DCU at VideoClef 2008

Eamonn Newman and Gareth J.F. Jones

Centre for Digital Video Processing, Dublin City University, Dublin 9, Ireland
{enewman|gjones}@computing.dcu.ie
http://www.cdvp.dcu.ie

Abstract. We describe a baseline system for the VideoCLEF Vid2RSS task in which videos are to be classified into thematic categories based on their content. The system uses an off-the-shelf Information Retrieval system. Speech transcripts generated using automated speech recognition are indexed using default stemming and stopping methods. The categories are populated by using the category theme (or label) as a query on the collection, and assigning the retrieved items to that particular category. Run 4 of our system achieved the highest f-score in the task by maximising recall. We discuss this in terms of the primary aims of the task, i.e., automating video classification.

Key words: Classification, Information Retrieval, Automatic Speech Recognition

1 Introduction

The VideoClef Vid2RSS task required users to classify videos into one (or more) of a set of categories. Audio content consists primarily of Dutch with some embedded English content. The data provided consists of automatic speech recognition (ASR) transcripts (generated independently using Dutch and English ASR systems), shot boundary keyframes, and catalogue metadata (in Dutch). Each category is then published as an RSS feed. The system described in this paper is based on an Information Retrieval approach. We built a standard free text index using the ASR transcripts and associated metadata as the content.

2 System Description

We used the open source Lucene Search Engine technology [1] as the base technology for our system. Dutch-language content was stopped, stemmed and tokenised using Lucene's built-in Dutch analyser, `DutchAnalyzer`[1]. English-language content was stopped and tokenised by the Lucene default tokeniser, `StandardAnalyzer`[2]. The `StandardAnalyzer` does not perform any stemming of tokens.

[1] org.apache.lucene.analysis.nl.DutchAnalyzer
[2] org.apache.lucene.analysis.standard.StandardAnalyzer

2.1 Run Configurations

Five separate runs were prepared and submitted to the task. The runs varied in terms of both system configuration and the data which was used.

1. **Dutch ASR transcripts:** In this run, we indexed the entire set of Dutch ASR transcripts (the `FreeTextAnnotation` elements). The index was queried with the labels in the order given in Table 1 and each item was classified into a single category.
2. **English ASR transcripts:** This is identical to Run 1, using English ASR transcripts and translations of the category labels as queries.
3. **Dutch ASR with query expansion:** We ran the same queries as Run 1, but added an additional step of query expansion in order to improve the recall of certain categories (some categories in earlier runs had nothing assigned to them). Because of this, items could be assigned to multiple categories. Queries were expanded by performing an initial query which consisted of just the category label. We take the first 10 retrieved documents and extract the 5 most frequently occuring terms in each. We process this set of 50 terms to remove any duplicates. The remaining terms are combined with the original query to form the expanded query.
4. **English ASR with query expansion:** This is identical in method to Run 3, but the data now consists of the English ASR transcripts, rather than the Dutch.
5. **Dutch metadata:** We indexed the catalogue metadata which were supplied in the data sets. Specifically, we used the `description` elements from the metadata documents. Once again, the Dutch category term labels were used as queries, and the items were restricted to appear in one feed only.

2.2 Category order

The categories were ordered from most specific to least specific, as in Table 1. For each category, a query was made to the IR system using the category name as the query keyword. All retrieved items were labelled as belonging to that category. The ordering meant that when an item was retrieved, it was placed into the most specific category possible. For our submitted runs, in Runs 1, 3, and 5, a retrieved item was placed only into the first category for which it was retrieved. In Runs 2 and 4, it was placed in all categories for which it was retrieved. This restriction was imposed to improve the precision of the classification task, since labels such as "film" were very general and tended to capture most, if not all, of the items.

3 Results

In Table 2 we present the retrieval scores attained by our system runs. The metrics are defined in Section 2.2 of the Track Overview paper [2]. A direct comparison of Runs 1 and 2 suggests that the Dutch transcripts were more

Dutch	English
archeologie	archeology
architectuur	architecture
chemie	chemistry
dansen	dance
schilderijen	paintings
wetenschappelijk onderzoek	scientific research
beeldende kunst	visual arts
geschiedenis	history
film	film
muziek	music

Table 1. Category Label Order

metric	Run 1	Run 2	Run 3	Run 4	Run 5
micro-average precision	0.50	0.32	0.16	0.17	0.83
micro-average recall	0.35	0.21	0.91	0.72	0.18
f-score micro-average	0.41	0.25	0.28	0.28	0.29
macro-average precision	0.54	0.62	0.42	0.50	0.93
macro-average recall	0.55	0.38	0.90	0.70	0.28
f-score macro-average	0.54	0.47	0.58	0.59	0.43

Table 2. Vid2RSS Scores for Runs 1 to 5

useful in identifying the subject categories than the English ones. Indeed, the English transcripts had the poorest f-scores at both micro and macro level. This is most likely attributable to the fact that the majority of the dialogue was in Dutch and so contained less "noise" than the English counterparts. Processing of the ASR transcripts to identify the points at which the language changed would allow for the combination of transcripts (or the removal of erroneous segments) which would improve classification performance.

Runs 3 and 4 used query expansion to add new keywords to the queries and allowed items to be placed in multiple categories. As we can see, this relaxation resulted in a large drop in the micro-average precision scores of these systems; conversely, the micro-average recall is much higher in these runs. As items were placed in multiple categories the chances of an item being correctly classified were much greater, however the number of false positives also increased.

As can be seen from the results, Run 5 performed particularly well in terms of precision, and relatively well (when compared to our other runs) in terms of recall. However, since this was on the metadata and not on the ASR transcripts, it cannot be directly compared to the others. The higher precision scores do suggest that there may be merit in combining the different data sets available.

One drawback that is immediately obvious with this system is that it is not possible to guarantee that all items will be classified. If an item is not retrieved for any of the queries, then it will not be placed in any of the category feeds. As it happens, this was not the case for any of the runs with this particular data

set (probably due to the presence of highly generic labels such as "film" and "music")

Additionally, the number of terms added in the query expansion phase could be reduced. The maximum for this was 50, but elimination of duplicates meant that the size of the set was generally much smaller. Nevertheless, it seems that too many terms were added to the queries, and this is supported by the difference in micro-average precision between Runs 1 and 3 and Runs 2 and 4. To overcome this, we plan to implement a standard query expansion method where this can be controlled.

4 Conclusions

On comparing the results of our runs with those of other participants (see Track Overview [2] for full comparative analysis), Run 4 was shown to have the highest f-score of all systems when averaged over the individual f-scores for each of the topic classes (macro-average). This was attained by deliberately promoting recall over precision, in allowing videos to be classified under multiple topics. Furthermore, as mentioned in the Overview paper, the imbalance between precision and recall may not result in a particularly useful system. From this point of view, Run 1, which has the closest balance between precision and recall, may be seen as most useful to a human evaluator.

The results suggest that there is room for improvement in our system. The precision scores could be improved by finer-grained query expansion, which will be examined in future experiments. Additionally, the performance on the English-language content could be improved by use of a stemming algorithm, such as Porter [3].

Acknowledgements

This work was supported by the EU IST programme of the 6th FP for RTD under project MultiMATCH contract IST-033104.

References

1. Apache Software Foundation. Lucene: Java-based Indexing and Searching technology, http://lucene.apache.org/.
2. M. Larson, E. Newman, and G. J. F. Jones. Overview of VideoCLEF2008: Automatic Generation of Topic-based Feeds for Dual Language Audio-Visual Content. In C. Peters, D. Giampiccolo, N. Ferro, V. Petras, J. Gonzalo, A. Peñas, T. Deselaers, T. Mandl, G. J. F. Jones, and M Kurimo, editors, *Evaluating Systems for Multilingual and Multimodal Information Access – 9th Workshop of the Cross-Language Evaluation Forum*, Lecture Notes in Computer Science, Aarhus, Denmark, September 2008 (printed in 2009).
3. M. Porter. An algorithm for suffix stripping. *Program*, July 1980.