

Cross Document Annotation for Multimedia Retrieval

Dennis Reidsma and Jan Kuper

University of Twente, Dept. of Computer Science, Parlevink Group
P.O. Box 217, 7500 AE Enschede, the Netherlands
{dennisr, jankuper}@cs.utwente.nl

Thierry Declerck

DFKI GmbH, Language Technology Lab
Stuhlsatzenhausweg 3, D-66123 Saarbrücken, Germany
declerck@dfki.de

Horacio Saggion and Hamish Cunningham

University of Sheffield, Dept. of Computer Science, NLP Group
Regent Court, 211 Portobello Street, Sheffield, S1 4DP, UNITED KINGDOM
{h.saggion, hamish}@dcs.shef.ac.uk

Abstract

This paper describes the MUMIS project, which applies ontology based Information Extraction to improve the results of Information Retrieval in multimedia archives. The domain specific ontology, the multilingual lexicons and the information passed between the different processing modules are all encoded in XML. The innovative aspect is the use of a cross document merging algorithm that uses the information extracted from textual sources to produce an integrated, more complete, annotation of the material. Ontology based reasoning and scenarios are used to merge and unify the separate annotations.

The techniques presented here have been implemented in a working demonstration prototype and have been tested on material from the European Championships Soccer 2000.

1 Introduction

The fast growth of the Web makes it increasingly hard to find the right information based on mea-

asures using keywords, vector models, site popularity, etc. Therefore new techniques are needed, to access content based on its meaning. There exist several initiatives that aim at semantic access of Web content. Much work in the Conceptual Graph community for example is centered around this issue. (Nicolas et al. (Nicolas et al., 2002) describe an information retrieval system in which both text documents and queries are translated to a CG representation, Zhong et al. (Zhong et al., 2002) use CG's to retrieve online descriptions of garments, Ounis and Pasca (Ounis and Pasca, 1998) and Myaeng (Myaeng, 1992) also use CG's for information retrieval purposes.)

To achieve semantic web access, several problems have to be solved.

- A formalism has to be developed that supports expression of and reasoning with knowledge. Much work has already been done in this direction, such as for example in the OntoWeb SIG-2¹ or by Motta et al., working on annotation formalisms and reasoning in Web environments (Motta et al., 2000; Domingue and Motta, 2000)

- The Web already contains a massive amount

¹For more information, cf <http://ontoweb.aifb.uni-karlsruhe.de/>

of information that will not be rewritten to fit a knowledge encoding formalism. Furthermore the majority of people writing new texts are probably or not willing, or possibly not able, to enrich these with formal annotations.

Therefore techniques are needed for *automatic* semantic annotation of natural language material.

In the OntoWeb SIG-5 strategies for using language technology for the Semantic Web are discussed. Several of the CG projects mentioned above are concerned with automatic annotation.

- The annotated content should be made available to users.

For this either a sophisticated interface to aid the user in constructing a query directly in the formalism or a translation from natural language queries to the formalism is needed.

- Lastly, the formalized queries need to be matched to the annotated multimedia content to provide the user with exactly the information that was requested.

This paper presents the MUMIS² (Multi-Media Indexing and Searching) project, which addresses all of the above mentioned problems using techniques such as information extraction, automatic speech recognition and keyframe extraction from video content to facilitate multilingual information retrieval on multimedia archives. In addition the MUMIS project contains a module that combines annotations extracted from separate sources into one integrated, more complete, formal description of their content. This so-called cross-document merging of annotations is one of the main issues in this paper.

The rest of the paper is organized as follows: Section 2 gives a general overview of the MUMIS project; Section 3 presents the ontology based information extraction on textual sources in three

²MUMIS is an on-going EU-funded project within the Information Society Program (IST) of the European Union, section Human Language Technology (HLT). Project participants are: University of Twente/CTIT, University of Sheffield, University of Nijmegen, Deutsches Forschungszentrum für Künstliche Intelligenz, Max-Planck-Institut für Psycholinguistik, ESTEAM AB, and VDA.

different languages as well as the different XML formats that are used within the project; Section 4 presents the cross-document merging process; Section 5 shows how the resulting annotations can be used for retrieval of video fragments and the paper end with some conclusions and a short outline of future work.

2 Overview of the MUMIS Project

In the MUMIS project, ontology based information extraction is used to improve the results of information retrieval in multimedia archives. The content used as test case is a collection of video recordings of soccer matches in three different languages (Dutch, English and German).

Though progress has been achieved in the content detection of salient objects and events in a sequence of images, like goals in a soccer game (cf the IST project ASSAVID, (Mukunoki et al., 2001; Assfalg et al., 2002; Assfalg et al., to appear)), automatic indexing and retrieving of image and video fragments solely on the basis of analysis of their visual features is still not really feasible. Many research projects therefore have explored the use of *parallel textual descriptions* of the multimedia information for automatic tasks such as indexing, classifying, or understanding.

Within the MUMIS project these textual descriptions are taken from several sources. Newspaper reports give a very verbose description of the match in general. Formal texts on the soccer matches describe global properties such as which players were in the teams, who scored at what time and whom were given red or yellow cards. So called 'tickers' contain a minute-by-minute report of all salient events in the match. Transcriptions of the speech in the video recordings are the last textual source. It is important to note that for each textual source it is known *a priori* which soccer match is described in it.

The global architecture of the MUMIS demonstrator system is shown in Figure 1. The flow through this architecture is as follows:

1. Multilingual information extraction is applied to sources of different types and in different languages, using an XML-encoded ontology. Each source gives partial (incom-

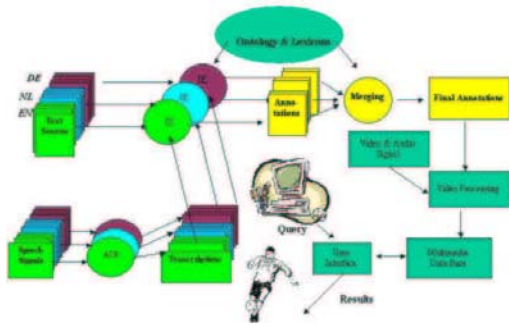


Figure 1: Overview of the MUMIS architecture

plete) information about a soccer match. The resulting knowledge is encoded as lists of events with their attributes, using an XML format.

2. The separate event lists for one match are merged into one cross-document annotation using a merging algorithm developed in this project. This merged annotation, encoded in the same XML format, provides a more complete view on the events that took place during the match.
3. In the next step, the cross-document annotation must be matched to the video recordings of the soccer match, using the time codes and information contained in transcriptions of the speech signal. This results effectively in semantic markup of video fragments.
4. Users can query the system using an interface tailored to the kind of information that has been annotated. The fact that searching is done in the more complete cross document annotations makes the relevance of the returned fragments much more reliable.
5. After determining which fragments are relevant, the corresponding video sequences can be retrieved for the user.

3 Information Extraction

The first step in the process is single-document information extraction on the different textual sources. This means that each text is processed separately, any reasoning that is done does not

cross documents boundaries. This section discusses the domain specific ontology and its relation to the three languages and the three different language specific modules that perform the information extraction using that ontology. The ontology, the lexicons and the information passed to the merging module is all encoded in XML. A mapping of our annotation into RDF and OWL is under investigation at the moment but since that work has not been finished yet this section will use the XML format where needed.

3.1 Ontology

The basic unit of information in the MUMIS project is an *event*, such as KICK-OFF, GOAL, FOUL or SUBSTITUTION. These events are associated with elements such as players, teams and time stamps. An ontology has been developed for these events and the information related to them. Missing concepts were identified and added during manual annotation of some sources.

An important consideration in building the ontology was that concepts were only added if they seemed to be of interest for the final application, information retrieval in video archives. That meant they should possibly be used as subject in a user query.

The link between the ontology and the three languages consists of a flexible XML format which maps concepts to lexical expressions. In this format, every concept can have several children of the class `<term-lang>`. Such a term entry describes a possible lexical realization of a concept. These terms allow several constructions.

- A CAT attribute indicates the part-of-speech class of a lexicon entry. This allows for phrases as well as words to be included as expressing a concept and makes it possible to describe words that belong to distinct categories as synonyms for one concept. In this sense the lexicon reflects the EuroWordNet strategy to allow for cross-category listing of synonyms (Vossen, 1997).
- Regular expression-like syntax can be used to combine entries with the same structure that differ for example only in the choice of

```

<lex-element id="2.4.1.17"
  concept="Out-of-field">
  <term-lang lang="NL" type="synonym"
    cat="EXCL">
    uit!
  </term-lang>
  <term-lang lang="NL" type="main"
    cat="PP">
    {uit|buiten}
    [field-of-play:2.4.1.7]
  </term-lang>
</lex-element>

```

Figure 2: Example of a lexical entry.

a word (such as alternative prepositions expressing the same meaning in a certain context).

- The possibility to refer to another *concept* in relation with a lexical entry, indicating that any lexical instantiation of that concept is allowed at that point in the pattern. E.g. the German verb stem “geh” (goes) with some instantiation of the concepts GOAL and BALL together represent a goal-event.

As an example, consider the XML-fragment in Figure 2. This lexical element describes two possible ways of expressing an OUT-OF-FIELD event in Dutch. The first is a simple exclamation “uit!”; the second is a prepositional phrase with either “uit” or ”buiten” as preposition and any lexical realization of the concept FIELD-OF-PLAY as complement.

3.2 Information Extraction Subsystems

In Section 2 it was already mentioned that there are several sources of textual input. Formal texts are supposed to provide accurate information on the more relevant events (i.e., result, goals), but don’t contain enough information for indexing the whole match. Tickers provide a detailed account on most of the important events, but the temporal information provided by them is not very exact (variations of minutes have been found). Newspaper reports contain little temporal information at all and comments combine information from the actual match with references to related matches (e.g., how a particular player performed in the previous match). Automatic speech transcriptions from the video recordings contain more errors than

the other sources and relatively few events. Instead they contain a larger amount of player names that are mentioned without an indication of the event they took place in. Furthermore the time codes in transcriptions are very exact but have a different base (the beginning of the recording instead of the beginning of the match). But since they are the only link with the video recordings they are needed to map the annotations to the video recordings.

The formal texts and the tickers are the most reliable and informative source of information about what happened during the match. The information extraction modules use NLP techniques to annotate these texts. The English information extraction is based on the GATE architecture (Cunningham., 2002; Saggion et al., 2002) The output of the information extraction modules for the different languages on these texts consists of language independent XML describing all events that have been detected in the text together with their attributes (see Figure 3 for an example of this XML output). This output serves as source material for the language independent module described in the next section.

4 The Merging Module

The most important innovation in this project is the fact that the annotations produced by the information extraction modules are not used separately but are first merged into one single annotation of knowledge about the soccer match. This more complete semantic markup should improve the reliability of search results.

As an example consider the following situation (Netherlands-Yugoslavia match): One of the information extraction components extracted from document A that in the 30th minute of the match a FREE-KICK was taken, but did not discover who took it. It did find the names of two players though: Mihajlovic (a Yugoslavian player) and Van der Sar (the Dutch keeper). From document B a SAVE in the 31st minute was extracted by the information extraction component, and the names of the same two players were recognized. From these two results it now can be concluded that it was Mihajlovic who took the freekick, and that Van der Sar made the save, thus giving a more complete

```

<event_entry>
<event_type>Yellow-card</event_type>
<event_ID>2</event_ID>
<event_time>43</event_time>
<original_doc_name/>
<player_1>Jeremies</player_1>
<team_player_1>Germany</team_player_1>
<player_2></player_2>
<team_player_2></team_player_2>
<location/>
<destination/>
<preceding_event/>
<following_event/>
<score/>
<source_reference/>
</event_entry>

<event_entry>
<event_type>Yellow-card</event_type>
<event_ID>21</event_ID>
<event_time>41</event_time>
<player_1>Jeremies</player_1>
<team_player_1>Germany</team_player_1>
<player_2></player_2>
<team_player_2></team_player_2>
<location></location>
<destination></destination>
<preceding_event></preceding_event>
<following_event></following_event>
<score></score>
<source_reference></source_reference>
</event_entry>

```

Figure 3: Two examples of extracted event entries, one from an English formal text, one from a ticker

picture of what happened in the 30-31st minute of the match.

The rest of this section discusses how this merged representation is obtained.

4.1 Overview of the Merging Process

Information extraction and merging from multiple sources has been tried in the past (Radev and McKeown, 1998) but only for single events. The approach used in MUMIS consists of applying merging to multiple-events extracted from multiple sources.

The MUMIS merging process can be separated into three subproblems.

Alignment: Annotation fragments from different sources describing the same events should be aligned.

Unification: Fragments of annotations selected as describing the same events should be unified into one integrated annotation containing all information from all sources about these events. It is possible that ambiguities or conflicts need to be resolved.

Reordering: Many events in the ticker fragments are mentioned in the wrong order. The merging of unrelated events from different sources also introduces some ambiguity with respect to their order. In the final annotation all events should be present in the order in which they took place.

These three aspects are described in the following subsections, but first *scenes* are introduced as an effective way of grouping events.

4.2 Scenes

When examining the ticker texts and annotations, some observations can be made. Ticker texts consist of small fragments of text separated by time markers indicating the minute in which the described events took place. Ticker writers might be considered as a kind of semantic filter, writing short fragments describing interesting scenes on the field. There seems to be a limited amount of different scenes that are most often used and it seems that different ticker writers group events in

the same way. That does not mean that they all consider the same situations to be important, but if they consider a situation important, they more or less mention the same aspects.

An important observation is the fact that one tickerfragment often contains only one scene of interdependent events.

With respect to the order in which scenes and events are mentioned in the texts the following can be said: most *scenes* (fragments) took place in the order in which they are described in the text, but the events *within a scene* are not necessarily mentioned in the order in which they hapened on the field.

As a result of these observations, the merging algorithm is partly based on the alignment and merging of scenes rather than separate events.

4.3 Alignment

Bi-document alignment: given two source documents A and B, every scene from A is checked for compatibility with every scene from B. In determining the strength of a possible connection between two scenes, various aspects play a role: number of common player names, distance in time, etc. First, the program calculates the strength of all bindings between all pairs of scenes from documents A and B respectively. Suppose that the binding strength between a scene SA from document A and a scene SB from document B is the strongest, then the program concludes that these two scenes are about the same episode in the match, and the combination is confirmed. Choosing the combination rules out certain other combinations from the two documents A and B, e.g. combinations between scenes from document A which are before scene SA and scenes from document B which are after scene SB are eliminated (see the observation on ordering of scenes in source texts in section 4.2). This process is repeated recursively until only confirmed bindings between scenes from documents A and B are left.

Multi-document alignment: the above process is performed on every pair of documents, thus yielding pairs of scenes. The next step is to build sets of scenes over all documents, connected as follows. Create a set consisting of any scene,

and add all scenes to this set which have been connected to this scene by the process in the bi-document step. Repeat this for all these newly added scenes recursively until no new scenes are found which should be added to the set. This set naturally forms a (connected) graph of combined scenes. Notice that the graph is not necessarily complete, i.e. not every pair of scenes in the graph needs to be connected. In fact, scenes may be incompatible and nevertheless occur in the same graph through a sequence of intermediate scenes. Since a graph is supposed to contain scenes from various documents which all are about the same episode during the match, a graph should not contain such scenes which are incompatible in that sense. In order to exclude such scenes from a graph, the program splits a graph into complete subgraphs, such that only graphs remain in which all scenes are connected to all other scenes. This splitting up again is based on the strongest connections in a given graph.

4.4 Unification

The partial events from the various scenes in a given graph are combined and empty slots are filled in. At this point several (semantical) rules expressing domain knowledge are used. There are several kinds of rules to be used at this point. First, event internal rules describe which events are possible (i.e., a keeper will not take a corner) Second, event external rules express possible combinations of events (i.e., a player shooting at goal will belong to the other team as a player who blocks this shot). As a result, more completely filled in events are produced.

4.5 Ordering

Finally the events inside such a scene have to be put into the correct order. For example, a shot on goal in the same scene as a goal typically will take place before that goal and not after. For this ordering process scenarios are used. These scenarios describe 'usual sequences of events' in soccer matches. Within one scene the time codes are not reliable for ordering, since different ticker texts can describe the same scenes as occurring at times which are minutes apart and events in one scene from one document all have the same timestamp.

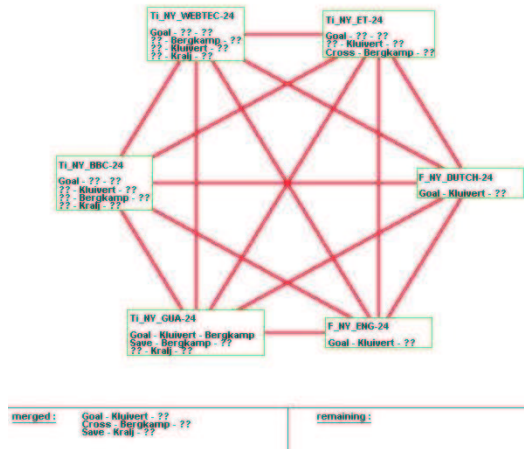


Figure 4: An example of alignment and unification of scenes from six different sources, as produced by the prototype system

4.6 Merger Output

The alignment and unification steps described above are implemented in a functional programming language, resulting in a demonstrator prototype that performs quite well on the first example matches. After performing these three steps, the final result of the merger is a new XML annotation of the match in the same format as used for the information extraction. Figure 4 shows an example of alignment and unification as produced by the prototype system.

5 Retrieval of Multimedia Content

The previous sections described the process through which the textual content of the corpus is annotated with semantic markup. Since the aim of the project is to disclose a *multimedia* archive of soccer matches, these annotations should still be matched to fragments of the video recordings of the soccer matches. Furthermore the fragments should be made available to the user, with the help of the semantic annotations. These two aspects are described further in this section.

5.1 Annotation of Video Recordings

The only textual sources that can be directly matched to the video recordings are the transcripts of the speech in those recordings (Wester et al.,

2002). Even though information extraction results for these transcripts is possible, the transcripts themselves contain more errors than the other sources. Furthermore they contain relatively few actual events but a larger amount of names mentioned without any indication of the event the player took place in. The time codes in transcriptions are very exact but cannot be directly mapped onto the time codes in the tickers. The transcriptions start at the beginning of the recording, whereas the tickers annotate the kick off as minute 0. To facilitate retrieval based on the cross document annotation, a mapping or alignment of the semantic markup produced by the merging module to the annotation of the transcriptions has to be found. The names of the players that participated in certain scenes are the most reliable source of information to achieve this mapping.

5.2 Retrieval of Fragments

To make the annotated content accessible to the user, queries should be formulated in the annotation formalism to match them to the annotated content. In the MUMIS project the focus is on information extraction rather than on retrieval. Therefore the retrieval module does not process natural language queries to obtain the formalized query. An interface has been developed instead that allows the user to enter queries directly in the event-format. The interface makes use of the lexica in the three target languages and the domain ontology to assist the user while entering his or her query. The mapping of annotation to video recordings described in the previous section makes it possible to search in the annotations but return results from video fragments. The hits of the query are indicated to the user as thumbnails in the storyboard together with extra information about each of the retrieved events. The user can select a particular fragment and play it (see Figure 5)

6 Conclusions

MUMIS is the first multimedia indexing project which carries out indexing by applying information extraction to multimedia and multilingual information sources, merging information from many sources to improve the quality of the annotation database, and combining database queries



Figure 5: Screenshot of the User Interface

with direct access to multimedia fragments. The approach taken in this project yields good results in first testing. The next step will be to perform extensive structured tests to obtain an objective evaluation of the quality of the merging algorithm.

References

- J. Assfalg, M. Bertini, C. Colombo, and A. Del Bimbo. 2002. Semantic annotation of sports videos. *IEEE Multimedia*, 9(2):52–60.
- J. Assfalg, M. Bertini, A. Del Bimbo, W. Nunziati, and P. Pala. to appear. Soccer highlights detection and recognition using HMMs. *Proceedings of the International Conference on Multimedia and Expo (ICME2002)*.
- H. Cunningham. 2002. Gate, a general architecture for text engineering. *Computers and the Humanities*, 36:223–254.
- J.B. Domingue and E. Motta. 2000. Planet-onto: From news publishing to integrated knowledge management support. *IEEE Intelligent Systems*, pages 26–32.
- E. Motta, S. Buckingham-Shum, and J. Domingue. 2000. Ontology-driven document enrichment: Principles, tools and applications. *International Journal of Human-Computer Studies*, 52:1071–1109.
- M. Mukunoki, M. Bertini, J. Assfalg, and A. Del Bimbo. 2001. Classification of raw material sports videos for broadcasting using color and edge features. *Proc. of Int'l Conf. on Multimedia and Expo (ICME2001)*.
- S.H. Myaeng. 1992. Using conceptual graphs for information retrieval: A framework for adequate representation and flexible inferencing. In *Proc. of Symposium on Document Analysis and Information Retrieval*.
- S. Nicolas, G.W. Mineau, and B. Moulin. 2002. Extracting conceptual structures from english texts using a lexical ontology and a grammatical parser. *Foundations and Applications of Conceptual Structures, supplementary proceedings of the 10th International Conference on Conceptual Structures*.
- I. Ounis and M. Pasca. 1998. A promising retrieval algorithm for systems based on the conceptual graphs formalism. In B. Eaglestone, B.C. Desai, and Jianhua Shao, editors, *Proceedings of the International Database Engineering & Application Symposium (IDEAS'98)*, pages 121–130.
- H. Saggion, H. Cunningham, K. Bontcheva, D. Maynard, O. Hamza, and Y. Wilks. 2002. Multimedia indexing through multi-source and multi-language information extraction: The mumis project. *Data & Knowledge Engineering Journal*.
- P. Vossen. 1997. Eurowordnet: a multilingual database for information retrieval. In *Proceedings of the DELOS workshop on Cross-language Information Retrieval*.
- M. Wester, J.M. Kessens, and H. Strik. 2002. Goal-directed asr in a multimedia indexing and searching environment (mumis). In *7th International Conference on Spoken Language Processing, ICSLP 2002, INTERSPEECH 2002*.
- Manuel Montes y Gomez, Aurelio Lopez, and Alexander F. Gelbukh. 2000. Information retrieval with conceptual graph matching. In *Database and Expert Systems Applications*, pages 312–321.
- Jiwei Zhong, Haiping Zhu, Jianming Li, and Yong Yu. 2002. Conceptual graph matching for semantic search. In U. Priss, D. Corbett, and G. Angelova, editors, *Conceptual Structures: Integration and Interfaces. Proceedings of the 10th International Conference on Conceptual Structures*. LNAI 2393.