 8384 valid mp4 videos from the test set. To avoid a large set of key frames, we extracted only one key frame per shot for each valid video using a solution based on libavcodec/ffmpeg. The time point data of video segments are extracted from given MPEG-7 files. We found the original master shot ref mp7 files had abnormal tags and corrected the format errors before we used those files to extract key frames. We extracted totally 144988 key frames from the IACC.1.A data collection.

As in the previous year, our group participated in the collaborative annotation for TRECVID 2010 and did more than minimum 30.000 key frame annotations required. The annotation data on the IACC.1.tv10.training were used in our systems.

3. Low-level Feature Extraction

There were six low-level features we extracted from the key frames: Color Moments, Color Histogram, Color Coherence Vector, Auto Correlagram, Canny Edge Histogram and Wavelet texture. Due to limitation

in time and calculation, we dropped the former three features, and only used other three features to build svm models. In our experiments, we actually merged them into a single feature. The mixed feature was derived from the direct concatenation of three basic features: Auto Correlagram and Canny Edge Histogram, and Wavelet texture. Our baseline run Fuzhou_Run3__3 was a direct svm result using the fused feature.

4. Training and Fusion

All our svm models are trained on IACC.1.tv10 training data. The training data were created by sampling all positive concept examples and a quarter of negative examples from the collaborative annotation. Since there are 130 concepts to be detected, the model training processing is quite time consuming. Due to limitation in time, we finally used models trained with cost factor c equal 0.25 and gamma parameter g set to 0.25 in the RBF kernel function, and dropped other models.

In the fusion stage, we use the above SVM result as a baseline. Both text retrieval fusion and ontology relation fusion are employed to enhance the final results. In TRECVID 2010 there is a metadata file associated with each video in both test and development sets, we try to make use of these mpeg-7 files in our detection system. In the text retrieval, we use info automatically derived from labels like title, keywords, and descriptions in the metadata mp7 file. Since the text retrieval result only tell the confidence of the video that might contain a semantic concept, so we have to assign a score to each shot in the video. In our experiment, we use the following score formula to calculate the shot score:

$$f(i,k) = M_k - \frac{2(M_k - m_k)}{s_k - 1} | i - \frac{s_k + 1}{2} |;$$

$$M_k = \frac{p_k}{n \max\{p_1, p_2, ..., p_{n,}\}}, m_k = \frac{p_k}{n s_k \max\{p_1, p_2, ..., p_{n,}\}}$$

$$k = 1, 2, ..., n; i = 1, 2, ..., s_k.$$

Where n is the number of videos retrieved with respect to a concept, and s_k the number of shots in kth video, and p_k the confidence of kth video containing the concept. If a video is not in the returned set of the text retrieval, the score of each shot in the video is simply set to zero. In the text retrieval fusion, the above score f is then combined with svm score g to

re-rank the shots. We use linear weighted fusion:

$$h = g + f / M;$$

where M is the maxim confidence of the videos returned with respect to a concept in the text retrieval. We use the above fusion to generate the run, Fuzhou_run1_1, using the direct svm prediction as a baseline. Our next run was formed by using the ontology relation fusion. In TRECVID 2010, ontology relations are available in a text file with two types of relations: A implies B and A excludes B. Relations that can be derived by transitivity are not included. In the ontology fusion, we determine relation coefficients a(i,j) in as follows:

```
if strRelation='implies'
        a(j,i)=1;
        a(j,i)=0.1;
elseif strRelation='excludes'
        a(j,i)=-1;
        a(i,j)=-1;
end
```

The shots then were reordered by using the following linear weighted fusion:

$$u_i = h_i + \sum_j a(i,j) * h_j \qquad i=1,2,\ldots,130$$

Our run Fuzhou_run2_2 was the result of the ontology fusion scheme using Fuzhou_run1_1 as its baseline. With ontology relation fusion, we observed a slight improvement over the Fuzhou_run1_1 in the number of the true shots returned (3687, up by 77). But the mean inferred average precision of the Fuzhou_run2_2 remained almost the same.

5. Result

Our group submitted 4 runs for the semantic indexing task. Three of them (Fuzhou_run1_1, Fuzhou_run2_2, Fuzhou_run3_3) are of “full” type, the other is of “light” type. The Fuzhou_Run3__3 which serves as a baseline is a direct svm prediction. Its Inferred Average Precision (InfAP) is only 0.001. It shows that the svm models are not chosen well and parameters are not well tuned. Fortunately, Fuzhou_run1_1, as a fusion outcome of the metadata text retrieval result with the svm prediction, performs much better with 0.012 InfAP. So does our next run. Fuzhou_run2_2 is the result of the ontology relation fusion. Both runs witness significant improvement over Fuzhou_Run3__3 and reach the best infAP with respect to the Explosion_Fire feature among the 30 features evaluated.

Figure 1 and Figure 2 show the evaluation results of the run

Fuzhou_run1_1.

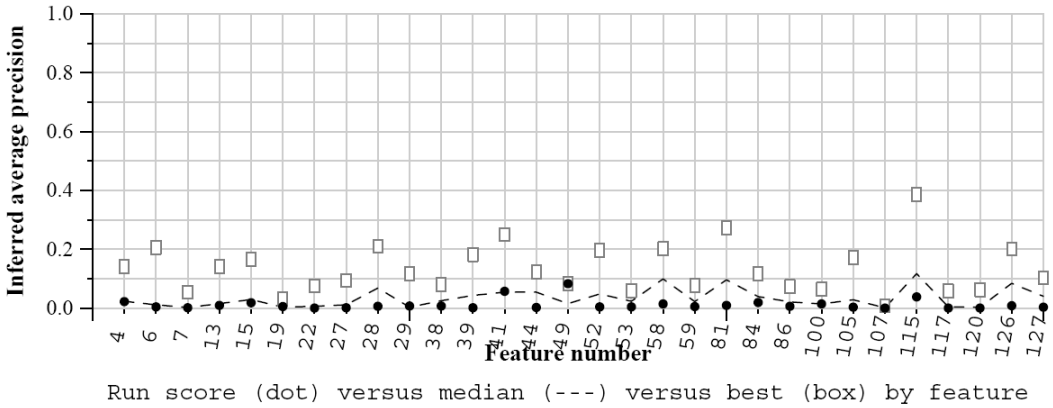

Figure 1: Fuzhou_run1_1 Run score

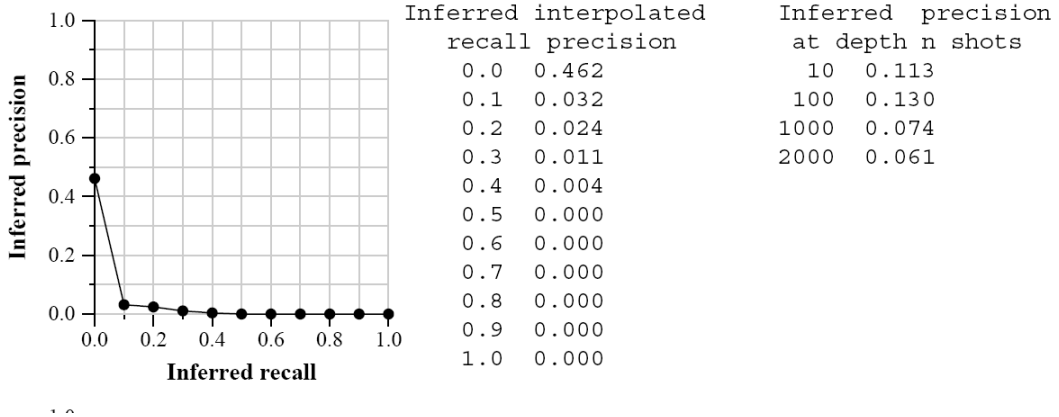

Figure 2: The Inferred AP of Fuzhou_run1_1

Both run1 and run2 have an Inferred Average Precision 0.083 for the Explosion_Fire (49th concept) which is significantly above the median infAP 0.015 for the concept. But evaluation results for most of other concepts are not so satisfactory. Some of them are well below the median. Due to restriction in time and resources we didn't train SVM models with more SVM parameters and low level features. This limitation affected the overall performance of our system.

6. Conclusion and Future Work

In this paper we present our participation in the semantic indexing task in TRECVID 2010. We have explored several fusion strategies including the low-level feature concatenation, metadata text retrieval fusion and ontology

relation fusion. Our experiments showed that the fusion of text retrieval results with the baseline enhanced our results. Our approach to ontology relation fusion increase the number of the inferred true shots returned. Two of our runs using fusion schemes have the best infAP for the Explosion Fire concept. Experiments also revealed that the selection of low-level features and fusion schemes is vital for achieving a good system performance.

It is apparent that there is a lot of work to be done to improve the detection system. For instance, more models should be trained and more low-level features, especially local descriptors such as SIFT and SURF, should be put in place. Adding some special detectors to the detection system is also important to obtain a better performance. Moreover, other fusion strategies need to be explored. The scarcity of positive examples and the data imbalance should be addressed. These measures, once taken, are definitely helpful to achieve a better system performance.

7. References

[1] Yuxin Peng, Zhiguo Yang, et al. PKU-ICST at TRECVID 2009: High Level Feature Extraction and Search, In Proceedings of TRECVID 2009 workshop.

[2] Bahjat Safadi and Georges Qu´enot, LIG at TRECVID 2009: Hierarchical Fusion for High Level Feature Extraction. In Proceedings of TRECVID 2009 workshop.

[3] Jinqiao Wang, Si Liu, Chao Liang, and Hanqing Lu ,IVA-NLPR-IA-CAS TRECVID 2009: High Level Features Extraction. In Proceedings of TRECVID 2009 workshop.

[4] Yingyu Liang, Binbin Cao, Jianmin Li, et al. THU-IMG at TRECVID 2009. In Proceedings of TRECVID 2009 workshop.

[5] Jason Hochreiter1, Silvino Barreiros1, et al. UCF @ TRECVID 2009:High-level Feature Extraction. In Proceedings of TRECVID 2009 workshop.

[6] Markus Mühling, Ralph Ewerth, et al. University of Marburg at TRECVID 2009: High-Level Feature Extraction. In Proceedings of TRECVID 2009 workshop

[7] A. Barla, F. Odone, and A. Verri. Histogram intersection kernel for image classification. In Image Processing, 2003. ICIP 2003. Proceedings.2003 International Conference on, volume 3, 2003.

[8] D. G. Lowe, “Distinctive image features from scale-invariant keypoints”, International Journal of Computer Vision(IJCV), vol. 60, no.2, pp.91-110, 2004.

[9] Herv´e Glotin, Zhongqiu Zhao, et al. IRIM at TRECVID 2008: High Level Feature Extraction. In Proceedings of TRECVID 2008 workshop.

[10] D. Wang, X. Liu, L. Luo, J. Li, B. Zhang. Video Diver: Generic Video Indexing with Diverse Features. MIR workshop at ACM Multimedia, 2007.

[11]Shih-Fu Chang , et al. Columbia University/VIREO-CityU/IRIT TRECVID2008 High-Level Feature Extraction and Interactive Video Search. In Proceedings of TRECVID 2008 workshop.

[12] C.C. Chang and C.J. Lin. LIBSVM: A Library for Support Vector Machines, 2001. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm