IRISA at MediaEval 2012: Search and Hyperlinking Task

Camille Guinaudeau
IRISA/University of Rennes 1
Campus de Beaulieu
35042 Rennes Cedex, France
cguinaud@irisa.fr

Guillaume Gravier
IRISA/CNRS
Campus de Beaulieu
35042 Rennes Cedex, France
ggravier@irisa.fr

Pascale Sébillot
IRISA/INSA
Campus de Beaulieu
35042 Rennes Cedex, France
psebillo@irisa.fr

ABSTRACT

We describe our approach and results towards the Hyperlinking sub-task at MediaEval 2012. We approached this as an Information Retrieval task and used re-ranking strategies for finding relevant videos. A three-step approach was then applied on results to extract the most relevant part of the video regarding the query content. Our results show that re-ranking strategies and integration of metadata information both improve the system performance.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Content Analysis and Indexing—*Indexing methods*

Keywords

Information retrieval, automatic transcripts, re-ranking strategies, segment extraction, multimedia documents.

1. INTRODUCTION

The growing amount of multimedia material available on Internet is creating needs for new navigation schemes, allowing users to access relevant information inside a collection of video documents.

In this paper, we present the participation of IRISA at the Search and Hyperlinking Task at MediaEval 2012. For this participation, we chose to focus only on the second sub-task proposed by the organizers, namely the hyperlinking task, that consists of returning a ranked list of video segments which are relevant to a video segment query.

The main difficulty of this task is to figure out what kind of related content to return. Indeed, as no ground-truth was provided by the organizers—which reflects real life where user needs aren't known in advance— deciding what kind of information to rely on is challenging. If a user is interested in videos that show similar visual content to the query video, then visual information has to be favoured. But if he/she needs a video that presents spoken content similar to the query content, then the method developed should rely on textual information. The absence of ground-truth also gives us no possibility to evaluate our methods but manually, making it difficult to decide if a parametrization or a linking strategy gives better results than another.

The approach presented in this paper relies only on textual information in order to retrieve videos that share the same semantic content than the query video. This decision was made after an analysis of the development query set in which all videos had spoken content, sometimes associated with meaningless video[1].

In this paper, an overview of the hyperlinking system developed by IRISA is presented in Section 2, with the description of the two modules needed for hyperlinking achievement. The results are provided in Section 3, followed by future work in Section 4.

2. SYSTEM OVERVIEW

The hyperlinking system proposed consists of two separate modules. The first one is the *hyperlinking* module. This computes the similarity between a video segment query and the collection of videos and returns a ranked list of relevant videos. Different strategies and parametrizations, described in Section 2.2, were used for this first step. The second module is the *segment extraction* module, which takes the ranked list computed in the first step, and extracts from each video the segment that is the closest, from a meaning point of view, to the video segment query. As in the *hyperlinking* module, different strategies were defined to achieve this goal; see 2.3 for more details.

Both automatic transcripts provided by [2] and [4] and metadata associated with videos were used in our experiments (see [1] for more details on the experimental dataset) after preprocessing, as described in 2.1.

2.1 Data Preprocessing

The first step of preprocessing consists of dealing with data in languages other than English. One possibility we had was to return only videos in the language of the query, considering that if the user is seeing a video in a particular language then he/she is only interested in videos sharing that same language. However, as the development query set was only in English, this strategy amounted to discard all non-English videos. So, to take into account the whole data set, non-English transcripts were translated from their language to English with the Microsoft translator tool[2]. Second, transcripts and queries were lemmatized thanks to TreeTagger and only adjectives, nouns and non-modal verbs were kept to represent the data. Finally, the BM25 ranking function [3] was used to associate a score with each word in the documents. These scores, normalized between 0 and 1, are then used in the vectorial representation of the documents.

[1]For example, one of the queries was extracted from a radio show dealing with September 11th attacks and has visual content that consists only of natural landscapes.
[2]http://www.microsofttranslator.com/bv.aspx?

MediaEval 2012 Workshop, October 4-5, 2012, Pisa, Italy

2.2 Hyperlinking

Starting with the vectorial representation of documents, the *hyperlinking* module first computes the cosine distance similarity between the vector representing the query and the vectors of the collection's videos. This similarity calculation accounts for different kinds of information, either transcripts from LIMSI or LIUM, potentially combined with the metadata associated with the video query. In the different submitted runs, the similarity is computed between query and video transcripts provided by LIMSI (**RUN 1**); query and its metadata and video transcripts provided by LIMSI (**RUN 2**) or query and video transcripts provided by LIUM (**RUN 5**).

To this similarity computation, we also add two re-ranking strategies. The first one (**RUN 3**) consists of taking into account the metadata associated with the videos of the collection. In this case, the similarity between the query transcript and the metadata associated with the videos is computed and this score is used to re-rank the results of **RUN 1** by combining the two scores. The second re-ranking strategy (**RUN 4**) is based on the observation that a lot of videos in the collection are not stand-alone[3]. From this observation, it seems reasonable to believe that if a user is interested in an episode of one particular serie than he/she may be interested as well in the other episodes. To take this hypothesis into account, the results obtained by **RUN 1** are re-ranked so that videos that are part of the video query series, detected using the file name, appear on the top of the list. Finally, it has to be noted that the video from which the segment video query is extracted was removed from the results list.

2.3 Segment Extraction

The second module of our system extracts the most relevant segment of each video that appears in the ranked list of results. The segment extraction module is divided into three sequential steps. Given a relevant video, (1) the entire video or spoken part of the video is returned to the user if the video or the spoken content of the video is short (less than 2 minutes). If the video is longer, (2) the transcript associated with the video is segmented into topic segments if the video contains several topics[4], e.g. news programs. Then each topic segment is compared with the query (following the scheme presented in 2.2) and the most similar topic segment is returned to the user. Finally, if the video contains only one topic, (3) the transcript is scanned by a sliding window of 40 words (with an overlap of 10 words). For each position of the window, the number of content words of the query that appear in the window is computed. These values are used to produce a curve which is thresholded to extract potential segments (a potential segment is defined when the value is higher than the half of the maximal number of words per window position). The largest potential segment is then returned to the user.

In order to provide a segment that doesn't begin in the middle of an utterance, the boundaries of each segment were chosen so that they are located between two breath groups in the transcript.

[3]For example, ***Adobedreamer***-*SiteFileManagement365* and ***Adobedreamer***-*ScratchingTheSurfaceOfHTML248* are two different episodes of the same show *Adobedreamer*.

[4]To decide if a transcript contains several topics, a topic segmentation algorithm [5] parameterized to over-segment documents is applied. If the number of topic segments returned is small (less than 10) then the transcript probably contains only one topic.

Table 1: MAP values for the 5 submitted runs

RUN 1	RUN 2	RUN 3	RUN 4	RUN 5
0.247	0.333	**0.346**	0.310	0.208

3. EXPERIMENT RESULTS

As mentioned in the introduction, no ground-truth was provided for the hyperlinking sub-task, making it difficult to figure out how well the different strategies worked and what the influence of the different parameters was.

However we were able to verify the effectivness of our methods thanks to two simple tests. First, when not removed from the results' list, the video from which the query was extracted appears at the first place. Second, when applied on the video containing the query, the *segment extraction* module provided a video segment that corresponds roughly to the video segment query. Moreover, a qualitative analysis of the results obtained on the development set showed us that for each query the first 10 results seemed relevant.

Mean Average Precision values for the top 10 results are presented in Table 1. This table shows that, compared to **RUN 1**, the system's performance is improved when accounting for metadata, either during similarity computation (**RUN 2**) or re-ranking calculation (**RUN 3**). From Table 1, it can also be observed that both re-ranking strategies improve the quality of the results (**RUN 3** and **RUN 4**). Finally, results obtained for **RUN 5**—i.e. using LIUM transcripts—are the lowest ones, which can be explain by the quality of the automatic transcripts.

4. FUTURE WORK

By analyzing the results, we observed that some queries obtained really bad results for some runs (**RUN 1**, **RUN 2** and **RUN 5**) and really good results for others (**RUN 3** and **RUN 4**). We explain this difference by the fact that the transcript conveys in these cases little information and that taking into account different kinds of clues (series' name, metadata, etc.) increases results' quality. Therefore, in order to improve our system, we plan to integrate visual information such as face clustering or concept detection.

5. REFERENCES

[1] M. Eskevich, G.J.F Jones, S. Chen, R. Aly, R. Ordelman, and M. Larson. Search and Hyperlitralalanking Task at Mediaeval 2012. In *MediaEval Workshop 2012*, 2012.

[2] J.-L. Gauvain, L. Lamel, and G. Adda. The LIMSI broadcast news transcription system. *Speech Communication*, 37:89–108, 2002.

[3] S.E. Robertson and S. Walker. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *the 17th ACM SIGIR conference on Research and development in information retrieval*, 1994.

[4] A. Rousseau, F. Bougares, P. Deléglise, H. Schwenk, and Y. Estève. LIUM's systems for the IWSLT 2011 Speech Translation Tasks. In *the 8th International Workshop on Spoken Language Translation*, 2011.

[5] M. Utiyama and H. Isahara. A statistical model for domain-independent text segmentation. In *Annual meeting of the* ACL, 2001.