Intelligent Multimedia Indexing and Retrieval through Multi-source Information Extraction and Merging*

Jan Kuper[§], Horacio Saggion^ Hamish Cunningham^ Thierry Declerck*,
Franciska de Jong[§], Dennis Reidsma^, Yorick Wilks*, Peter Wittenburg[8]

[5] Department of Computer Science, University of Twente, The Netherlands; *Department of Computer Science,
University of Sheffield, Sheffield, UK; *DFKI GMbH, Saarbruecken, Germany; "Max Planck Institute for
Psycholinguistics, Nijmegen, The Netherlands email: jankuper@cs.utwente.nl, saggion@dcs.shef.ac.uk

Abstract

This paper reports work on automated meta-data creation for multimedia content. The approach results in the generation of a conceptual index of the content which may then be searched via semantic categories instead of keywords. The novelty of the work is to exploit multiple sources of information relating to video content (in this case the rich range of sources covering important sports events). News, commentaries and web reports covering international football games in multiple languages and multiple modalities is analysed and the resultant data merged. This merging process leads to increased accuracy relative to individual sources.

1 Introduction

Multimedia repositories of moving images, texts, and speech are becoming increasingly available. This together with the needs for 'video-on-demand' systems requires fine-grain indexing and retrieval mechanisms allowing users access to specific segments of the repositories containing specific types of information. Annotation of video is usually carried out by humans following strict guidelines. Video material is usually annotated with 'meta-data' such as names of the people involved in the production of the visual record, places, dates, and keywords that capture the essential content of what is depicted. Still, there are a few problems with human annotation. Firstly, the cost and time involved in the production of fine-grained semantic "surrogates" of the programme is extremely high; secondly, humans are rather subjective when assigning descriptions to visual records; and thirdly, the level of annotation required to satisfy a user's needs can hardly be achieved with the use of mere keywords. In order to tackle these problems, indexing methods based on image processing have been developed [Chang *et al,* 1998]. Content-based indexing and retrieval of visual records is based on features such as colour, texture, and shape. Yet visual understanding is not well advanced and is very difficult even in closed domains. As a consequence, various ways to explore the use of collateral linginstic material have been studied for tasks such as automatic indexing [De Jong *et al,* 2000], classification [Sable and Hatzivassiloglou, 1999], or understanding [Srihari, 1995] of visual records.

In this paper, we present an integrated solution to the problem of multimedia indexing and search: the MUM1S[1] concept. Our solution consists of using information extracted from different sources (structured, semi-structured, free, etc.), modalities (text, speech), and languages (English, German, Dutch) all describing the same event to carry out data-base population, indexing, and search. The novelty of our approach is not only the use of these 'heterogeneous' sources of information but also the combination or cross-source fusion of the information obtained from the separate sources. Single-document, single-language information extraction is carried out by independent systems that share a semantic model and multi-lingual lexicon of the domain. The results of all information extraction systems are merged by a process of alignment and rule-based reasoning that also uses the semantic model. In the rest of this paper we describe in detail the context of the project, and each of the modules involved in the automatic derivation of annotations. However, the emphasis is on the merging component.

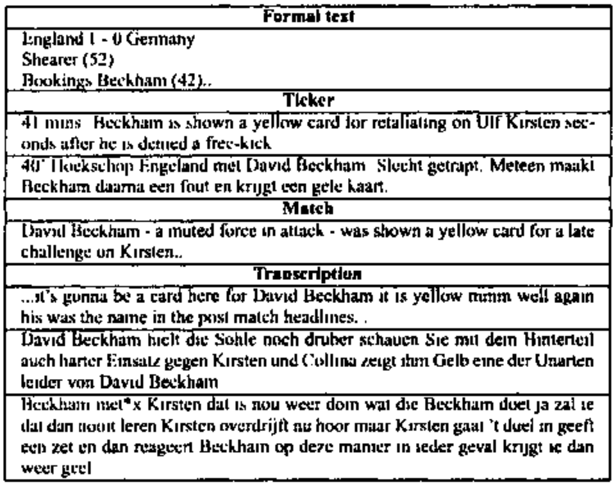
2 The MUM1S Project

In MUM1S various software components operate off-line to generate formal annotations from multi-source linguistic data in Dutch, English, and German to produce a composite time-coded index of the events on the multimedia programme. The domain chosen for tuning the software components and for testing is football.

A corpus of collected textual data in the three languages was used to build a multi-lingual lexicon and shared ontology of the football domain. Based on this shared model, three different off-line Information Extraction components, one per language, were developed (see section 3). They are used to extract the key events and actors from football reports and to produce XML output. A merging component or cross-document co-reference mechanism has been developed to merge the information produced by the three IE systems (see section 4). Audio material is being analysed by Phicos [Steinbiss *et al,* 1993], an HMM-based recognition system,

"Funded by EC's 5th Framework HLT Programme under grant number 1ST-1999-10651

[1] Multimedia Indexing and Searching Environment, see http://parlevink.cs.utwente.nl/projects/mumis/



Formal text
England 1 - 0 Germany Shearer (52) Bookings Beckham (42)..
Ticker
41 mins Beckham is shown a yellow card for retaliating on Ulf Kirsten seconds after he is denied a free-kick
40' Hoekschop Engeland met David Beckham Slecht getrapt. Meteen maakt Beckham daarna een fout en krijgt een gele kaart.
Match
David Beckham - a muted force in attack - was shown a yellow card for a late challenge on Kirsten..
Transcription
...it's gonna be a card here for David Beckham it is yellow mmm well again his was the name in the post match headlines. .
David Beckham hielt die Sohle noch druber schauen Sie mit dem Hinterteil auch harter Einsatz gegen Kirsten und Collina zeigt ihm Gelb eine der Unarten leider von David Beckham
Beckham met*x Kirsten dat is nou weer dom wat die Beckham doet ja zal ie dat dan nooit leren Kirsten overdrijft nu hoor maar Kirsten gaat 't duel in geeft een zet en dan reageert Beckham op deze manier in ieder geval krijgt ie dan weer geel

Table 1: Different accounts of the same event in different languages

in order to obtain transcriptions of the fooiball commentaries (spontaneous speech). It uses acoustic models, word-based language models (unigram and bigram) and a lexicon. For Dutch, English, and German different recognition systems have been developed (i.e., different phone sets, lexicons, and language models are used).

JPEG keyframe extraction from MPEG movies around a set of pre-defined time marks - result of the information extraction component - is being carried out to populate the database. The on-line part of MUMIS consists of a state of the art user interface allowing the user to query the multimedia database. The interface makes use of the lexica in the three target languages and domain ontology to assist the user while entering his/her query. The hits of the query are indicated to the user as thumbnails in the story board together with extra information about each of the retrieved events. The user can select a particular fragment and play it.

2.1 Domain and Ontology

An analysis of the domain and a user study led us to propose 31 types of event for a football match *{kick-off, substitution, goal, foul, red card, yellow card,* etc.) that need to be identified in the sources in order to produce a semantic index. The following elements associated with these events are extracted: players, teams, times, scores, and locations on the pitch.

An ontology has been developed for these events and their actors, it contains some 300 concept nodes related as an 'is-a' hierarchy. The link between the concepts and the three languages consists of a flexible XML format which maps concepts into lexical entries.

Sources used for information extraction are: formal texts, tickers, commentaries, and audio transcriptions (see Table 1). Ticker reports are particularly important in the generation of formal annotations. These texts are a verbal account of events over time stamps. They also follow a specific text structure consisting of a 'ticker header', in which information about lists of players and the result of the game is usually stated, and 'ticker sections' grouping together sentences describing events under single time stamps. Another very valuable source for the generation of the annotations are the spoken transcriptions that, even with the many errors they contain, still provide exact temporal information.

3 Extracting Information from Heterogeneous Sources

Information extraction is the process of mapping natural language into template-like structures representing the key (semantic) information from the text. These structures can be used to populate a database, used for summarization purposes, or as a semantic index as in our approach. Key to the information extraction process is the notion of "domain", scenario or template that represents what information should be extracted. IE has received a lot of attention in the last decade, fuelled by the availability of on-line texts and linguistic resources and the development of the Message Understanding Conferences [Grishman and Sundheim, 1996]. Traditionally, IE applications have tended to concentrate on a small number of events (typically one), MUMIS addresses the challenge of multi-event extraction.

Multi-lingual IE has been tried in the M-LaSIE system [Gaizauskas *et al,* 1997], where the same underlying components and a bi-lingual dictionary are used for two different languages (English and French). MUMIS differs from that system in that it operates with three different off-line information extraction components, one per language, that produce the same "language-free" representation. In this paper we give only a brief description of the English and German IE systems.

3.1 Extraction from English Sources

IE from English sources is based on the combination of GATE[2] components for finite state transduction [Cunningham *et al.,* 2002] and Prolog components for parsing and discourse interpretation. The components of the system are: tokeniser, segmenter, gazetteer lookup (based on lists of entities of the domain), semantic tagger, shallow pronominal co-referencer, part-of-speech tagger, lemmatiser, chart parser, discourse interpreter (ontology-based co-referencer), and template extractor. These components are adapted and combined to produce four different system configurations for processing different text-types and modalities (transcriptions, formal texts, semi-formal texts, and free texts). The analysis of formal texts and transcriptions is being done with finite state components because the very nature of these linguistic descriptions make appropriate the use of shallow natural language processing techniques. For example, in order to recognise a *substitution* in a formal text it is enough to identify players and their affiliations, time stamps, perform shallow co-reference and identification of a number of regular expressions to extract the relevant information. In our system, regular expressions operate on annotations (not on strings) and produce semantic information. We make use of the Java Annotation Pattern Engine (JAPE) formalism [Cunningham *et al.,* 2002] to code our regular grammar. Below, we present

[2] GATE is a free architecture for natural language engineering.

one example of the use of JAPE that accurately identify *substitutions* in speech transcriptions:

```
Rule: Subs1
((({Player}) ({Token.string == ''is''}) ({Replace})
({Player})):annotate  --> :annotate.SubsEvent =
{rule = "Subs1"}
```

Complex linguistic descriptions are fully analysed because of the need to identify logical subjects and objects as well as to solve pronouns and definite expressions (e.g., "the Barcelona striker") relying on domain knowledge encoded in the ontology of the domain. Domain knowledge establishes, for example, that the two players involved in a *substitution* belong to the same team. This semantic constraint is used in cases such as "he is replaced by Ince" to infer that the antecedent of the pronoun "he" belongs to the "English" team (because "Ince" does).

Logic-based information extraction rules operate on logical forms produced by the parser and enriched during discourse interpretation. They rely on the ontology to check constraints (e.g., type checking, ontological distance, etc.). The following logic-based rule is used to extract the participants of a substitution when syntactic and semantic information is available:

```
replace_event(E) & lsubj(E,P1) & lobj(E,P2)
& player(P1) & player(P2) => substitution_event(E) &
player 1(E,P1) &  player 2(E,P2)
```

Msubj' and iobj represent the logical subject and object of an event. Note that, contrary to the regular case, the 'lsubj' and iobj' relations, being semantic in nature, are long distance relations.

3.2 Extraction from German Sources

The German IE system is based on an integrated set of linguistic tools called SCHUG: Shallow and Chunk based Unification Grammar [Declerck, 2002]. The chunking procedure of SCHUG consists of a rule-based sequence of cascades (based on the work by [Abney, 1996]), which produces a rich linguistic representation, including grammatical functions and resolution of co-reference and ellipsis. In order to detect these accurately, an analysis of the clauses of a sentence is required. Clauses are subparts of a sentence that correspond to a (possibly complex) semantic unit. Each clause contains a main verb with its complements (grammatical functions) and possibly other chunks (modifiers).

Applied to the football domain, SCHUG inspects the common MUMIS ontology and enriches the linguistic annotation produced with domain-specific information encoded in the ontology. Below (see Table 2) we show one example of the semantic annotation generated by SCHUG when applied to an on-line ticker text (game England-Germany). Here, various relations (player, location, etc.) and events *{free-kick, fail to score a goal)* that are relevant to the football domain are recognised.

```
7.  Ein Freistoss von Christian
Ziege aus 25 Metern geht ueber das Tor.
''A 25-meter free-kick by Christian
Ziege goes over the goal.''

<events>
<event id="e1" clause="cls1" event-name=
"free-kick">
<relations>
<relation id="r1" reltype="player"
player-name="Christian Ziege" in-team="Germany"/>
...
</relations>
</event>
<event id="e2" clause="cls1" event-name=
"goal-scene-fail">
...
</event>
</events>
```

Table 2: Semantic annotation in SCHUG

Some relations are not explicitly mentioned, but can still be inferred by the MUMIS system. For example, the team for which "Ziege" is playing can be inferred from the ontological information that a player is part of a team and the instance of this particular team can be extracted from additional texts or meta-data. In this way, information not present in the text directly can be added by additional information extraction and reasoning.

Since formal texts require only little linguistic analysis, but rather an accurate domain-specific interpretation of the jargon used, a module has been defined within SCHUG, which in a first step maps the formal texts onto an XML annotation, giving the domain semantic of the expressions in the text (the approach taken for formal texts is similar to the one followed by the English IE system). In a second step SCHUG *merges* all the XML annotated formal texts about one game. Those merged annotations are generated at a level that requires only little linguistic analysis, and basically reflect domain specific information about actors and events involved in the text. The SCHUG module applied at this level also extracts meta-data information: name of the game, date and time of the game, intermediate and final scores etc. This is quite important, since the meta-data can guide the use of the annotations produced so far for supporting linguistic analysis and information extraction applied to more complex documents.

4 Merging

Merging, also known as cross-document co-reference [Bagga and Baldwin, 1999], is the process of deciding whether two linguistic descriptions from different sources refer to the same entity or event. The merging component in MUMIS combines the partial information as extracted from various sources, such that more complete annotations can be obtained. Radev and McKeown [1998] developed a knowledge-based multi-document summarization system based on information extraction and merging of single 'terrorist' events. The novelty of our approach consists in applying merging to multiple events extracted from multiple sources.

As is to be expected, complete recognition of events in natural language sentences is extremely difficult. Often, events will be only partially recognised. The merging component of the MUMIS project aims to fill in missing aspects of events with information gathered from other documents. For example, the Dutch information extraction system recognised in document *A* on the match Netherlands-Yugoslavia from the European championships 2000 that a *save* was performed in the 31st minute. In addition, it recognised the names of two players: Van der Sar (the Dutch goalkeeper) and Mihajlovic (a Yugoslavian player), but it could not figure out which of these two players performed the save. In document *B* it recognised a *free-kick* in the 30th minute, and the names of the same two players. Again, it did not succeed in finding out which player took the *free-kick.*

The fact that the same two players are involved, plus the small difference between the time-stamps, strongly suggests that both descriptions are about the same event in reality. The merging part of the MUM1S project matches these partial data together, and concludes that it was Mihajlovic who took the *free-kick,* followed by a *save* by Van der Sar.

The merging component consists of several parts which will be described in more detail below.

4.1 Scenes

Given the example above, it is clear that matching together individual events from two different documents is not the right approach: a *save* event cannot be matcled with a *free-kick* event, they are two totally different events. Besides, it is clear that players' names will play an important role in the matching of information from one document with information from another document. In order to take players' names into consideration, an *unknown* event was introduced, such that if it was not clear what a player did, this could be represented by letting that player perform this unknown event (the information extraction systems provide this information).

Now the event is still the fundamental concept, but the merging process aims at matching together *groups* of events instead of single events. Such a group of events is called a *scene.* The author of a text is considered as a "semantic filter", which determines which events should be taken together in the same scene. If events are mentioned in the same text fragment, they belong to the same scene. In the ticker documents this does not give rise to ambiguities, since their text fragments are clearly distinguished from each other (see Table 1).

4.2 Two Document Alignment

The merging algorithm compares all the scenes extracted from document *A* with all the scenes extracted from document *B,* and examines whether or not a scene from document *A* might be matched with a scene from document *B.* There are several aspects to be taken into account: players involved in the scenes, distance between time stamps, whether the scenes contain the same events or not, etc.

A scene from document *A* may match more than one scene from document *B* (see Figure 1).

The strength of a matching can be calculated in different ways, and it will not be surprising that the number of players involved in both scenes will be of great influence. This influence is that great, that taking only the number of common players' names as a measure of the strength of a binding, gave the best results. We restricted matching of scenes to those scenes which were not further than (arbitrarily) five minutes apart.

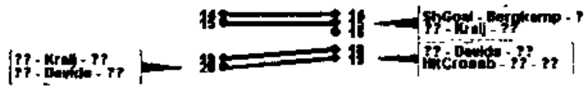


Figure 1: Two document alignment. Vertical lines denote documents, numbers are time stamps, thin lines indicate possible bindings, thick lines denote strongest bindings.

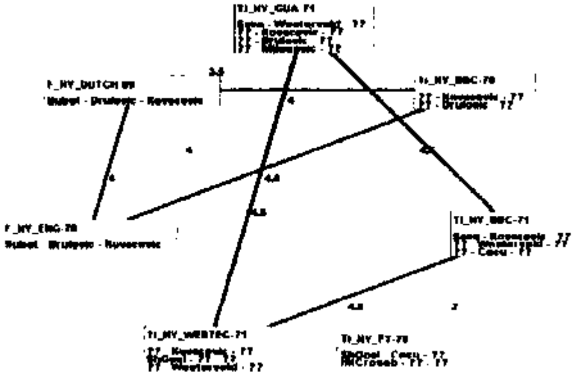


Figure 2: Finding complete subgraphs. The graph as a whole is found by the alignment process. Inside 3 complete subgraphs are found (two triangles indicated by the thick lines, and one single node).

In order to choose the best matchings, the algorithm starts by selecting two scenes *s* and s_2 such that the binding between s_1 and S_2 is (one of) the strongest. In general, such a choice will remove other bindings, in particular the bindings between .si and .s_2 with other scenes. Since these two describe the same fragment of the game, scenes before s_1 can now no longer match scenes after s_2 (and vice versa). That is to say, bindings which "cross" the matching of s_1 and s_2 are removed. Thus the possible bindings are cut in two parts, and the algorithm continues recursively with both halves, until all choices have been made. This two document algorithm is applied to every pair of documents.

4.3 Multi Document Alignment

The next step is to join connected scenes from various documents together. Start with a set consisting of one (arbitrary) scene, and extend this set by those scenes which are connected to the starting scene. Repeat this for all the scenes added to the set until no further extension is found. This set of scenes, together with the bindings between the scenes (chosen by the two document algorithm) naturally form a *connected graph.*

Repeat this graph building procedure for the remaining scenes, until all scenes are included in a graph. Notice, that such a graph may consist of one scene only. Notice also, that a graph may contain more scenes than there are documents, since scenes may be connected through a sequence of two document matches.

4.4 Complete Sub-graphs

Ideally, a graph should be *complete,* expressing that every two scenes in the graph do match, and thus all scenes do contain some common information. That is to say, the scenes in a complete graph are all about the same fragment during the football game which is described in the documents. However, in practice not every graph which results from the procedure above, is complete (see Figure 2).

Scenes may partially overlap, and thus give rise to sequences of connections where the first and the last arc no longer connected. We isolate the *strongest complete subgraphs* from every non-complete graph, since such a subgraph describes one fragment in reality. A given (non-complete) graph is divided into its strongest complete subgraphs by going through all the bindings in the graph, ordered by their strength (starting with the strongest one), and adding scenes and edges to a sub-graph whenever possible without violating the completeness of this sub-graph. If that is not possible, a new sub-graph is started, and some bindings may be removed. The final result is a set of complete graphs of matching scenes from many documents, such that the scenes inside a graph describe the same fragment in reality.

4.5 Rules

Consider the example given before, about the *save* event and the *free-kick* event, with the players Van der Sar and Mihajlovic. These scenes are now in the same graph, and thus may be combined (see Figure 3). However, not every combination will be correct, for example, Mihajlovic and Van der Sar will not both take the free-kick. Also, it will not be correct to let Van der Sar (the Dutch keeper) take the free kick, and Mihajlovic perform the save. In order to combine this partial information into scenes containing complete events, *rules* are needed. In the MUMIS project several kinds of rules, all expressing some *domain knowledge,* have been developed. A first kind are the event internal rules, for example, rules saying that the two players involved in a substitution event belong to the same team, or that a keeper typically will not take a corner.

A second kind of rule takes into account the role of the teams in the fragment, i.e., whether a team is attacking or defending at that moment. To determine whether a team is attacking or defending, all players involved in the scene are checked for their normal position in the field. Then events arc characterised as offensive events (such as *corner, shot on goal),* defensive events *(save, clearance),* and neutral events *(throw-in, yellow card).* The second kind of rule makes sure that offensive events are performed by players from the attacking team, and defensive events are combined with payers from the defending team.

A third kind of rule makes sure that players will not perform impossible combinations of events, e.g., the player who takes a corner will (unless he's very fast) not deliver a shot-on-goal as well.

To apply these rules, a background database of player information is created, containing names of players, their normal position in the field, the team they belong to, etc.

Yet another kind of rule is based on an ontolgy of events and is used to unify certain events which according to the chosen semantics in MUMIS are not the same. For example, a clearance and a save are two different events according to the MUMIS semantics, but an author may well use these terms in a way different from the MUMIS semantics. Therefore the relationships between such concepts are expressed in an ontology, such that rules may use sub-typing or super-typing relationships between concepts.

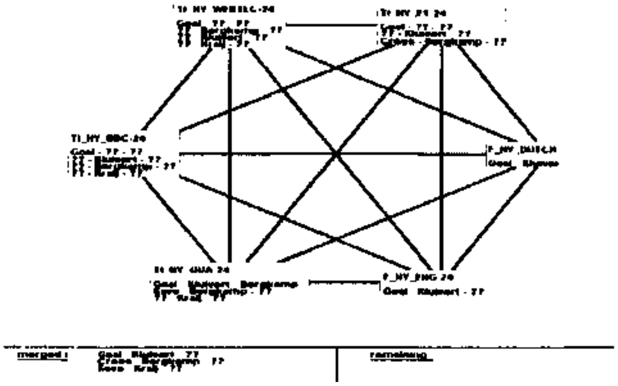


Figure 3: Merging. Bottom left is the result of merging all information in all nodes of the graph. Bottom right is the remaining information that could not be decided upon (in this case nothing).

4.6 Scenarios

After merging the scenes extracted from the various documents into more complete scenes, the order of the events within a scene may be incorrect with respect to the order as it was in reality. The merging process itself does not take the ordering of events into account, and besides, authors often mention events in the opposite order. For example, a player scores after a corner - but the scoring is mentioned first.

Based on the original texts in the source documents, a series of *scenarios* is extracted, describing typical orders in which events occur [Schank and Abelson, 1977].

4.7 Evaluation

Given the limited availability of background information on players (to which team they belong, on what position they play, etc.), the merging component could only be tested in a *case study.* The match Netherlands-Yugoslavia from the European championships in 2000 to perform such a case study. Based on this example, the results of the merging approach are very promising.

The result of the alignment process applied to this match produced 63 complete graphs, of which 28 consist of only one node (a scene), containing information from one source text only. Looking at the original texts, only five of these one-scene graphs might have been combined with other graphs, the main problem being that some texts mention some information in one scene, whereas other documents divide the same information over several scenes. In such cases the alignment algorithm chooses the best match, leaving the other part unmatched.

Of the remaining 35 multi scene graphs, no graph contained unjustified bindings. This seems to be a very promising aspect, since the rules used in the graph forming part of the algorithm are formulated in a very general way, without using any specific information on the concrete football match.

The result of the merging process consisted partly of the elimination of several errors caused by syntactical ambiguities in the single document information extraction, and partly of joining together partial events into more complete events. Like the alignment process, this merging process is also based on general semantical rules, which do not use any concrete details about the specific football match taken as a case study. There were no errors introduced by these merging rules, and in many cases the quality of the extracted information improved considerably.

5 Conclusions and Future Work

The development of huge multimedia databases and the need for accurate navigation of its content require a new generation of "intelligent" tools for producing fine-grained surrogates or indices of the multimedia content. By taking advantage of ontology, domain lexica, and a "language-free" representation of the contents, MUM1S facilitates conceptual search overcoming the keyword barrier.

MUMIS takes advantage of reliable, but coarse-grained content, obtained from formal texts and fine-grained, but sometimes partial, content obtained from free texts and transcriptions. It indirectly solves problems of incomplete information found in any source by combining results from multiple sources. By relying on the analysis of textual instead of visual sources, MUMIS makes possible the derivation of fine-grained semantic indices.

Cross-document co-reference is still in its infancy, MUMIS advances research in that direction by providing a methodology that uses robust named entity alignment from multiple sources together with domain specific, semantic rules.

There are still many points on which the merging algorithm may be improved. For example, some documents are more reliable than other documents. Furthermore, not all texts are of the same type, so-called formal texts differ from ticker texts, and from more free texts. As yet, differences in quality of the source documents has not been taken into account, and a weighted integration of formal texts and other texts still has to be performed. A related point of improvement is to check the consistency of certain elements of information in comparison to other elements of information, and in particular to the state on the football field as may be derived from all previously mentioned events. Finally we mention the improvements in connection to overlapping scenes, where events mentioned in one text fragment in one document may be spread over several text fragments in another document. At this point improvements are also possible.

Acknowledgements

We wish to thank Eduard Hoenkamp and Marco Puts for many useful discussions.

References

[Abney, 1996] Steven Abney. Partial parsing via finite-state cascades. In *Workshop on Robust Parsing, 8th Europen Summer School in Logic, Language and Information* (ESSLLI), 1996.

[Bagga and Baldwin, 1999] A. Bagga and B. Baldwin. Cross-document corcference: Annotations, Experiments, and Observations. In *Proceeding of the Workshop on Corcference and its Applications* (ACL99), 1999.

[Change^/., 1998] S.F. Chang, W. Chen, H.J. Meng, H. Sundaram, and D. Zhong. A Fully Automated Content-based Video Search Engine Supporting Spatio Temporal Queries. *IEEE Transactions on Circuits and Systems for Video Technology,* 8(5), 1998, pp. 602-615.

[Cunninghams*al,* 2002] H. Cunningham, D. Maynard, K. Bontcheva, and V. Tablan. GATE: A framework and graphical development environment for robust NLP tools and applications. In *ACL2002,* 2002.

[Declerck, 2002] Thierry Declerck. A set of tools for integrating linguistic and non-linguistic information. In *Proceedings ofSAAKM 2002, ECAI-2002,* Lyon, 2002.

[Gaizauskas *etal,* 1997] R. Gaizauskas, K. Humphreys, S. Azzam, and Y. Wilks. Concepticons *vs.* lexicons: An architecture for multilingual information extraction. In M.T. Pazienza, editor, *SCIE-97,* LNCS/LNAI, Springer-Verlag, 1997, pp. 28-43.

[Grishman and Sundheim, 1996] R. Grishman and B. Sundheim. Message understanding conference - 6: A brief history. In *Proceedings of the 16th International Conference on Computational Linguistics,* Copenhagen, June 1996.

[De Jong *etal.,* 2000] F.M.G. de Jong, J.-L. Gauvain, D. Hiemstra, K. Netter, Language-Based Multimedia Information Retrieval". In *Proceedings ofRIAO,* 2000, Paris, pp. 713-725.

[Radev and McKeown, 1998] R. Radev and K.R. McKeown. Generating Natural Language Summaries from Multiple On-Line Sources. *Computational Linguistics,* 24(3), September 1998, pp. 469-500.

[Sable and Hatzivassiloglou, 1999] C. Sable and V. Hatzivassiloglou. Text-based approaches for the categorization of images. In *Proceedings of ECDL,* 1999.

[Schankand Abelson, 1977] R. Schank and R. Abelson. *Scripts, Plans, Goals and Understanding.* Lawrence Erlbaum Associates, Publishers, 1977.

[Srihari, 1995] R.K. Srihari. Automatic Indexing and Content-Based Retrieval of Captioned Images. *Computer,* 28(9), September 1995, pp. 49-56.

[Steinbiss et a/., 1993] V. Steinbiss, H. Ney, R. Haeb-Umbach, B.-H. Tran, U. Essen, R. Kneser, M. Oerder, H.-G. Meier, X. Aubert, C. Dugast, and D. Geller. The Philips Research System for Large-Vocabulary Continuous-Speech Recognition. In *Proc. of Eurospeech '93,* Berlin, Germany, 1993, pp. 2125-2128.