Ghent University-IBBT at MediaEval 2012 Search and Hyperlinking: Semantic Similarity using Named Entities *

Tom De Nies[1]
tom.denies@ugent.be

Pedro Debevere[1]
pedro.debevere@ugent.be

Davy Van Deursen[1]
davy.vandeursen@ugent.be

Wesley De Neve[1,2]
wesley.deneve@ugent.be

Erik Mannens[1]
erik.mannens@ugent.be

Rik Van de Walle[1]
rik.vandewalle@ugent.be

[1]Ghent University - IBBT - Multimedia Lab
[2]Korea Advanced Institute of Science and Technology (KAIST) - IVY Lab

ABSTRACT

In this paper, we attempt to tackle the MediaEval 2012 Search and Hyperlinking challenge, which focuses on video segment retrieval from a large dataset, based on short natural language queries, as well as linking the resulting segments to related ones. Our approach makes use of three semantic similarity metrics, merged by applying late fusion.

1. INTRODUCTION

In this paper, we describe our approach for tackling the MediaEval 2012 Search and Hyperlinking shared task [1]. This task focuses on information retrieval from the blip10000 dataset using audio transcripts. The videos are accompanied by two different automatic speech recognition (ASR) transcripts (generated by LIMSI [3] and LIUM [4]), textual metadata (tags) and automatically identified shot boundaries and keyframes [2]. There are two sub-tasks: the "Search Task" focuses on the search for known video segments using 30 natural language queries, whereas the "Linking Task" involves suggesting links to related video segments, either starting from the ground truth, or the acquired results of the Search Task. For each sub-task, a run is made using each of the ASR transcripts, as well as two additional runs combining the transcripts with user-generated tags.

We developed an approach to tackle both the Search and Linking task using one system, consisting of three steps:

1. create an enriched representation of the videos and the queries;
2. apply multiple similarity metrics to compare the input queries/segments to the dataset;
3. merge and sort the results by applying late fusion.

In the next sections, we discuss these steps in more detail.

2. STEP 1: ENRICHED REPRESENTATION

Before analyzing the data, we create an object container for each video and each query, as shown in Figure 1. Using the shot boundaries, the ASR transcripts are divided into shots based on their timing. This way, each shot has two transcripts, one for each ASR technology. Additionally, the user-generated tags are added as metadata to the video container. To add even more metadata, the concatenated transcripts from all shots of each video are fed to a Named Entity Recognition (NER) service (in our case, we used DBpedia Spotlight). The NER service will return a number of Named Entities (NEs), linked to DBpedia resources. The same approach is applied to the queries. Each natural language query is placed in a query object container. This object includes the text (title and short title) of the query, the detected Named Entities and tags. The tags are obtained by feeding the query text to a stop word removal and keyword extraction algorithm. In our implementation, all words after stop word removal are considered tags. In the Search sub-task, the query objects are compared to the entire video dataset. In the Linking sub-task, the anchor segments are converted to query objects by using their LIUM or LIMSI transcript as text. This way, we can use the same system for both sub-tasks.

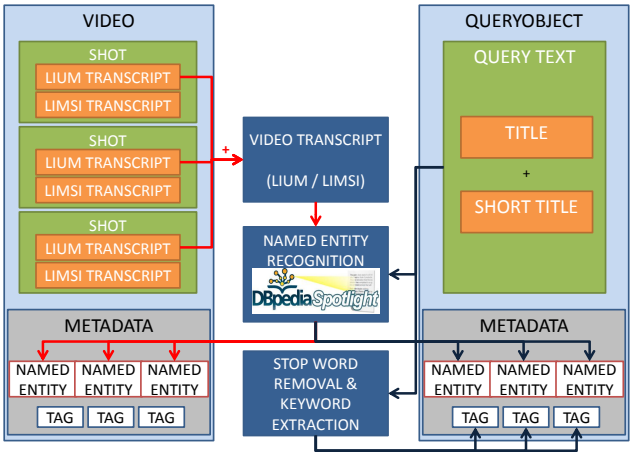

Figure 1: Structure of the data representation.

3. STEP 2: SIMILARITY METRICS

For our approach, we use three separate similarity metrics, and join their results using a late fusion technique. Each metric has its own advantages and disadvantages. This way, we aim to achieve good recall of the required segments.

*The research activities in this paper were funded by Ghent University, IBBT, the IWT Flanders, the FWO-Flanders, and the European Union, in the context of the IBBT project Smarter Media in Flanders (SMIF).

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

3.1 Bag of Words Similarity

This similarity metric uses the traditional Vector Space Model (VSM) or "Bag of Words" (BoW) representation of documents. Here, texts are represented as vectors of *Term Frequency-Inverse Document Frequency* (TF-IDF) weights, and their similarity is calculated as the cosine similarity between these two vectors. The full text of both documents is considered, as well as the entire dataset. The IDF ensures that very common terms in the corpus get a lower weight, thereby exploiting the more unique terms. However, this approach has a high computational complexity and does not disambiguate words (e.g. "Apple" could mean the company or the fruit in a different context).

3.2 Named Entity Similarity

This metric obtains a lower computational complexity than the BoW similarity by using a sparser representation of the text, and a faster way of assigning lower weights to common terms. For this purpose, we use the NEs extracted from the documents during the enrichment step. NEs are linked to URLs in an RDF graph and thus, are unambiguous. The $TF(e, D)$ of a NE e in document D remains the same: the number of occurrences of e in D. However, instead of the IDF, we introduce the *Inverse Support* (IS). If $support(e)$ is the number of incoming links of NE e, then the Inverse Support of e in document D is:

$$IS(a, A) = \frac{\sum_{e \in A} support(e)}{support(a)}. \quad (1)$$

The NE-based similarity is then calculated as in the cosine similarity of the vectors TF-IS weights. The weight of a NE e in document D is $TF(e, D) \cdot IS(e, D)$.

3.3 Tag Similarity

As a final similarity metric, we make use of the user-generated tags associated with the videos. Since these tags were added by humans, there is a high probability that humans will use the same keywords in their queries. The Tag-based similarity is calculated as the Jaccard similarity: the number of common tags divided by the total number of tags.

$$SIM_{\text{tag}}(A, B) = \frac{|\{t : t \in A \,\text{AND}\, t \in B\}|}{|\{t : t \in A \,\text{OR}\, t \in B\}|} \quad (2)$$

4. STEP 3: LATE FUSION

When executing a query, all three comparators are used in parallel, all using a different metric, and the similarity scores are merged afterwards. Figure 2 gives an overview of how this is done in our system. A suitable threshold T_{Ci} is chosen for each comparator C_i. All videos that are more similar to the query object than this threshold are passed to the fusion step. Note that a maximum number of results MAX is maintained, to avoid returning too many results. In our implementation, the parameters were set as follows: $T_{C1} = 0.1$, $T_{C2} = 0.6$, $T_{C3} = 0.4$, and $MAX = 60$.

If the same video is found by two or more comparators, its similarity scores are added up, ensuring a higher rank for this video. After all candidates are merged, they are sorted by descending similarity score. The first candidate in this sorted list is assigned rank 1, the second rank 2, and so on. Finally, the result segment is chosen from each candidate video, by selecting the shot with the highest BoW similarity to the query.

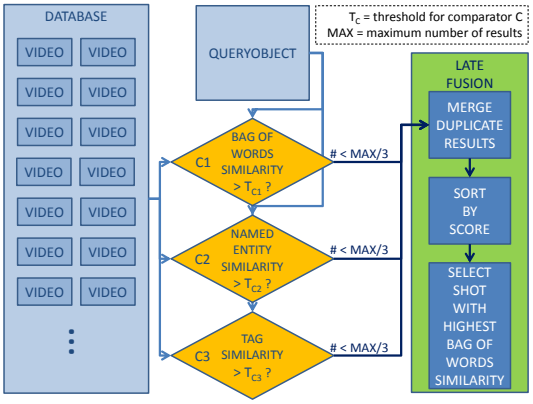

Figure 2: Overview of the late fusion approach.

5. EXPERIMENTS

We submitted twelve runs in total, four for the Search task with textual queries, four for the Linking task with the ground truth as anchor segments (LinkGT), and four for the Linking task with the search results as anchors (LinkSR).

1. Using LIMSI and only BoW and NE similarity.
2. Using LIUM and only BoW and NE similarity.
3. Using LIMSI and all 3 similarity metrics.
4. Using LIUM and all 3 similarity metrics.

As explained in [1],the Search task was evaluated using the Mean Reciprocal Rank (MRR) with a window size of 60 seconds, and the Linking task using the Mean Average Precision (MAP). The results are summarized in Table 1.

run	Search: MRR	LinkGT: MAP	LinkSR: MAP
1	0.188	0.157	0.014
2	0.254	0.171	0.040
3	0.165	0.157	0.003
4	0.221	0.171	0.037

Table 1: Evaluation results of the submitted runs

6. DISCUSSION AND FUTURE WORK

We observe that the MRR of the late fusion approach when using three comparators is worse than when using only two. The explanation for this is that while the tag-comparator find more results, this pushes the other results to a lower rank, thus decreasing the MRR of correct results. In future work, we plan to sort the final results using an optimally weighted sum of all comparators to counter this.

7. REFERENCES

[1] M. Eskevich, G. J. F.. Jones, S. Chen, R. Aly, R. Ordelman, and M. Larson. Search and Hyperlinking Task at MediaEval 2012. In *MediaEval 2012 Workshop*, Pisa, Italy, October 4-5 2012.

[2] P. Kelm, S. Schmiedeke, and T. Sikora. Feature-based video key frame extraction for low quality video sequences. In *Proceedings of the WIAMIS '09 Workshop*, pages 25–28, May 6-8 2009.

[3] L. Lamel and J.-L. Gauvain. Speech Processing for Audio Indexing. *Advances in Natural Language Processing. (LNCS 5221)*, pages 4–15, 2008.

[4] A. Rousseau, F. Bougares, P. Deléglise, H. Schwenk, and Y. Estève. LIUM's systems for the IWSLT 2011 Speech Translation Tasks. In *Proceedings of the IWSLT Workshop, San Francisco, CA*, 2011.