Web-Assisted Annotation, Semantic Indexing and Search of Television and Radio News

Mike Dowman* Valentin Tablan* Hamish Cunningham* Borislav Popov[†]

*Department of Computer Science,
University of Sheffield
Sheffield, S1 4DP, UK
+44 114 22 21800

{mike, valyt, hamish}@dcs.shef.ac.uk

[†]Ontotext Lab, Sirma AI EAD,
135 Tsarigradsko Chaussee,
Sofia 1784, Bulgaria
+359 2 9768 310

borislav@sirma.bg

ABSTRACT

The Rich News system, that can automatically annotate radio and television news with the aid of resources retrieved from the World Wide Web, is described. Automatic speech recognition gives a temporally precise but conceptually inaccurate annotation model. Information extraction from related web news sites gives the opposite: conceptual accuracy but no temporal data. Our approach combines the two for temporally accurate conceptual semantic annotation of broadcast news. First low quality transcripts of the broadcasts are produced using speech recognition, and these are then automatically divided into sections corresponding to individual news stories. A key phrases extraction component finds key phrases for each story and uses these to search for web pages reporting the same event. The text and meta-data of the web pages is then used to create index documents for the stories in the original broadcasts, which are semantically annotated using the KIM knowledge management platform. A web interface then allows conceptual search and browsing of news stories, and playing of the parts of the media files corresponding to each news story. The use of material from the World Wide Web allows much higher quality textual descriptions and semantic annotations to be produced than would have been possible using the ASR transcript directly. The semantic annotations can form a part of the Semantic Web, and an evaluation shows that the system operates with high precision, and with a moderate level of recall.

Categories and Subject Descriptors

H.3.1 Content Analysis and Indexing – *linguistic processing*; I.2.7 Natural Language Processing – *language models, speech recognition and synthesis, text analysis*; G.3 [Mathematics of Computing]: Probability and Statistics – *probabilistic algorithms.*

General Terms

Algorithms, Measurement, Design, Experimentation.

Keywords

Multi-media, natural language processing, automatic speech recognition, Semantic Web, web search, media archiving, topical segmentation, key-phrase extraction, semantic annotation.

WWW 2005, May 10-14, 2005, Chiba, Japan.
ACM 1-59593-046-9/05/0005.

1. INTRODUCTION

The problem that the work described in this paper sought to address was that of how to improve access to the large amounts of broadcast audio and visual material produced by media organizations. Material can only be effectively accessed if meta-data describing it is available in some sort of cataloguing system. Production of such meta-data normally requires manual annotation by an archivist, a time consuming and hence costly task. This paper describes a system, Rich News, that can annotate audio and video files automatically, producing both textual descriptions and summaries, and semantic annotations that can form part of the Semantic Web.

The British Broadcasting Corporation (BBC), who produced the material on which Rich News was developed, produce material for four television channels, nine network radio stations, and numerous local radio stations. Annotation of this material is an expensive and labor-intensive task. For example, it takes a BBC archivist almost seven hours to catalog *Newsnight*, a fifty minute daily news broadcast, in detail[1]. Because of the high cost of cataloging, 90% of the BBC's output is annotated only at a very basic level, making it difficult to re-use it after its initial broadcast. Furthermore, because of the time it takes for cataloging to be completed, there is a delay before the material is available, which can be a problem in areas such as news and current affairs, when the material is most likely to be useful immediately after it is broadcast.

A system that was able to automate, or partly automate the annotation process would be very useful. While producing a system that annotates as accurately and with as much detail as a human annotator does is beyond the scope of present technology, it would seem that a system that provided less detailed and less reliable annotations would still be useful. With such an annotation system inaccuracies or omissions might prevent access to some material, or suggest that material was relevant when it was not. However, at present no annotations are produced for much broadcast output, preventing effective access to it, so a level of performance for such a system of well below 100% would be acceptable. In addition, the automatic linking of web and multimedia content enables a new model of mixed-mode media consumption [8].

[1] This information was supplied by Richard Wright of the BBC archives.

Blinkx and Google[2] have both recently launched television search engines, but those systems rely on a simple text-matching search, and do not use the inherent structure of broadcasts to aid in the retrieval process. Previous work has adopted similar information extraction technologies to those used here (see for example [19]), but our work is novel in both the use of web-based content augmentation and in the use of semantic annotation [17]. The annotation process starts by first performing automatic speech recognition to achieve a rough transcript for each program, and then analyzing this transcript to try to determine the boundaries between the various news stories that it describes. This task is made difficult due to errors in the output of current large vocabulary speech recognition systems. Rich News then tries to find key words or phrases that describe the content of each story. Using these key phrases, and the date of the program, it is possible to search on the BBC website to find web pages that are likely to be related to the story. By downloading the candidate web pages and comparing their text to the transcript of the broadcast, it is usually possible to find a web page reporting the same news story that was in the broadcast. The section of the web page that the news story is in can give us a classification for the story, which in some cases is quite detailed, such as the particular English county it relates to. We can also extract summaries and titles for the stories from meta-data in the web-pages.

Furthermore, because the text in the web-pages is error free, and contains useful cues such as capitalization and punctuation that is missing from the transcripts, it is much easier to use this data as a basis for further analysis. The KIM information extraction system was therefore used to find entities in the web pages related to each story, and these were annotated with semantic classes, allowing the stories to be indexed and queried in much more flexible ways than if text search alone were used.

Rich News therefore allows high quality textual and semantic meta-data to be produced fully automatically for news broadcasts. The resulting annotations can be viewed together with the original media file in a multi-media annotator, thus allowing the annotations to be searched, manually corrected or for supplementary annotations to be added by an archivist. Rich News can then produce index documents for individual news stories, containing links to the recordings of the broadcasts in which they occur, as well as textual and semantic meta-data. These can be searched using the Web User Interface (the Web UI) of the KIM system. Most of the system, including parts of KIM, was developed using the GATE natural language processing architecture [7], which allowed rapid development, because many pluggable components were already available, and which facilitated the modular design of the system, making it easy to develop and maintain.

The overall annotation system can be divided into seven modules, as shown in Figure 1. These modules must execute sequentially, as each builds on the output of the previous one, with the exception that the module allowing manual annotation is optional, as it is simply there to allow correction of annotations produced automatically, and addition of those omitted by the automatic system. The first four modules are firstly a speech recognition module, secondly a module that divides the broadcast into segments corresponding to individual news stories, thirdly a module that finds key words for each story, and a fourth module that finds web pages that report the same story as that reported in the broadcast. At this stage, manual annotation may be undertaken. The penultimate module makes a *story index document* for each story in the broadcast, and the final module, KIM, performs information extraction and semantic annotation on the text of the web document, thus allowing the named entities in the broadcast story to be identified. This paper proceeds by discussing each of these modules in turn.

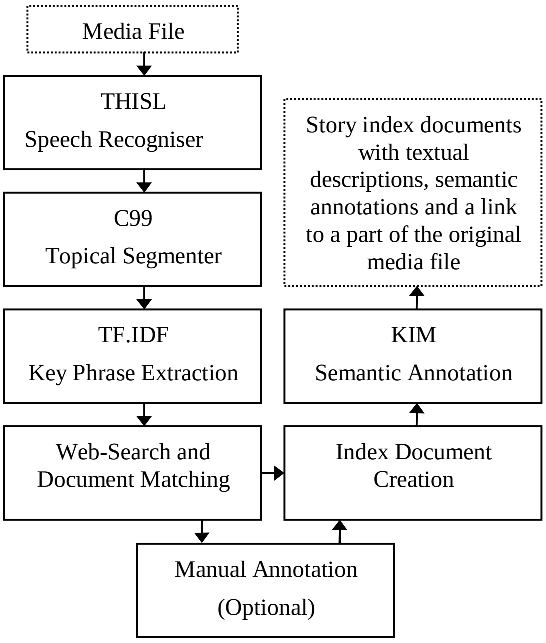


Figure 1. Architecture of Rich News Annotator

2. Speech Recognition

Speech recognition was achieved with the THISL speech recognition system [21, 20]. This system uses the ABBOT connectionist speech recognizer [22], which was optimized specifically for use on BBC news broadcasts. ABBOT uses an acoustic model based on connectionist networks, and was used with a 64,000 word pronunciation dictionary, resulting in a low out-of-vocabulary rate, typically about 1 to 3%. The tri-gram language model used was tailored specifically to broadcast news. 50 hours of broadcast material was transcribed, and this was supplemented with 130 million words of newspaper and newswire text, so as to achieve a large enough corpus for building the language model.

The speech recognizer does not mark punctuation in the transcripts it produces, and all words are output in lower case. However, it will interpret short pauses as sentence breaks, and mark them as <*s*> in the transcript. Longer silences are marked as <*SIL*>. Application of the speech recognizer results in an average word error rate of 29.2%, but this varied from 24.1% to 37.7% depending on the TV or radio program concerned. More significantly, recognition accuracy varied greatly from one part of a news broadcast to another, with newsreader speech generally having a much lower error rate than out of studio recordings, in which case word error rates can exceed 90%. Figure 2 shows a fragment of the recognizer's output, demonstrating that while the transcription is generally intelligible, its quality is poor, and it

[2] See http://www.blinkx.com and http://video.google.com

would not form an acceptable basis for textual annotations intended to be read by users of the system.

> <s> thousands of local people have been protesting at the way the authorities handle the operation <SIL> can marshal reports from the coastal village of mitch a <SIL> crash patches of oil has started to perk up and dalglish encased <SIL> are the main body of the thick blue is several miles offshore <s> dozens of volunteers working on a beach in which at <SIL> having to use a blade to carve up the thick

Figure 2. Example of the Speech Recognizer's Output. (The story, from BBC Radio 4 news, reports an oil spill.)

3. Topical Segmentation

When a user wishes to search or browse an archive of news stories, he or she will typically be interested in stories about a particular topic. However, news broadcasts typically contain a succession of unrelated stories, so it would not generally be useful to produce any summary or description for the broadcast as a whole. Instead what is needed is meta-data for each of the constituent stories. However, before such data can be created, it is first necessary to determine the boundaries between individual stories in the broadcast. This is a far from trivial task, because there is no reliable overt marker of such boundaries in the broadcast material.

There is a considerable literature concerning methods for segmenting both textual documents and audio and video material by topic. When working with audio and video material there is the potential to exploit audio cues, and with video material it is possible to use visual cues, such as cuts from studio to location shots, to help in segmentation. However, with all these kinds of media, it is also possible to segment based simply on the language that the media contains. Most approaches to topical segmentation of media have used textual cues, but often this has been in conjunction with cues from non-textual sources.

Chaisorn et al [4] describe a system that segments television news using a wide range of cues, including analysis of the television picture, and the captions that appear on it. These supplement segmentation that is based on the analysis of a transcription produced using ASR (automatic speech recognition). Visual features (such the color histogram of the screen, and the number of faces showing on it) and audio features (such as the presence of background noise or music) were used to train a hidden Markov model (HMM). Addition of these features improved precision for the segmentation task to 0.802, from the 0.584 figure achieved using only the ASR text, while recall improved from 0.488 to 0.749[3]. This clearly shows that addition of audio and visual features can greatly improve segmentation performance, but there are also drawbacks to this approach. Firstly visual features are only available for television news, not radio news, so this restricts the range of broadcast output to which this technique could be applied. Furthermore, Chaisorn et al's system was trained on news broadcasts which were in the same format as those it was tested on, so it would be unclear how well it would adapt to news in other formats, and retraining the system would necessitate the potentially costly task of producing more training data. For these reasons, it seems that, if acceptable results can be achieved by a system that does not use visual cues, and which does not need training, then it is preferable to use such a system.

There are several different ways of segmenting documents based only on the text that they contain, all of which can be applied to ASR generated transcriptions. The first kind of methods segment based on text that is indicative of boundaries between segments. Such a system could be implemented simply by determining which words or phrases tend to occur at the boundaries of topical segments, and searching for them in the transcripts. There are some such key phrases in the transcripts of the BBC programs, such as 'and [name of reporter] thank you' or 'back to the studio', but, partly as a result of recognition errors, such cues do not occur reliably, suggesting that such an approach would not produce good results.

Most published work seems to have taken a somewhat more sophisticated approach to detecting boundaries. Franz et al [10] trained a decision tree model which looked at the distribution of nouns on both sides of candidate boundaries, and tried to find words and bigrams that were indicative of segment boundaries, evaluating the utility of each cue using a mutual information criterion. Mulbregt et al [16] describe a system that used HMMs to detect boundaries, in which the hidden states of the HMM represented topics, and the observations with respect to the HMMs were words or sentences. This system was trained on a 15 million word corpus, in which the topic boundaries were marked. This is problematic in the case of BBC news, because no such corpus of BBC news programs was available for training. Kehagias et al [14] used product partition models to achieve text segmentation. Their system was also trained on a corpus in which topic boundaries were marked, so that the parameters of the model could be set.

A somewhat different approach, that does not rely on boundary cues, was taken by Kan et al [13], who used the concept of lexical chains to locate boundaries between topics. Where there were repeated occurrences of a noun-phrase close together in a document, a chain would be made between all the occurrences. Points in the document that were linked by many chains would be likely to be about the same topic, but those points linked by few chains would be likely to be on opposite sides of a topic boundary. Therefore topic boundaries would be placed at the locations that were linked by the fewest chains. This technique has a big advantage over the techniques described above, in that it does not require training data, meaning that it can be deployed even when no training data is available.

However, most approaches to topical segmentation use measures of lexical cohesion to determine which sections of the text are about the same topic. Such techniques require predefined input segments, which would typically be sentences or paragraphs. Stop words (the most common words in the language) are then removed from the input segments, and the remaining words are usually stemmed (any affixes are removed). A comparison is then made of the extent to which neighboring segments contain the same stems. Where there is a high degree of overlap between neighboring segments, it is unlikely that they will be about different topics, but when there is little similarity they probably should be placed in different segments. Such methods segment based on an analysis of the text as a whole, and so should be relatively robust even when words at topic boundaries have been misrecognised. It was therefore decided to proceed using such an

[3] *Precision* is the proportion of results found that were correct, while *recall* is the proportion of the potential results that were detected. In this case they refer to the proportion of story boundaries found that were correct, and the proportion of the story boundaries in the broadcast that was detected.

approach, and the specific segmentation algorithm used was the C99 segmenter [5].

Kehagias et al [14] report that C99's performance was not greatly below that of their own segmenter, which relied on training data, and which they claimed achieved the highest performance of any segmenter reported in the literature. (C99's performance on the test corpus used by Kehagias et al was 13.0% in terms of Beeferman's P_k metric [1], compared to 5.38% for their own algorithm. Lower Beeferman scores indicate higher performance.) Therefore there seems to be little justification for using a system that requires training, when comparable results can be achieved without the need for training data.

A problem in employing C99 was that it requires input segments, and will only place topic boundaries in between such segments, not within them. In the case of written material, this is generally unproblematic, as each sentence or paragraph would normally be about a single topic. However, this is more problematic in the case of audio or video material, as the transcripts produced using ASR could not be expected to reliably mark sentence boundaries. This problem was avoided simply by making input segments from sequences of words between silences marked in the ASR transcripts. As C99 does not work well with very short input segments, segments less than twenty words long would be grouped with the following segment.

C99 calculates the similarity between input segments using the cosine measure (see for example Jurafsky and Martin [12]). This measure gives high scores to segments that contain mainly the same words, and in which those words occur with similar frequencies, and lower scores to documents that contain different words, or in which the words occur with different frequencies. However, the absolute values of this measure are not reliable when assessing the similarity of short segments, and so C99 looks instead at a ranking of cosine scores for similarities between each pair of segments in each part of the input text. Initially, all of the input segments are grouped together in one big super-segment, and segmentation then proceeds by breaking this initial segment into successively smaller super-segments. Each time a super-segment is divided, an attempt is made to divide it so that the break is made between the most dissimilar neighboring segments it contains. C99 can decide when the optimal segmentation has been achieved, and so there is no need to specify how many topical segments should be created.

C99 has been found to work well on the BBC news programs, though it often fails to create separate topical segments for very short stories (which are often covered in one or two sentences). Headlines also create a problem, as C99 will often break these into topical segments in a fairly arbitrary manner, usually resulting in several stories appearing in each topical segment. We will see below that the document matcher can compensate for such errors.

4. Key-phrase Extraction

Once the segmenter has segmented the ASR transcript so that we have a section of text for each story in the original broadcast, the next stage is to try to find key words or phrases that are representative of the story. These key phrases can then be used as the basis of a search on the BBC website for web pages that report the same story. Therefore the aim of the key phrase extractor component is simply to extract several phrases that are likely to occur on relevant web pages, and which are unlikely to appear on unrelated web-pages. Whether or not these phrases are coherent as far as human readers are concerned is irrelevant.

There is a significant literature on the subject of key-phrase extraction, and the closely related topic of title generation. Jin and Hauptmann [11] describe an algorithm that automatically generates titles for transcribed broadcast news (the titles could be seen as a kind of key phrase), and Turney [24] describes a system that aims to extract key-phrases for use in indexing or summarizing documents. However, both these systems have a major, drawback, in that they require training on large collections of documents on which key phrases or titles have already been marked. No such collection was available for the BBC data, and, in general, NLP systems do not perform well if the data on which they are trained is not similar in topic and structure to the data on which they are applied. Therefore it would seem that these systems cannot be used for extracting key-phrases from the BBC news broadcasts, as no suitable training data set is available.

Both Jin and Hauptman and Turney's systems used *term frequency inverse document frequency* (TF.IDF) as a central part of their mechanism for selecting key-phrases. This method looks for phrases that occur more frequently in the text under consideration than they do in the language as a whole. This is likely to find phrases that are characteristic of the text, while ignoring phrases that occur frequently in the text simply because they are common in the language as a whole. It requires training data in order to determine how common each phrase is, but this training data need not be marked up with any annotations, and so the ASR transcripts of the broadcasts could themselves be used as the training data. TF.IDF is really a family of methods, because there are several different formulas that can be used to calculate TF.IDF scores for each phrase, and various criteria for deciding what constitute candidate phrases. The chosen method was the same as that used for KEA, and reported by Frank et al [9].

Firstly, any sequence of words up to length six was considered to be a 'phrase', except that phrases that began or ended on stop words were ignored. The transcripts of 13,353 news broadcasts were used for collecting phrase frequency data. Each word was stemmed, and how many times each phrase occurred in the training data was determined. However, because the number of phrases up to length six occurring in the training data was so large, once more than 300,000 distinct phrases had been observed, those with the lowest frequencies were removed until there were less than 100,000 remaining. This process was repeated every time the number of phrases stored exceeded 300,000.

Key phrases were extracted for each topical segment. Firstly the frequency of each stemmed phrase in the topical segment was found, and if it occurred two or more times, its TF.IDF score was calculated using equation (1), in which N is equal to the number of transcripts in the training data, n is the number of documents in which the phrase occurs, t is the frequency of the phrase in the topical segment, and p is the number of candidate phrases in the current document. (n would be zero for phrases not recorded in the phrase frequency data.)

$$\text{(1) TF.IDF Score} = \log\left(\frac{N}{n+1}\right) \times \frac{t}{p}$$

The four phrases with the highest TF.IDF scores were then taken to be key-phrases for the topical segment. (Sometimes fewer than four key-phrases would be extracted, as fewer than four phrases would occur at least twice in the segment.) This technique was usually successful in finding appropriate key phrases, although often there would be fragments that could not really be described as phrases, such as 'minister nick' instead of 'local government minister nick raynsford', or inappropriate phrases were returned, sometimes including mis-recognized words, or words that would not help to identify the topic of the story. However, we will see below that such errors do not seriously detract from the accuracy of the meta-data produced as the final product of the annotation system.

5. Search of the Web for Related Documents

The purpose of extracting key-phrases was so they could be used to search for web pages reporting the same story on the BBC website. Searches were conducted using Google, which was accessed via the Google Web API[4]. Searches were restricted to the news section of the BBC website, by adding the term *site:news.bbc.co.uk* to each search. An attempt was made to restrict the search only to the day of the original broadcast, or the day before, by adding a term specifying either of these dates in the format that they appear on the BBC website, for example *"1 December, 2004" OR "30 November, 2004"*. The inclusion of the previous day as well as the date of the broadcast was necessary because sometimes news stories are broadcast the day after they appear on the website. The dates of the broadcasts were known, as they were always input to the speech recognition component of the system. This technique was usually successful in restricting the dates of the web pages returned, but sometimes the search would also return web pages containing references to events that happened on those dates, which were sometimes published years later.

Besides the site and date terms, key phrases were also added to the queries, each enclosed in quotation marks, so that Google would search for the phrase as a whole, and not just its component words. (2) gives one example of a complete search term, which concerned a story about UK government preparations for a possible terrorist attack involving smallpox. Up to five searches were performed for each topical segment. The first would include the two key phrases with the highest TF.IDF scores, while the other four would each search with just one of the four key-phrases extracted. For each search, the first three URLs returned by Google were retrieved. However, there might be less than 15 URLs returned in total, because sometimes fewer than four key phrases would be found for a segment, or Google would return fewer than three results for a search. Also, often two or more searches would return the same URL.

(2) site:news.bbc.co.uk "3 December, 2002" OR "2 December, 2002" "smallpox " "vaccination "

Google normally limits the use of their API service to 1000 searches per day. This is not sufficient to allow the annotation of all of the BBC's output, as several searches are performed for each news story. Full scale deployment of the system will therefore be dependent on negotiating a higher search limit with Google, or on adapting the system to use an alternative search engine. (The search facility of the BBC website itself is one possible candidate.)

Examination of the results showed that the first URL returned often pointed to the web page that most closely matched the story, and when it did not, then often the second or third URL returned did. In the cases in which the first search, using two key-phrases, did not return a correct URL, then often one of the other searches would. In the cases where no appropriate URL was returned for a story, this was most usually because the segment contained two separate stories, or corresponded to a part of the news broadcast containing headlines, in which case no single web page would be appropriate. However, even when a correct URL was returned, a procedure was still needed for determining which of the URLs it was. This was achieved by the addition of a document matching component.

The document matching component loads the documents found by Google, starting with those found using the first two key-phrases, and then subsequently those found using the first, second, third and finally the fourth key-phrase. The text of each web page was then compared to that of the input segment, and if they were sufficiently similar then the web page would be associated with the topical segment, and no more web pages would be considered for that segment.

The method for determining the degree of similarity between web-pages and topical segments is based on the technique that C99 (see section 3) uses to determine the similarity between different parts of a document. Where they exist, annotations on the BBC web page are used to find the content text of the web page, excluding all the titles and links that typically make up about half the text. Stop words are then removed from both the web pages and topical segments, and both are stemmed. The frequency of each word in each document is then determined so that a cosine similarity score (see section 3) can be obtained for their similarity.

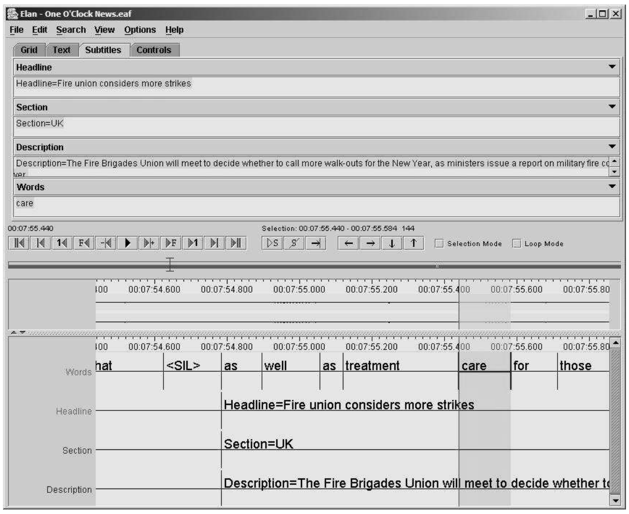
By comparing similarity scores for topical segments and matching web pages, and scores for topical segments and non-matching web pages, it was found that when the similarity score was greater than 0.32 the topical segment and the web page almost always matched, and when it was less than 0.32 they almost always did not match. When scores close to 0.32 were obtained this was usually because part, but not all, of the segment matched the web page, or else the web page described a related story, but could not actually be said to be reporting the same event as that reported in the broadcast. So we can see that the document matcher increases the precision of the annotation process, by preventing unrelated web pages from being associated with topical segments, and by preventing sections of broadcasts such as headlines (for which there are no matching web pages) from being associated with any web page at all. It can also improve the recall of the system, because it enables web pages other than the first one returned by Google to be considered. A more formal evaluation of the performance of the annotation system, in terms of what proportion of news stories are ultimately associated with a web page describing the same story, is presented in section 10.

6. Manual Annotation

We are now at the stage of having an ASR transcript for each news program that is divided into sections, each corresponding to a news story, and some of which contain links to web pages describing the same story. Most news stories on the BBC website also contain *meta* HTML tags, defining a headline for the story, a short summary of the story, and a classification of the story in terms of what type of news the story reported (example categories are health, Africa, politics and London). Where these existed they were extracted from the web pages, and added as annotations to the transcript, with the same start and end points as the corresponding topical segment.

[4] See http://www.google.com/apis/

At this point, it is possible to output the transcript in a format that is readable by the ELAN linguistic annotator [3]. This allows the transcript to be loaded into ELAN, where the timing information it contains allows each word of the transcript, and the annotations on it, to be associated with the appropriate part of the video or audio file, as shown in Figure 3. The segmentation and annotations can then be manually corrected, and any missing annotations added, before the annotation proceeds any further. Such manual intervention will be necessary if accurate annotations are to be produced for every story, but when a lower level of recall is acceptable, this step can be omitted.



Headline=Fire union considers more strikes

Section=UK

Description=The Fire Brigades Union will meet to decide whether to call more walk-outs for the New Year, as ministers issue a report on military fire co ver

care

00:07:55.440

Selection: 00:07:55.440 - 00:07:55.584 144

hat <SIL> as well as treatment care for those

Headline=Fire union considers more strikes

Section=UK

Description=The Fire Brigades Union will meet to decide whether t

Figure 3. Editing Annotations on a Radio Broadcast.

7. Creating Story Index Documents

Users of the final search and browsing system are much more likely to be interested in finding stories about a particular topic, than in finding all the stories covered in one news broadcast. Therefore, the next stage of the meta-data creation process was to produce a new GATE document for each story for which a matching URL had been found. Topical segments for which no matching URL had been found were discarded at this stage (unless annotations had been added to them during the manual annotation phase).

The content text of each new GATE document would be the same as the main content text of the matched web page, or the whole page when such text could not be identified. The *headline*, *summary* and *section* features that were mentioned in section 6 were then added to the document as features, if they were available, along with the URL of the media file containing the original broadcast, timing information giving the start and end times of the story in the broadcast, the channel and date on which the program was broadcast, and the name of the program. (The program name, channel and date must be provided when the file is input into the annotation system, while the rest of the information is derived during the annotation process.) These documents create the kind of data necessary for a detailed catalog record of each news story, and can be used to index the broadcast in search or browsing systems.

8. Semantic Annotation

Up to the present point, the meta-data we have created for news stories has been in a textual format. This could allow searches for stories whose meta-data contains particular text, which would work in much the same way as an ordinary search engine. However, it would be better if it were possible to perform more specific searches, which could make reference to specific unique entities, such as people or countries. For this purpose, the KIM knowledge and information management platform [18] was used. KIM produces meta-data for the Semantic Web in the form of annotations with respect to a basic upper-level ontology called PROTON[5] encoded in OWL. These annotations can be associated with particular words or phrases in the documents.

KIM can automatically identify entities referred to in text documents, and associate each individual entity with a URI. These URI's are organized through the use of PROTON, which is composed of three modular layers and includes about 250 classes and 100 properties. Its Top module contains categories for the most common types of entities, such as people, companies, cities, etcetera. The Upper module is more extensive and contains branches of more specific sub-categories of the basic top categories. Both the ontology and the annotation system were designed for use over English news texts, and so they could be expected to work well for BBC news, although they were designed for use over textual data, not spoken material. KIM, in common with most of the other components of Rich News, is based on the GATE natural language processing framework.

KIM identifies entities in texts using a number of techniques. Firstly, and most simply, text is looked up in gazetteers (lists of particular types of entity, such as names of cities, or days of the week). More complex approaches make use of a shallow analysis of the text, and pattern matching grammars. In normal operation, KIM combines the results of all these methods in order to produce more annotations, and more accurate annotations, than could be extracted using a single method alone. Kiryakov et al [15] report results showing that KIM achieves a meta average F1 score of 91.2% when identifying dates, people, organizations, locations, percentages, and money in a corpus made up of UK, international, and business news. This score clearly indicates high, but by no means perfect, levels of precision and recall, but it should be taken into account that the performance of such systems is very dependent on the type of data on which they are run, therefore a direct comparison of KIM to any system tested on a different corpus would not be valid.

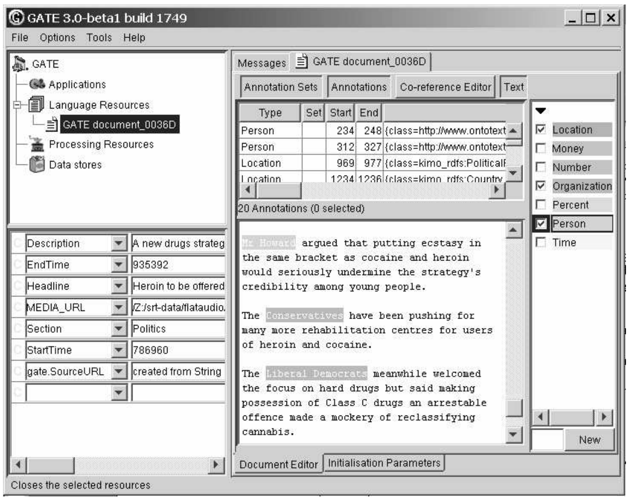
The most important difference between KIM and the majority of the other information extraction systems is that KIM can identify unique entities, and it can annotate all occurrences of the same named entity with the same URI. This will assign the entities to a position in an ontology that is a formal model of the domain of interest. Each entity will also be linked to a specific instance in the knowledge base for that entity (which will be added if it does not exist at that point), and it will also be linked to a semantic entity description. KIM will also try to determine when the same named entity appears in two different formats, so that, for example, *New York* and *N.Y.* would both be mapped to the same instance in the knowledge base. For a more detailed description of the information extraction capabilities of KIM, see Popov et al [16].

Initial attempts to use KIM with the ASR transcripts themselves produced very poor results, owing to the generally poor quality of the transcripts, and in particular to the absence of punctuation and

[5] http://proton.semanticweb.org

capitalization. Consequently, KIM failed to detect most of the entities that the transcripts contained. Therefore, in the final version of Rich News Annotator, annotation was only attempted on the text extracted from web pages. This has the advantage of achieving much better recall and precision for the annotation process, but the disadvantage is that there is a much more indirect relation to the original broadcast. If the broadcast and the web page report the same story, it is likely that they will contain many of the same named entities, but it is also likely that each one will contain some named entities that the other does not.

However, an inspection of the annotations produced by KIM and a comparison between them and the corresponding broadcast recordings reveal that many of the named entities found by KIM actually do occur in the broadcasts, and that we find more relevant named entities by annotating the web pages than we do by annotating the transcripts themselves. An example of a story index document that has been annotated by KIM is shown in Figure 4 where we can see the features associated with the document, recording details such as its headline and the media file. We can also see a part of the text of the index document, in which two organization and one person annotation have been marked.



Type	Set	Start	End	
Person		234	248	{class=http://www.ontotext
Person		312	327	{class=http://www.ontotext
Location		969	977	{class=kimo_rdfs:PoliticalF
Location		1234	1236	{class=kimo_rdfs:Country

20 Annotations (0 selected)

Description	A new drugs strateg
EndTime	935392
Headline	Heroin to be offered
MEDIA_URL	/Z:/srt-data/flataudio.
Section	Politics
StartTime	786960
gate.SourceURL	created from String

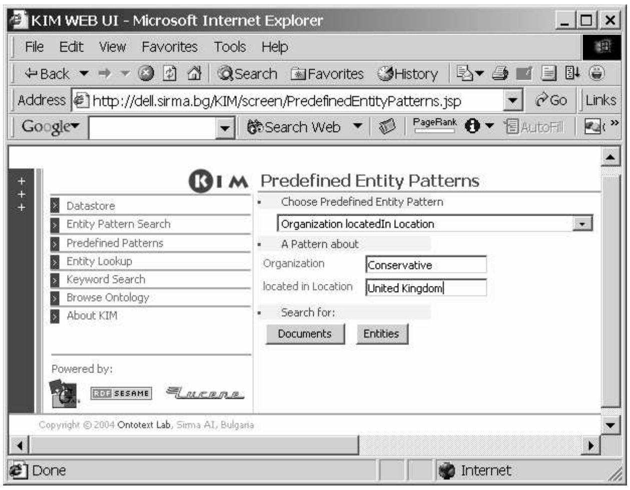
```
Mr Howard argued that putting ecstasy in
the same bracket as cocaine and heroin
would seriously undermine the strategy's
credibility among young people.

The Conservatives have been pushing for
many more rehabilitation centres for users
of heroin and cocaine.

The Liberal Democrats meanwhile welcomed
the focus on hard drugs but said making
possession of Class C drugs an arrestable
offence made a mockery of reclassifying
cannabis.
```

Figure 4. An Example of a Story Index Document that has been annotated by KIM, displayed in the GATE GUI.

9. Search and Retrieval of Broadcasts

Up to the present point, this paper has described the annotation part of Rich News. However, annotation by itself is only a means to an end. Annotations could be used in a number of ways, for example to enable browsing of the news stories. However, most often a user of the system will have a particular topic or person in mind, and they will want to undertake a search for related material.

The KIM platform has a web user interface (the KIM Web UI), that is suitable for this purpose. The KIM Web UI not only allows simple text searches, it also allows semantically enhanced searches. So, for example, if we wanted to search for a person whose last name was *Sydney*, we could specify that only entities annotated as *person*, or as some ontological sub-class of person (such as *woman*) were to be considered. This would prevent references to the city of Sydney being returned, which are far more numerous than references to people called Sydney. It would be difficult to perform this search on a conventional text-based search engine, as results returned for the city of Sydney would swamp those that referred to people called Sydney.



Address http://dell.sirma.bg/KIM/screen/PredefinedEntityPatterns.jsp

Predefined Entity Patterns

- Choose Predefined Entity Pattern

Organization locatedIn Location

- A Pattern about

Organization: Conservative

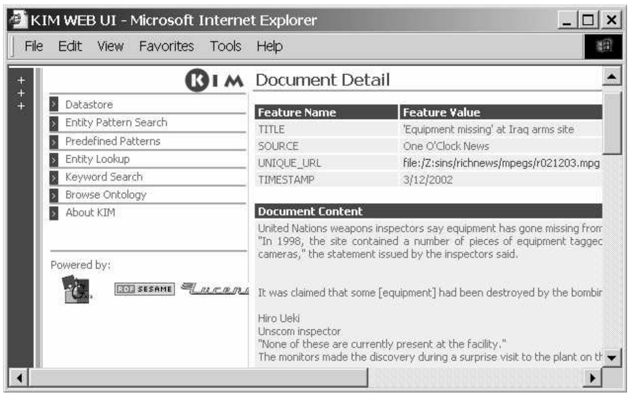
located in Location: United Kingdom

- Search for:

Copyright © 2004 Ontotext Lab, Sirma AI, Bulgaria

Figure 5. The KIM Web UI.

The KIM Web UI has several pages, of varying complexity, on which searches can be entered. One of these pages, which allows searches for organizations in particular locations, is shown in Figure 5. This shows a search for organizations located in the United Kingdom whose name is *Conservative.* (The most complex search page allows for almost total flexibility in specifying any combination of entities of different types.) Clicking on the documents button will then retrieve a list of headlines for stories that are believed to contain references to entities fulfilling these search criteria. (That is, those stories for which the story index documents contain annotations that appear to fulfill the search criteria.) Clicking on one of the headlines will bring up the full details of the corresponding story, as shown in Figure 6. These documents include links to the media files containing the original broadcasts, and clicking on these links will open a media player that will play the audio or video. This shows how the KIM Web UI allows convenient and sophisticated search and retrieval of both story meta-data and the original recordings of broadcasts.



Document Detail

Feature Name	Feature Value
TITLE	'Equipment missing' at Iraq arms site
SOURCE	One O'Clock News
UNIQUE_URL	file:/Z:sins/richnews/mpegs/r021203.mpg
TIMESTAMP	3/12/2002

Document Content

United Nations weapons inspectors say equipment has gone missing from
"In 1998, the site contained a number of pieces of equipment taggec
cameras," the statement issued by the inspectors said.

It was claimed that some [equipment] had been destroyed by the bombir

Hiro Ueki
Unscom inspector
"None of these are currently present at the facility."
The monitors made the discovery during a surprise visit to the plant on th

Figure 6. A Story Found by the KIM Web UI.

10. Evaluation of Rich News Annotator

The performance of Rich News Annotator, and of the indexing and search systems that rely on it, is dependent on how successfully the annotator produces index documents. This is in turn dependent on how successful it is in finding web pages for the stories in the broadcasts. The evaluation of the system was therefore based on a measure of for what proportion of news stories in broadcasts Rich News was able to produce appropriate index documents automatically.

Three of the key components in the annotation process, topical segmentation, key-phrase extraction, and web search, are all particularly prone to error, and therefore unreliable. However, even if one of these components fails, a later component may be able to correct or compensate for the error, so the performance of the system as a whole could be better than that of any of its individual components. For example, the topical segmenter might misidentify story boundaries, and so put two stories together into one segment. Some of the key-phrases extracted by the key-phrase extractor would then be likely to be for one story, and some for the other. However, web pages returned by the web-search could only refer to one story or the other, so whichever (if any) of these was found to match sufficiently closely by the document matcher, would be used as the basis of an index document. Clearly, the start or end time recorded on this document would in this case be incorrect, as one of these would extend into the preceding or following story, but the index document would still allow access to the story, and give approximately correct timing information about which part of the media file contains the story. In most cases the inaccuracy of the start or end time would likely be more of a minor annoyance than a factor that would cause retrieval of the media to fail.

The evaluation of the system's performance was conducted by first playing nine broadcasts, and noting the stories that occurred in each. The programs used in the evaluation were BBC Radio 4's *The World at One*, which is a half hour long daily national news program. The broadcasts concerned were all from the last six months of 2002. For the purposes of the evaluation, simple announcements of financial figures (such as stock market indices and interest rates) were not counted as stories, and sports reports were also ignored, as the system was not designed to work with sports.

Once each story appearing in each broadcast had been noted, Rich News Annotator was run on each of the broadcasts, and story index documents were produced. For each index document, it was determined whether it reported a story covered in the corresponding radio broadcast, whether it reported a closely related story, but could not be said to be reporting a story in the broadcast, or whether it reported an unrelated story. An example of a story that was considered closely related, but not correct, was when the radio program reported a politician's comments about the trial of a royal servant, but the web page found reported only the trial itself.

Table 1. Results of the Evaluation on 66 News Stories.

	Correct	Incorrect	Precision (%)	Recall (%)	F1
Strict	25	2	92.6	37.9	53.8
Lenient	27	0	100	40.1	57.2

Table 1 shows the results under two conditions. In the first condition, *strict*, annotation was only considered successful if the correct story was matched, but in the second, *lenient*, it was considered correct if a closely related story was matched. The nine broadcasts considered (making a total of approximately four and a half hours of material), contained a total of 66 news stories. The results of the evaluation show that the system achieved very high precision, but that recall was much lower. This gives a moderate F1 score (a measure that trades off precision and recall, to provide an overall measure of a system's performance). However, this clearly demonstrates that Rich News Annotator, running in its fully automatic mode, can give access to a large volume of material that would be inaccessible if no annotation was provided, which is the case with much of the BBC's output at present. In future work we plan to exploit the redundancy available in multiple news websites in order to improve recall.

The stories that were missed by the annotator were often those that consisted of only one or two sentences, rather than those that were reported in more depth. It would seem likely that users of the system would typically be less interested in retrieving such short stories than those reported at more length. Therefore, the performance of the final system is probably better than is suggested by the recall scores. Furthermore, this evaluation demonstrates that the annotation system is very reliable, and therefore that searches performed using the search system would rarely return references to irrelevant media.

11. Future Developments

While Rich News is a complete, fully functional system at present, it will remain under development for some time. While the individual components it contains all function well enough for acceptable results to be produced, all of them could clearly be improved. The biggest single improvement in the overall system would result if the quality of the speech recognition could be improved. However, the speech recognition component used is already state of the art, and it is not clear that significantly better speech recognition systems will be available in the near future, so it is more likely that improvements will have to be made in other areas.

One solution to the problem of poor quality speech recognition would be to use teletext subtitles (closed captions) to provide an approximate transcription, instead of relying on speech recognition, an approach that is already in use by Google's video search. This could be expected to produce a somewhat higher quality transcript, although live to air subtitles tend to contain quite a large number of errors, as they are themselves produced by automatic speech recognition, using a system trained to the voice of a subtitler, who repeats what he or she hears in the broadcast. Our intention is to adapt our system so that such data can be used where it is available, but at present subtitles are not available for all BBC news broadcasts, and this will never be an option for Radio broadcasts. The BBC would also like to improve access to their extensive back-catalogue of un-annotated news, and no subtitling data is available for most of that material.

One technology that could potentially improve the performance of several Rich News components, namely the topical segmenter, the key phrase extractor, and the document matching component, is *latent semantic analysis* (LSA). Latent semantic analysis is a technique in which large collections of documents are analyzed to see which words co-occur in the same documents more often than would be expected given their frequencies. Words that tend to co-occur are likely to be semantically related. If semantically related words could be detected, it would help the segmentation algorithm, because similarity between document segments could be calculated not only based on the presence of the same word stems in each document segment, but also on the presence of semantically related words stems. LSA versions of the C99 segmenter have been reported in the literature [6, 2], so adding an LSA capability to the version of C99 used in the present system should be unproblematic. As many of the failures of the present system result from segmentation errors, any such improvement to the segmentation component could be expected to make a significant improvement to the performance of the system as a whole.

LSA could also help to improve the key-phrase extractor, because semantically related phrases could be counted together, increasing the count for phrases with a particular meaning. At present roughly the same idea is often expressed with several different words, none of which might occur frequently enough to be considered to be a key-phrase on its own. However, if counts for all these words were totaled, these words would more likely be chosen as key-phrases. As these words have a meaning frequently expressed in the segment, it should be a good key-phrase, and so this use of LSA would be likely to improve the performance of the system. LSA could also be used in the document matching component, where it would fulfill the same role as in the topical segmenter, that is providing a more accurate measure of which document segments are related. However, as the document matching component already performs to a very high level of accuracy, addition of LSA functionality to it could not be expected to greatly improve performance.

A different kind of improvement to the system would be to use a more sophisticated approach to searching for related web pages. At present the system searches first with the first two key phrases, and then with each of the other four individually, and only three URLs are returned. However experimenting with variations on these parameters, for example extracting more key phrases, returning more URLs, or searching with other pairs of search terms might yield better results. News is also available on the web at multiple related sites (for example the BBC, newspapers, and CNN all have news websites). In previous work on annotating sports video [23] we found that merging redundant results from multiple sources improves performance. Future work will apply this approach in Rich News.

Changes could also be made to the document matcher. At present this stops loading web pages when it finds one that is a sufficiently close match to the ASR transcript, but it might be better if it matched all web pages, and chose that which matched most closely, rather than just choosing the first good match found. The drawback of that approach would be that the document matching is a time consuming process, as each web page has to be downloaded, which is why at present Rich News takes the first matching web page found.

There is also the potential to use the matched web pages in much more sophisticated ways. At present the system simply extracts the main body text, or the whole text in the event that main body text cannot be identified. However, Rich News could be extended so that it can determine exactly which parts of a web page are relevant, and it could align particular sentences or paragraphs of the web pages with the corresponding parts of the broadcast recording, rather than just aligning the whole text with the entire story, as happens at present. Another approach would be to perform a more sophisticated linguistic analysis on the web pages, so that the text that they contain could be transformed to produce text that better describes the web pages. This might involve a semantic analysis of individual sentences, both in the web pages and the transcript, which would allow a new document to be generated based on those parts of the web page that match the transcripts semantically.

12. Conclusion

Rich News addresses the task of how to produce high quality, semantic meta-data for broadcast news programs. The key challenge in this area has been how to obtain a description of what the broadcasts contain given only an audio or video recording. Rich News shows how this problem can be overcome, by using speech recognition to produce a low quality transcript, and then using resources found on the web as the basis of high quality annotations. It can produce semantic annotations that can form a part of the Semantic Web, and which enable sophisticated ontology aided search of the meta-data, and retrieval of the original broadcast material. The present system is specialized for English language news, but it could be adapted to work on other genres, and even other languages. The main requirements are the availability of a suitable textual source that mirrors the content of the broadcasts, and language processing tools for the language in question. Systems such as Rich News will allow effective access to much material that up to now has effectively been lost after its initial broadcast, due to the lack of adequate annotation resources.

13. ACKNOWLEDGMENTS

The research for this paper was conducted as part of the European Union Sixth Framework Program projects PrestoSpace (FP6-507336) and SEKT (EU IST IP 2003-506826). We would like to thank four anonymous reviewers for their helpful comments, and the BBC archives for providing information about their annotation process, and for making broadcast material available to us.

14. REFERENCES

[1] Beeferman, D., Berger, A. and Lafferty, J. Statistical models for text segmentation. Machine Learning, Volume 34 (1999), 177-210.

[2] Brants, T., Chen, F. and Tsochantaridis, I. Topic-based document segmentation with probabilistic latent semantic analysis. In Proceedings of CIKM (McLean, VA, USA, November 2002), 211-218.

[3] Brugman, H. and Russel, A. Annotating multi-media / multi-modal resources with ELAN. In proceedings of LREC (Lisbon, Portugal, May, 2004), 2065-2068.

[4] Chaisorn, L., Chua, T., Koh, C., Zhao, Y., Xu, H., Feng, H. and Tian, Q. A Two-Level Multi-Modal Approach for Story Segmentation of Large News Video Corpus. Presented at TRECVID Conference, (Gaithersburg, Washington D.C, November 2003). Published on-line at http://www-nlpir.nist.gov/projects/tvpubs/tv.pubs.org.html.

[5] Choi, F. Y. Y., Advances in domain independent linear text segmentation. In Proceedings of NAACL, (Seattle, USA, April, 2000), 26-33.

[6] Choi, F. Y. Y., Wiemer-Hastings, P. and Moore, J. Latent semantic analysis for text segmentation. In Proceedings of EMNLP (Pittsburgh, USA, June 2001), 109-117.

[7] Cunningham, H., Maynard, D., Bontcheva, K., and Tablan, V. GATE: A framework and graphical development

environment for robust NLP tools and applications. In proceedings of ACL (Philadelphia, USA, July 2002).

[8] Dimitrova, N., Zimmerman, J., Janevski, A., Agnihotri, L., Haas, N., Li, D., Bolle, R., Velipasalar, S., McGee, T. and Nikolovska, L. Media personalisation and augmentation through multimedia processing and information extraction. In L. Ardissono and A. Kobsa and M. Maybury (Eds.), Personalized Digital Television, 201-233, Kluwer Academic Publishers, Dordrecht, Netherlands, 2004.

[9] Frank, E., Paynter, G. W., Witten, I. H., Gutwin, C. and Nevill-Manning, C. G. Domain-specific keyphrase extraction. In Proceedings of IJCAI, (Stockholm, Sweden, July-August, 1999), 668-673.

[10] Franz, M., Ramabhadran, B., Ward, T. and Picheny, M. Automated transcription and topic segmentation of large spoken archives. In Proceedings of Eurospeech (Geneva, Switzerland, September 2003), 953-956.

[11] Jin, R. and Hauptmann, A. G. A new probabilistic model for title generation. In proceedings of COLING (Taipei, Taiwan, August, 2002).

[12] Jurafsky, D. and Martin, J. H. Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition. Prentice Hall, Upper Saddle River, NJ, 2000.

[13] Kan, M., Klavans, J. L., and McKeown, K. R. Linear segmentation and segment significance. In Proceedings of the 6th International Workshop on Very Large Corpora (Montreal, Canada, August, 1998), 197-205.

[14] Kehagias, A., Nicolaou, A., Petridis, V. and Fragkou, P. Text Segmentation by Product Partition Models and Dynamic Programming. Mathematical and Computer Modelling, 39, Issues 2-3, (January 2004), 209-217.

[15] Kiryakov, A., Popov, B., Terziev, I., Manov, D. and Ognyanoff, D. Semantic annotation, indexing, and retrieval. Journal of Web Semantics, 2, Issue 1, (2005).

[16] Mulbregt, P. V., Carp, I., Gillick, L., Lowe, S. and Yamron, J., Text segmentation and topic tracking on broadcast news via a hidden Markov model approach. The 5th international conference on spoken language processing (Sydney, Australia, November 1998). Published on-line at http://www.shlrc.mq.edu.au/proceedings/icslp98/WELCOME.HTM.

[17] Popov, B., Kiryakov, A., Ognyanoff, D., Manov, D. and Kirilov, A. KIM – a semantic annotation platform for information extraction and retrieval. Natural Language Engineering, 10, Issues 3-4, (September 2004), 375-392.

[18] Popov, B., Kiryakov, A., Ognyanoff, D., Manov, D., Kirilov, Al, and Goranov, M. Towards semantic web information extraction. In proceedings of ISWC (Sundial Resort, Florida, USA, October, 2003).

[19] Przybocki, M., Fiscus, J., Garofolo, J. and Pallett, D. 1998 HUB-4 information extraction evaluation. In Proceedings of the DARPA Broadcast News Workshop (Herndon, VA, February, 1999), 13-18.

[20] Renals, S., Abberley, D., Kirby, D. and Robinson, T. Indexing and Retrieval of Broadcast News. Speech Communication, 32, Issues 1-2 (September 2000), 5-20.

[21] Robinson, T., Abberley, D., Kirby, D. and Renals, S. Recognition, indexing and retrieval of British broadcast news with the THISL system. In Proceedings of Eurospeech, (Budapest, Hungary, September 1999), 1067-1070.

[22] Robinson, T., Hochberg, M. and Renals, S. The use of recurrent networks in continuous speech recognition. In C. H. Lee, K. K. Paliwal and F. K. Soong (Eds.), Automatic speech and speaker recognition – advanced topics, 233-258, Kluwer Academic Publishers, Boston, 1996.

[23] Saggion, H., Cunningham, H., Bontcheva, K., Maynard, D., Hamza, O. and Wilks, Y. Multimedia indexing through multisource and multilingual information extraction; the MUMIS project. Data and Knowledge Engineering, 48, (2003), 247-264.

[24] Turney, P. D. Coherent keyphrase extraction via web mining. In Proceedings of IJCAI (Acapulco, Mexico, August, 2002), 434-439.