Eurecom-Polito at TRECVID 2016: Hyperlinking task

Benoit Huet
EURECOM
Sophia Antipolis, France
huet@eurecom.fr

Elena Baralis, Paolo Garza, and Mohammad Reza Kavoosifar
Department of Control and Computer Engineering
Politecnico di Torino
Torino, Italy
{name.surname}@polito.it

ABSTRACT

In this paper, we describe the system we used to address the Hyperlinking task at TRECVID 2016 and the achieved results. Our system combines several features (textual and visual) in order to consider the different facets of the input videos. Specifically, we combined automatically generated transcripts, visual concepts, and the text extracted by means of an OCR tool. We also exploited WordNet to find synonyms which are used to apply a simple query expansion technique. The four submitted runs aimed at analyzing the impact of the considered features on the quality of the retrieved hyperlinks.

1. INTRODUCTION

This paper describes the framework used by the Eurecom-Polito team to tackle the Hyperlinking task inside a video collection at TRECVID 2016 [2]. The main goal of the Hyperlinking task is to find video segments similar to a given (query) segment, called anchor, in a collection of videos. The returned hyperlinks enable the end users to find segments correlated to the anchor.

The data used in the TRECVID 2016 competition consists of 14,838 videos, for a total of 3,288 hours, provided by blip.tv.

We have proposed a system that exploits (i) automatic speech recognition transcripts [4, 6], (ii) visual concepts, (iii) the text extracted by means of an OCR tool, and (iv) a simple query expansion technique (based on WordNet [3] for identifying synonyms and related words).

The paper is organized as follows. Section 2 introduces the proposed system and the exploited video features. Section 3 describes the configurations of the four submitted runs and discusses how they have been selected, while Section 4 discusses the achieved results. Finally, Section 5 draws conclusions.

2. SYSTEM OVERVIEW

For the Hyperlinking task, we developed a system based on both textual and visual features. We exploited all the data and metadata provided by the task organizers, except visual concepts. Specifically, we decided to use the visual concepts extracted using the Caffe framework with the BVLC GoogLeNet model [7]. We also considered some other extra features. Specifically, we also developed a program, based on the Microsoft OneNote OCR tool, to detect the text inside the keyframes of the analyzed videos. Finally, WordNet [3] was used to find synonyms to perform query expansion.

The proposed system uses (i) automatic speech recognition transcripts [4, 6], (ii) visual concepts, based on the Caffe framework, (iii) the text extracted by means of the Microsoft OneNote OCR tool, and (iv) query expansion (based on WordNet for identifying synonyms and related words). We also considered the meta-data of the videos (specifically, title and tag have been considered).

The propose system is composed of four stages: Data segmentation (Section 2.1), Data enrichment (Section 2.2), Indexing (Section 2.3), Query formulation and Retrieval (Section 2.4).

2.1 Data segmentation

The first step that is applied on the video collection consists in splitting the videos in segments. We used two segmentation approaches during our experiments: (i) Fixed-segmentation, for which we considered 60 sec fixed segments, and (ii) Shot-based segmentation (provided by the organizers).

2.2 Data enrichment

Each video segment is enriched with a set of features. Figure 1 depicts the steps applied to enrich segments. Specifically, our data enrichment stage consists of a set of steps. The first step associates each segment with the related ASR transcript. We also extracted the visual concepts from the videos, by using the Caffe framework. Specifically, we applied the visual concept extraction tool every second. Then, each segment was enriched with the set of concepts appearing at least once with a score greater than 0.5 in that segment. Furthermore, we used an OCR tool to detect the text inside the keyframes of the segments. This text is also used to enrich each segment. Due to a time constraint, only a subset of the segments was processed by means of the OCR tool.

All the textual data associated with the segments have been preprocessed to remove irrelevant words. Specifically, we used a punctuation removal tool and we also removed stop words. We used 665 different English stop words for that. This way we narrowed down the word list of each segment to its core concepts. For a better match we decided to enrich the textual features of the segments with synonyms and conceptually connected items (hypernyms). For this

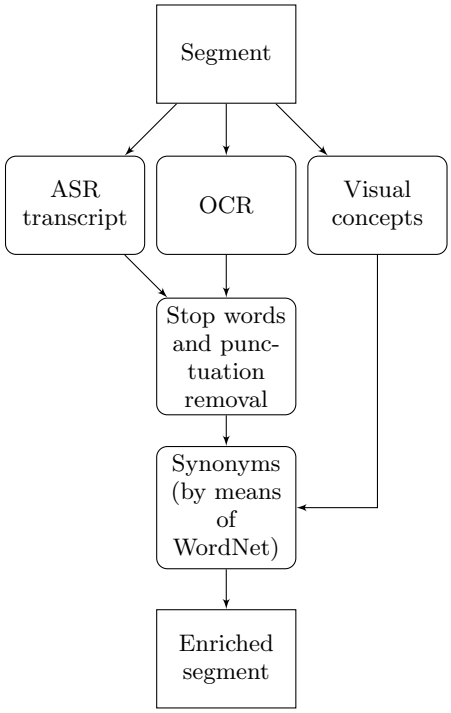

Figure 1: Enrichment phases

step, we used WordNet [3], which can give us synonyms and also other words and concepts connected to the words associated to each segment. Our main goal was to generate enriched concepts for each segment in order to improve the possibility of matching with the textual content of the search query. For example if the search query is "dog" and there is a segment which has the word "puppy" associated with it, the aim is to connect them and include the segment associated with "puppy" in the returned results.

2.3 Indexing

We used Apache Solr [1] to index the textual and visual features associated with each segment. We created multiple indexes for the enriched segments. Specifically, we created indexes on transcripts, visual concepts, synonyms and the text extracted by the OCR tool. The index created by Solr is known as an inverted index. An inverted index stores, for each term, the list of documents containing it. This makes term-based searches very efficient [5].

2.4 Query formulation and Retrieval

In this stage, we first transform the anchor (query) segment into a textual query by including in the text of the query all the textual information associated with the anchor (i.e., ASR transcript, visual concepts, OCR text) and also the meta-data of the video containing the anchor (i.e., title and tags of the video containing the anchor). A query expansion step is also applied by using WordNet. Specifically, WordNet is used to extend the content of the query with the synonyms of the words appearing in it.

Since some anchors are short, we decided to extend the segment boundary by including the context surrounding the anchor. We used a 30-second-long passage before and after each segment.

Table 1: Evaluation result

Measure	Run 1	Run 2	Run 3	Run 4
P@10	0.161	0.164	0.160	0.128
MAP	0.044	0.047	0.047	0.043
MAiSP	0.029	0.031	0.029	0.043

After the query preparation phase, a tool executes it by using Apache Solr and returns the related segments ranked by relevance.

3. SUBMITTED RUNS

For the Hyperlining sub-task, we submitted 4 runs by using different components/features for each run. Before selecting the configurations of the four runs, we performed a set of experiments on the test anchors to evaluate the impacts of the available transcript tools (LIUM vs LIMSI [4, 6]) and the video segmentation techniques (shot segmentation vs fixed length segmentation). When the test anchors are considered, the LIUM transcripts and the shot segmentation technique achieves the best results. For this reason, 3 of the submitted runs use the LIUM transcripts and 3 of them are based on the shot segmentation technique.

We selected the four configurations of the runs with the goal of analyzing the impact of some of the salient components of our system. Specifically, the four submitted runs are the followings:

Run 1. For the first run, we considered, to enrich the segments, the LIUM transcripts, visual concepts, and the output of the OCR tool. WordNet was also applied to include synonyms. The shot segmentation technique was considered to split the videos in segments. Hence, this run exploits all the components/features of our system.

Run 2. Similarly to Run 1, also this second run uses all the components of our system (i.e., transcripts, visual concept, OCR and synonyms based on WordNet). The only difference with respect to Run 1 is the use of the LIMSI transcripts. We decided to change the transcripts in order to evaluate the impact of different ASR tools. Also in this case the shot segmentation technique is considered.

Run 3. For the third run, we removed the component that uses WordNet to include synonyms in order to find how synonyms impact on the results. We used the LIUM transcripts, visual concepts and the output of the OCR tool. Again, the shot segmentation has been applied.

Run 4. For the fourth run, the segmentation is changed to fixed length segmentation. We considered 60 sec consecutive fixed length segments. We used the LIUM transcripts, visual concept and OCR for this run. Also in this case WordNet was used to include also synonyms in the search procedure.

4. RESULTS

The results of the four runs we submitted at the Hyperlinking task are summarized in Table 1.

The best results in term of Precision at 10 (P@10) and Mean Average Precision (MAP) are achieved by Run 2, which exploits all the available features (transcripts, visual concepts, and OCR). Analogously to Run 2, also Run 1 uses all the available features. The only difference between the two runs is the used ASR tool. Specifically, Run 2 is based on the LIMSI transcripts, whereas Run 1 is based on the LIUM ASR tool. Hence, with these data, the LIMSI transcripts seem to be slightly better than the LIUM ones.

The comparison between Run 1 and Run 3 allows analyzing the impact of synonyms. Based on the achieved results the use of synonyms has a negligible positive impact on P@10 (0.161 vs 0.160) and a negative impact on the MAP value (0.044 vs 0.047). Hence, it seems that our system is not able to effectively exploit synonyms.

Finally, Run 4 allows analyzing the impact of the segmentation technique. On the one hand fixed segmentation has a significant negative impact on P@10. On the other hand, the fixed segmentation approach achieved a higher MAiSP value. The segments retrieved by our system, when the shot segmentation technique is used, are on the average short. This is a possible reason of the decrease of MAiSP when the shot segmentation technique is considered. We should probably take into consideration the length of the returned segments in our system in order to avoid the selection of very short segments because they provide limited additional knowledge.

5. CONCLUSION

The proposed system has explored the use of textual and visual features for solving the Hyperlinking task. Specifically, we have considered two ASR tools, visual concepts and OCR. Moreover, the impact of synonyms has also been studied. The results achieved by our system, in the four considered configurations, are similar. Hence, the obtained results do not provide a significant insight about which of the considered components should be included or excluded on order to achieve the best performance in terms of P@10, MAP, and MAiSP.

6. REFERENCES

[1] Apache Solr. http://lucene.apache.org/solr.

[2] G. Awad, J. Fiscus, M. Michel, D. Joy, W. Kraaij, A. F. Smeaton, G. Quénot, M. Eskevich, R. Aly, and R. Ordelman. Trecvid 2016: Evaluating video search, video event detection, localization, and hyperlinking. In *Proceedings of TRECVID 2016*. NIST, USA, 2016.

[3] C. Fellbaum. *WordNet: An Electronic Lexical Database*. Bradford Books, 1998.

[4] J.-L. Gauvain. The quaero program: Multilingual and multimedia technologies. In *International Workshop on Spoken Language Translation (IWSLT)*, 2010.

[5] J. Kumar. *Apache Solr Search Patterns*. Packt Publishing Ltd, 2015.

[6] L. Lamel. Multilingual speech processing activities in quaero: Application to multimedia search in unstructured data. In *Baltic HLT*, pages 1–8, 2012.

[7] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–9, 2015.