Speech retrieval experiments using XML Information Retrieval

Djoerd Hiemstra, Roeland Ordelman, Robin Aly, Laurens van der Werff, Franciska de Jong
University of Twente, the Netherlands
d.hiemstra, r.j.f.ordelman, r.aly, l.b.vanderwerff, f.m.g.dejong@utwente.nl

Abstract

This report presents the University of Twente's first cross-language speech retrieval experiments in Cross-Language Evaluation Forum (CLEF). It describes the issues our contribution was focusing on, it describes the PF/Tijah XML Information Retrieval system that was used and it discusses the results for both the monolingual English and the Dutch-English cross-language spoken document retrieval (CL-SR) task. The paper concludes with an overview of future research plans.

1 Introduction

Traditionally the retrieval of elements from a multimedia document collection has been approached by creating indexes relying on manual annotation with controlled vocabulary index terms. With the emergence of digital archiving this approach is still widely in use and for many archiving institutes the creation of manually generated metadata is and will be an important part of the daily work. When the automation of metadata generation is considered, it is often seen as something that can enhance the existing process rather than replace it. The available metadata will therefore often be a combination of highly reliable and conceptually rich manual annotations, and (semi)automatically generated metadata. A big challenge for Information Retrieval is to combine the various types of metadata and to let the user benefit from this combination.

Audio content is an example of information for which automatically generated metadata could very well be combined with other information sources in order to enhance a user's search. Audio metadata may be regarded as a set of special document features; each feature adds to the overall representation of a document. Traditionally, these features include for example the localisation of the speech/non-speech fragments, the identification of the speaker and full text transcripts of the speech within the document. In some cases, speech recognition transcripts even provide the only means for searching a collection, for example when manual annotation of a collection is not feasible. More recently tools have become available for the extraction of additional speaker features from the audio signal such as emotions (e.g., affect bursts such as sobs, crying or words that express emotions), language, accent, dialect, age, gender and sociolect.

A document can be very long and probably not every part is relevant given a specific user request. Browsing manually through a long (multimedia) document to find the most relevant parts can be cumbersome. Therefore, (multimedia) retrieval systems have to be capable of retrieving only relevant *parts* of documents. As a consequence partitioning or segmentation of documents should take place prior to the actual extraction of features.

Usually, the segmentation 'algorithm' for defining sensible document parts comes almost naturally with a document collection. In the Multilingual Access to Large Spoken Archives (MALACH) collection, which is the target collection for CLEF's CL-SR task, parts of interviews on a single coherent subject have been selected as a segments. In a broadcast news retrieval system, single news items would be the obvious unit to choose for segmentation purposes.

However, given that the currently available techniques can generate multiple annotations, multiple segmentations can be supported. This complicates the MALACH task. For the MALACH collection an Information Retrieval system should be able to handle the following type of query: "give me document parts of male, native Dutch speakers talking about [...] without expressing any emotions". With this scenario the following questions need to be addressed: (i) how to efficiently store these multiple annotations, (ii) how to combine and rank retrieval scores on them and (iii) how to select appropriate segment boundaries that determine what is going to be returned as part of a result.

MPEG-7 [7] defines standardised ways of storing annotations of multimedia content. It has the potential of solving some of the issues related to the use of multiple annotations in multimedia retrieval. For this reason the underlying XML [8] format was chosen as a starting point for our research. An XML database is employed to store collections. For the Information Retrieval requests PF/Tijah was used, an XQuery extension made for this purpose.

CLEF 2006 CL-SR is an ideal evaluation scenario for our research because: (i) The provided test collection existed in XML-like format, (ii) the collection was pre-segmented, (iii) each segment contained manual and automatic annotations, and (iv) these annotations also had to be combined. CLEF 2006 CL-SR offers therefore a welcome framework for evaluating the preliminary implementation of our ideas described below. A second reason is that it links up other work spoken document retrieval for oral hostory collections done at the University of Twente, more in particular to the CHoral project, which is focusing on speech retrieval techniques for Dutch heritage collections. [9].

The rest of the paper is organised as follows. In section 2 the PF/Tijah module, its application to the retrieval task and the cross-lingual aspects are presented. Section 3 presents our results on the given retrieval topics. We end the paper by giving a conclusion and future work in section 4.

2 PF/Tijah: text search in XQuery

PF/Tijah is a research project run by the University of Twente with the goal to create an extendable search system. It is built on top of the PathFinder (PF) XQuery execution system [1] and uses the Tijah XML Information Retrieval system [4]. PF/Tijah is part of the XML Database Management System MonetDB/XQuery. The whole system is an open source package developed in cooperation with CWI Amsterdam and the University of Munich. It is available from SourceForge.[1] The rest of this section is structured as follows: In subsection 2.1 we describe the PF/Tijah system. The following subsection 2.2 gives insights on the structure of the annotations of the MALACH collection and the application of the PF/Tijah system to perform the queries on the given topics. The last subsection 2.3 gives details on the cross-language aspects.

2.1 Features of PF/Tijah

PF/Tijah supports retrieving arbitrary parts of XML data, unlike traditional information retrieval systems for which the notion of a *document* needs to be defined up front by the application developer. For instance, if the data consists of MPEG-7 annotations, one can retrieve complete documents, but also only parts inside documents with no need to adapt the index or any other part of the system.

Information retrieval queries are specified in PF/Tijah through a user-defined function in XQuery that takes a NEXI query as its argument. NEXI [5] stands for *Narrowed Extended XPath*. It is narrowed since it only supports the descendant and the self axis steps. "Extended" because of its special `about()` function that takes a sequence of nodes and ranks those by their estimated probability of relevance to the query consisting of a number of terms.

PF/Tijah supports extended result presentation by means of the underlying query language. For instance, when searching for document parts, it is easy to include any XML element in the collection to the result. This can be done in a declarative way (i.e., not focusing on how-to

[1] http://sourceforge.net/projects/monetdb/

include something but what to include). This could be additional information on the author of the document or technical information such as the encoding. The result is gathered using XPath expressions and is returned by means of XQuery element constructions.

As a result of the integration of Information Retrieval functionality in XQuery, PF/Tijah Information Retrieval search/ranking can be combined with traditional database querytechniques, including for instance joins of datasets on values.

For example, one could search for video parts in which a word is mentioned, that is listed in a name attribute in an XML document collection on actors. It is also possible to narrow down the results by constraining the actor element further by, say, the earliest movie he played in.

2.2 Querying the MALACH collection

The Shoah Foundation manually segmented the MALACH interviews into coherent parts on the same topic. The available part of the MALACH collection comprises 8104 interview parts. These parts were defined a the unit of retrieval in the evaluation task. This task consisted of 105 topics. Out of these, 63 topics came with relevance judgements which could be used for training. They contain manual annotations like anonymized references to the interviewed person, keywords and a 3-sentence summary. The automatically generated annotations comprised speech recognition transcripts (ASR) and automatic keyword annotation from the interview parts. The ASR was provided by IBM [6] in four different variations: describing different versions of the recognition system used and different content (for example a mixture of Version 2004 and 2006).

```
<CLSRDATA>
 <DOC>
  <DOCNO>Identifier of interview and current segment</DOCNO>
  <INTERVIEWDATA>Metadata about interviewee</INTERVIEWDATA>
  <NAME>Full name of every person mentioned (anonymized)</NAME>
  <MANUALKEYWORD>Keywords assigned to the segment</MANUALKEYWORD>
  <SUMMARY>3-sentence segment summary</SUMMARY>
  <ASRTEXT2003A>ASR transcript from 2003</ASRTEXT2003A>
  <ASRTEXT2004A>ASR transcript from 2004</ASRTEXT2004A>
  <ASRTEXT2006A>ASR transcript from 2006</ASRTEXT2006A>
  <ASRTEXT2006B>ASR transcript from 2006 or 2004</ASRTEXT2006B>
  <AUTOKEYWORD2004A1>Keywords from a kNN classifier</AUTOKEYWORD2004A1>
  <AUTOKEYWORD2004A2>Keywords from a kNN classifier</AUTOKEYWORD2004A2>
 </DOC>
 <DOC>
   :
   :
</CLSRDATA>
```

Figure 1: Format of the standard video segments data

In order to evaluate the topics, the SGML input was transformed into valid XML. The format of the data is shown in Figure 1. A part of an interview is enclosed in a `DOC` element. The content of the `DOCNO` child-element consisted of three parts: (i) the IntCode uniquely identifying the interview within the collection, (ii) the SegId uniquely identifying the interview part in the collection and (iii) the SeqNum specifying the sequence number of this interview part in the whole interview. This allowes traditional Information Retrieval systems to combine parts of a an interview to one whole document (via the IntCode). In PF/Tijah however, a more hierarchical organisation of the data would be beneficial. For example, the introduction of an element, say `INTERVIEW`, that encloses all `DOC` elements of one interview would remove redundant data. The `INTERVIEW` should contain a element for the IntCode. The `DOC` element cound then have two seperate elements `SegId` and `SeqNum`. This way PF/Tijah will be able to generate the same query results but the hierarchical structure of XML would be better exploited.

PF/Tijah can be used to query the MALACH collection as follows. Considering topic 1133, *"The*

story of Varian Fry and the Emergency Rescue Committee who saved thousands in Marseille", a simple CLEF CL-SR query that searches the ASR transcript of 2004 is presented in Figure 2. To understand the query it is necessary to have some basic knowledge of XQuery as for instance described by Katz et al. [3]. In the query, we specify which elements to search (in this case `ASRTEXT2004A`) explicitly. It is easy to run queries on other elements (annotations), for instance the `SUMMARY` elements, without the need to re-index. Furthermore, queries on several elements can be done by combining multiple `about()` functions with `and` or `or` operators in the NEXI query.

```
let $c := doc("Segments.xml")
for $doc in tijah-query($c, "//DOC[about(.//ASRTEXT2004A,
   varian fry)];")
return $doc/DOCNO
```

Figure 2: Simple PF/Tijah query

```
let $opt := <TijahOptions returnNumber="1000" algebraType="COARSE2"
  txtmodel_model="NLLR"/>
let $c := doc("Segments.xml")
(:  for all topics :)
for $q in doc("EnglishQueries.xml")//top
  (:  generate NEXI Query :)
  let $q_text := tijah-tokenize(exactly-one($q/title/text()))
  let $q_nexi := concat("//DOC[about(.//ASRTEXT2004A,", $q_text, ")];")
  (:  start query on the document node :)
  let $result := tijah-query-id($opt, $c, $q_nexi)
  (:  for each result node print the standardized result :)
  for $doc at $rank in tijah-nodes($result)
  return
    string-join(($q/num/text(), "Q0", $doc/DOCNO/text(), string($rank),
      string(tijah-score($result, $doc)), "pftijah"), " ")
```

Figure 3: PF/Tijah query that completely specifies a CLEF run

Interestingly, we can specify one complete CLEF CL-SR run in one huge XQuery statement as shown in Figure 3. The query takes every topic from the CLEF topics file, creates a NEXI query from the topic title, runs each query on the collection, and produces the official CLEF output format. The `<TijahOptions />` element contains options for processing the query, such as the retrieval model that is supposed to be used. More about PF/Tijah can be found in [2].

2.3 Cross-Lingual IR

Cross-Lingual Information Retrieval has been limited to Dutch queries. The topics provided by the workshop organisers were in English. In order to simulate Dutch cross-language experiments, the topics first had to be translated (manually) into Dutch. These generated Dutch topics were applied as queries on an English collection. For this purpose all Dutch queries were fully automatically translated into English and further processed with the procedure described above.

As machine translation (MT) system we used the free web translation service *freetranslation.com* [2], which is based on the SDL Enterprise Translation Server, a hybrid phrase based statistical and example based machine translation system. There was no pre- or post-processing carried out. The results of *freetranslation.com* seemed to be performing good, though no formal

[2] `http://www.freetranslation.com/`

evaluation was carried out. Since online MT systems like this are being adapted regularly, it will be difficult to repeat the achieved results. This is one of the reasons we consider to build our own statistical MT system for a future CLEF evaluation.

3 Experimental results

We tested the out-of-the-box performance of PF/Tijah on the standard interview's metadata. Table 1 shows the experimental results of 9 official experiments. One experiment is the obligatory ASR run using long queries using topic title and topic description (denoted as 'TD' in the table) as a query. The other eight runs were done using short queries with only the topic title ('T' in the table). Five out of nine runs use the original English topics ('EN' in the table). The four remaining runs use the manually translated Dutch topics ('NL' in the table) to search the English annotations. Runs were done on different annotations: on the ASR transcripts version 2004 ('ASR04'), on the manual keywords ('MK') and/or on the manual summaries ('SUM'), see also Figure 1.

Run name	Lang.	Query	Annotations	Train (map)	Test (map)
UTasr04aEN-TD	EN	TD	ASR04	0.063	0.038
UTasr04aEN	EN	T	ASR04	0.058	0.050
UTasr04aNl2	NL	T	ASR04	0.052	0.038
UTsummkENor	EN	T	MK, SUM	0.250	0.206
UTsummkNl2or	NL	T	MK, SUM	0.211	0.165
UTmkEN	EN	T	MK	0.203	0.155
UTmkNL2	NL	T	MK	0.164	0.120
UTasr04mkEN	EN	T	ASR04, MK	0.218	0.165
UTasr04mkNL2	NL	T	ASR04, MK	0.176	0.132

Table 1: Mean average precision (map) on train and test queries

Table 1 reports the official CLEF 2006 mean average precision results of 64 train topics and 33 test topics. The ASR-only results are significantly worse than any of the runs that use manual annotations (manual keywords or interview part summaries). Best results are obtained by combining the manual keywords and summaries. Interestingly, on the test data, the difference between manual keywords only (UTmkEN) and manual keywords plus ASR (UTasr04mkEN) is statistically significant at the 5% level according to a paired sign test.

4 Conclusions and Future Work

Expectedly, retrieval results using ASR transcripts are far off from the results of manual annotation. For example, manually annotated summaries combined with manual keywords have 0.25 mean average precision compared to 0.06 for the ASR transcripts. Taking costs and the available resources for the annotation process into account, manual annotation often turns out to be infeasible. Therefore, if no manual annotations are available – which is true for quite a number of audiovisual collections –, automatic annotation at least allows a specific document to be considered for relevance judgment. In addition, experiments show that the combined use of ASR and MK annotations improve the precision of the results significantly, compared to sole use of MK. From this the expectation is derived, that the combination of manual and automatically generated annotations can be beneficial in general.

As it was stated in the introduction, using multiple annotations gives rise to several important questions. The experiments described in this paper have been a first exploration of these issues using a well-defined spoken document retrieval evaluation corpus. We aim to continue this line of research by investigating (i) how annotation layers can most efficiently be stored, (ii) how retrieval scores could best be combined, and (iii) how document segmentation should be dealt with.

The used storage structure for the annotations was the predefined format (transformed to valid XML) of the MALACH collection. All annotations were aligned in one segmentation scheme: taking parts of interviews only on one coherent topic. This way all annotations nicely fitted into the hierarchical structure of XML. In other realistic settings segmentations could overlap. For example, an automatically detected emotion could span multiple parts of an interview. This leaves the question open of how to efficiently store multiple annotations. Here, existing standards for data structures for the storage of annotations –like MPEG-7– have to be considered and evaluated.

The retrieval of information based on manual and automatic annotations was done so far in "or" manner. This assumes that both annotation types contribute equally to the relevance of a segment. It is however more realistic to assume some kind of ordering or weighting. For example, manual keyword annotations might receive a larger weight than an ASR transcript. How to solve the problem of possible overlapping annotations during the combination of retrieval results on different annotations is still to be settled.

With possibly huge amounts of content in one single multimedia document a retrieval system has to be able to select only certain parts which are presented to the user as being relevant. Because annotations may segment a document in multiple ways, the issue of which part to return as relevant needs to be addressed in more detail.

Acknowledgements

We are grateful to the research programmes that made this work possible: Djoerd Hiemstra and Roeland Ordelman were funded by the Dutch BSIK project MultimediaN [3]. Robin Aly was funded by the Centre of Telematics and Information Technology's SRO-NICE programme [4]. Laurens van der Werff was funded by the Dutch NWO CATCH project Choral [5]. Many thanks to Arthur van Bunningen, Sander Evers, Robert de Groote, Ronald Poppe, Ingo Wassink, Dennis Reidsma, Dolf Trieschnigg, Lynn Packwood, Wim Fikkert, Jan Kuper, Dennis Hofs, Ivo Swartjes, Rieks op den Akker, Harold van Heerde, Boris van Schooten, Mariet Theune, Frits Ordelman, Charlotte Bijron, Sander Timmerman and Paul de Groot for translating the English topics to Dutch.

References

[1] P. Boncz, T. Grust, M. van Keulen, S. Manegold, J. Rittinger, and J. Teubner. MonetDB/XQuery: A fast XQuery processor powered by a relational engine. In *Proceedings of the ACM SIGMOD Conference on Management of Data*, 2006.

[2] D. Hiemstra, H. Rode, R. van Os, and J. Flokstra. PFTijah: text search in an XML database system. In *Proceedings of the 2nd International Workshop on Open Source Information Retrieval (OSIR)*, 2006.

[3] H. Katz, D. Chamberlin, D. Draper, M. Fernandez, M. Kay, J. Robie, M. Rys, J. Simeon, J. Tivy, and P. Wadler. *XQuery from the Experts: A Guide to the W3c XML Query Language*. Addison Wesley, 2003.

[4] J. List, V. Mihajlovic, G. Ramirez, A.P. de Vries, D. Hiemstra, and H.E. Blok. Tijah: Embracing IR methods in XML databases. *Information Retrieval Journal*, 8(4):547–570, 2005.

[5] R.A. O'Keefe and A. Trotman. The simplest query language that could possibly work. In *Proceedings of the 2nd Initiative for the Evaluation of XML Retrieval (INEX)*. ERCIM workshop proceedings, 2004.

[3] `http://www.multimedian.nl/`
[4] `http://www.ctit.utwente.nl/research/sro/nice/`
[5] `http://hmi.ewi.utwente.nl/project/CHoral`

[6] R.W. White, D.W. Oard, G.J.F. Jones, D. Soergel, and X. Huang. Overview of the CLEF 2005 cross-language speech retrieval track. In *Working Notes for the CLEF 2005 Workshop*, 2005.

[7] MPEG Systems Group ISO/MPEG N4285, Text of ISO/IEC Final Draft International Standard 15938-1 Information Technology - Multimedia Content Description Interface - Part 1 Systems Sydney, July 2001.

[8] Tim Bray, Jean Paoli, C.M. Sperberg-McQueen and Eve Maler The Extensible Markup Language (XML) 1.0 (Third Edition) W3C Recommendation February 2004.

[9] Ordelman, R.J.F. and Jong de, F.M.G. and Hessen van, A.J. THE ROLE OF AUTOMATED SPEECH AND AUDIO ANALYSIS IN SEMANTIC MULTIMEDIA ANNOTATION in Proceedings of the International Conference on Visual Information Engineering (VIE2006) September 2006 Bangalore, India, Note: accepted, not yet published