SEARCHING AND SUMMARIZING IN A MULTILINGUAL ENVIRONMENT

MICHAL TOMAN[1]; JOSEF STEINBERGER[1]; KAREL JEZEK[1]

[1] Faculty of Applied Sciences, University of West Bohemia,
Univerzitni 22, Plzen, Czech Republic
e-mail: mtoman@kiv.zcu.cz; jstein@kiv.zcu.cz; jezek_ka@kiv.zcu.cz

Multilingual aspects have been gaining more and more attention in recent years. This trend has been accentuated by the global integration of European states and the vanishing cultural and social boundaries. The ever increasing use of foreign languages is due to the information boom caused by the emergence of easy internet access. Multilingual text processing has become an important field bringing a lot of new and interesting problems. Their possible solutions are proposed in this paper. Its first part is devoted to methods for multilingual searching, the second part deals with the summarization of retrieved texts. We tested several novel processing techniques: a language-independent storage format, semantic-based indexing, query expansion or text summarization leading to faster and easier retrieval and understanding of documents. We implemented a prototype system named MUSE (Multilingual Searching and Extraction) and compared its qualities with the state-of-the-art search engine – Google. The results seem to be promising; MUSE shows high correlation with the market-leading products. Although for our experiments we used Czech and English articles, the main principle applies to other languages as well.

Keywords: multilingual text processing; searching; summarization; EuroWordNet

INTRODUCTION

There are over 3M Internet users in the Czech Republic. Most of them search not only Czech pages but also English, Slovak, German, and others as well. In addition, another 1.1M Americans of Czech origin access the Internet in Czech. The situation is similar in other countries [1]. Therefore, multilingual aspects are increasing in importance in text processing systems. We are proposing possible solutions to new problems arising from these aspects. We suppose that a multilingual system will be useful in digital libraries, as well as the web environment.

Our contribution deals with methods of multilingual searching enriched by the summarization of retrieved texts. This is helpful for a better and faster user navigation in retrieved results. We also present our system, MUSE (Multilingual Search and Extraction). The EuroWordNet thesaurus (EWN) [2] is the core of our multilingual searching approach, and the heart of our summarizer is the Latent Semantic Analysis (LSA) [3].

MUSE consists of several relatively self contained modules. Some of them, namely language recognition, lemmatization, word sense disambiguation and indexing, were described in [4]. In this paper, we mainly present a description of multilingual searching and user query expansion. These features are possible due to EWN that is also used in lemmatization, indexing and a query conversion into the language independent form. The internal format enables the creation of queries in various EWN languages. The summarization module can be used to deliver short summaries instead of full texts. The main search engine is based on the modified vector retrieval model with the TF-IDF scoring algorithm (see section Searching). It uses an SQL database as an underlying level to store indexed text documents, EWN relations and lemmatization dictionaries for each language. Queries are entered in one of the languages (currently Czech and English). However, it should be noted that the principles remain the same for an arbitrary number of languages. Methods based on the frequency of specific characters and words are used for language recognition. All terms are lemmatized and converted into the internal EWN format – Inter Lingual Index (ILI). The lemmatization module executes mapping of document words to their basic forms, which are generated by the ISPELL utility package [5]. The module complexity depends on the specific language. Our language selection includes both morphologically simple (English) and complicated (Czech) languages. Therefore, the Czech language requires a morphological analysis [6].

Optionally, the query can be expanded to obtain a broader set of results. EWN relations between *synsets* (sets of synonymous words) are used for query expansion. Hypernym, holonym, or other related synsets can enhance the query. The expansion setting is determined by user's needs.

The amount of information retrieved by the search engine can be reduced to enable the user to handle this information more effectively. We have developed an extractive summarizer, based on latent semantic analysis, with variable dimensionality reduction [7]. Its idea is to reduce the document term space to an automatically determined number of document topics. Lately, we enriched the latent semantic structure of a document by anaphoric relations [8], which resulted in a significantly better performance than the performance of a system not using the anaphoric information. The summarizer is very well comparable with other state-of-the-art systems [9]. MUSE uses the summarizer for presenting summaries of retrieved documents. Moreover, we study the possibility of speeding up document retrieval by searching in summaries, instead of in full texts.

MUSE was evaluated by means of a multilingual document corpus, and promising results were obtained. The corpus consists of English texts (Reuters Corpus Volume 1) and Czech texts (Czech Press Agency). The aim of the experiments was firstly, to verify the impact of multilingualism on the quality of searching; secondly, to test the use of the multilingual thesaurus in query expansion; and finally, to measure the precision and speed-up while using summarized documents for searching.

MUSE ARCHITECTURE

To verify our solution, we created a prototype system. It demonstrates possibilities, advantages, and disadvantages of the approach. MUSE was designed as a modular system, and it consists of relatively independent parts. The overall description is shown in figure 1. The system contains five logical parts: preprocessing, lemmatization, indexing, a summarizer, and searching.

It is necessary to acquire a high quality lemmatization dictionary for indexing and successive processing. This task is covered by the preprocessing module. It processes the word forms derived from ISPELL, and creates a lemmatization dictionary for each language. A morphological analyzer, which improves lemmatization precision, is applied to the Czech language. Basic word forms are mapped on EWN synsets, and the resulting dictionary is used in the indexing module for document transformation into the language independent form. The summarization module can be considered a breakthrough part of the system. It transforms full documents into shorter ones with a minimal information loss. It is very important for an easier user's navigation in a larger number of documents. This module is based on the LSA method. The main part of MUSE is the searching module enriched by query expansion. Terms can be expanded in different ways (e.g. hypernyms, hyponyms).

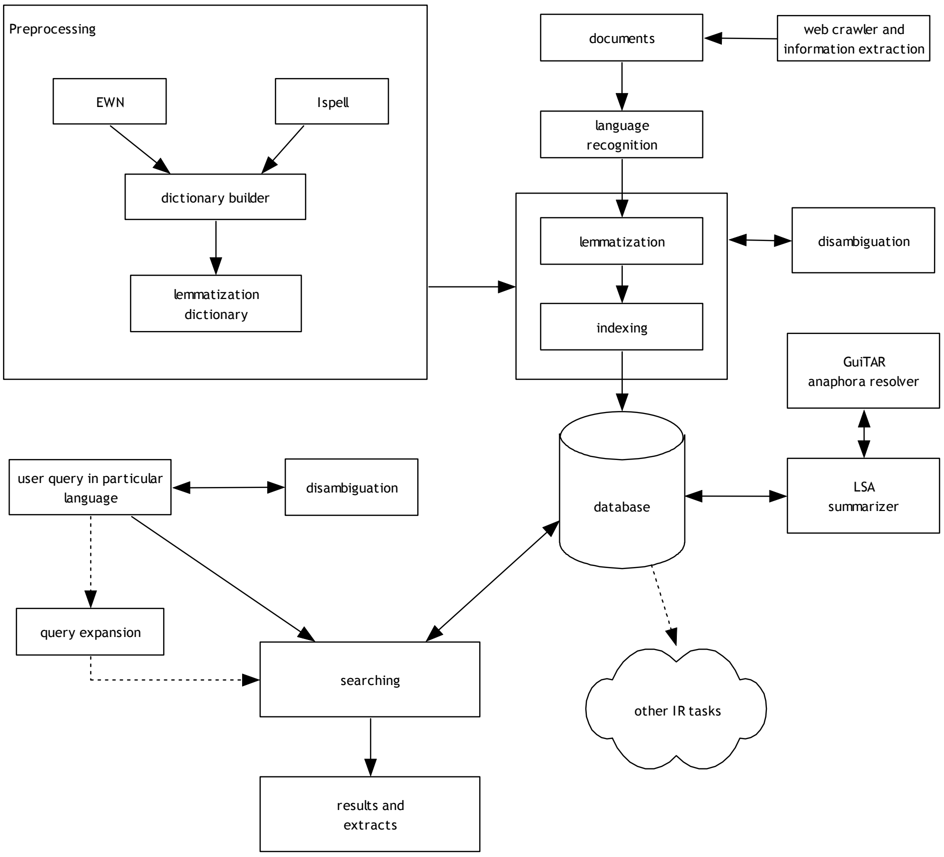

FIGURE 1 – SYSTEM OVERVIEW

LANGUAGE RECOGNITION

The multilingual processing implies a need for a language recognition module. Its task is not only to distinguish the language but to recognize the text coding as well. There are many language recognition approaches. We used two of them.

The first one results from a different letter frequency in languages. Except for language determination, letters are also used for text coding recognition. For each language and document, a binary vector is created where ones are at the position of characteristic letters (e.g. letters with diacritics). The document vectors are compared with the language vectors by the well-known Hamming distance measure (i.e. the number of disagreements between two vectors).

The second method is based on a stop-word list. The list includes words not carrying any particular information. They are highly specific for each language. Stop-words are for example: a, an, the, of, from, at, is, etc. Finally, the module chooses the correct lemmatization dictionary, according to the recognized language.

The comparison of both methods was discussed in [4].

LEMMATIZATION

Lemmatization transforms words into their basic forms. Dictionary lemmatization was used because of its simplicity and generality. The lemmatization dictionary was created by the extraction of word forms from the Ispell program (see [3]). Thanks to Ispell, we were able to generate all existing word forms from stems stored in the Ispell dictionary. We considered the stem a basic form of the word. This works perfectly in the case of English, but some problems appear in Czech. In general, languages with a rich flex are more difficult to process in general. We used a Czech morphological analyzer [6] to overcome this problem. In the case of

English, lemmatization is relatively simple. It is possible to apply an algorithmic method – Porter's algorithm.

WORD SENSE DISAMBIGUATION

Word sense disambiguation (WSD; [10]) is a necessary module in most of the natural language processing (NLP) systems. It allows distinguishing of the meaning of a text or a message. Polysemous words may occur in any language. Ambiguity causes many problems, which may result in the retrieval of irrelevant documents. Disambiguation is a relatively self-contained task, which has to be carried out within the indexing. It has to distinguish between words which have identical basic forms but different meanings. The decision about the right meaning requires the knowledge of the word's context.

We implemented a disambiguation method based on the Bayesian classifier. Each meaning of the word was represented by a class in the classification task. The total number of meanings for each ambiguous word was obtained from the EWN thesaurus. Our analysis discovered that nearly 20% of English words are ambiguous. This shows the importance of disambiguation in all NLP tasks. In the course of our implementation, some heuristic modifications were tested with the aim to refine the disambiguation accuracy, as discussed in [4].

INDEXING

We introduced a bit of an unusual approach to indexing. For language independent processing, we designed a technique which transforms all the multilingual texts into an easily processed form. The EWN thesaurus was used for this task (see [2]). It is a multilingual database of words and relations for most European languages. It contains sets of synonyms – *synsets* – and relations between them. A unique index is assigned to each synset; it interconnects the languages through an inter-lingual-index in such a way, that the same synset in one language has the same index in another one. Thus, cross-language searching can easily be performed. We can, for example, enter a query in English, and the system can retrieve Czech documents as a result, and vice versa.

With EWN, completely language independent processing and storage can be carried out, and moreover, synonyms are identically indexed.

SEARCHING

Our system deals with the representation, storage, and presentation of multilingual information sources. Documents are transformed into the internal language independent form. This is done in the lemmatization and indexing phase. Each document can be described by a set of indexes, representing its main topics. Such indexes can be determined in a fully automatic way. A weight is assigned to each word. It implies its expected semantic significance within the whole document. This framework is proposed to accomplish partial matching based on the similarity degree of a document and a query. Moreover, term weighting and scoring according to user queries enables the sorting of retrieved documents according to their relevance.

We use a slightly modified TF-IDF (Term Frequency - Inverse Document Frequency) principle for the term scoring algorithm. The weight of the term t_i in the document d_j denoted w_{ij} is the product $w_{ij} = tf_{ij} \cdot idf_{i,}$ where $\mathrm{tf}_{\mathrm{ij}}$ is the term frequency of t_i in d_j and idf_i is the inverted document frequency of t_i in the collection D.

A resultant candidate set is computed for each term in the user query. The set is scored by the relevance measured with regard to the term. If more terms are used in the query, candidate sets' intersection or union is performed according to the logical operation in the user query (AND or OR). In the case of intersection, document weights are adjusted by simple summation of candidate values.

From the user's point of view, the searching process is intuitive. The user query is interpreted as a set of terms describing the desired result set. Query terms are lemmatized and indexed into an internal form, and the query can be expanded with the use of EWN. This step is optional. Each word from the query should be disambiguated[1] to prevent a retrieval of irrelevant documents. Afterwards, the searching is performed, and the results are displayed. For each document, a full text and its summary are available. All operations are performed upon a relational database. It contains summarized data, the lemmatization dictionary, and the EWN thesaurus.

QUERY EXPANSION

It is not simple to create a query which fully covers the topic of our interest. We introduced a query expansion module that provides a simple, yet powerful, tool for changing the queries automatically. The expansion can be done in different ways. Synsets' interconnections were obtained from the EWN thesaurus for this purpose. We used 10 different relationships. They are presented together with their weights and types in the table below. The weights are used in the TF-IDF scoring algorithm. They were subjectively designed according to the relationship between the query term and its expansion.

Relationship	Relationship weight	Relation type
similar_to	8	Similar
be_in_state	6	Similar
also_see	8	Similar
derived	3	Similar
hypernym	2	Superordinates
Holo_portion	3	Superordinates
Holo_part	3	Superordinates
Holo_member	3	Superordinates
Particle	3	Subordinates
Subevent	2	Subordinates

TABLE 1 – EXPANSION RELATIONSHIPS

A query expansion can significantly improve the system recall. It will retrieve more documents, which are still relevant to the query (see Results section). The user is able to restrict the expansion level to any combination of similar, subordinate and superordinate words. The expanding terms have a lower weight than those entered directly by the user.

SUMMARIZATION

Within the scope of the MUSE system, we developed a summarizing module that should lead to better orientation in the retrieved texts and to faster searching. Our approach to summarization follows what has been called a term-based strategy: find the most important information in a document by identifying its main terms, and then extract from the document the most important information about these terms [11].

[1] This is not done at the moment but we plan to implement this feature in the next few months.

The summarization algorithm is based on the Latent Semantic Analysis (LSA) [3]. LSA is a technique for extracting the ‘hidden’ dimensions of the semantic representation of terms, sentences, or documents, on the basis of their contextual use. In other words, it can capture interrelationships among terms, so that terms and sentences can be clustered on a ‘semantic’ basis, rather than on the basis of words only. It has been extensively used for various NLP applications, and lately for summarization as well. The core of the analysis is an algebraic method called Singular Value Decomposition (SVD). Let us briefly, and without any deeper mathematical background, explain the SVD principles. First of all, we create a terms-by-sentences matrix, where each value represents a weighted frequency of a term in a sentence. The matrix is further processed by SVD. As a result, we obtain information about the topics of the text[2], and their significances. Moreover, we are able to quantify the importance of each sentence for each topic. For a more detailed mathematical description, see [7].

The summarization method proposed by [12] uses the representation of a document thus obtained to choose the sentences to go in the summary on the basis of the relative importance of the ‘topics’ they mention. The summarization algorithm simply chooses for each ‘topic’ the most important sentence for that topic. This method has a significant drawback. The number of important ’topics’ that have to be identified must be the same as the number of sentences we want to include in the summary. As a result, a summary may include sentences about ’topics’ which are not particularly important. In order to solve the problem we changed the sentence selection criterion. Our idea is to choose sentences with the greatest combined weight across all topics, possibly including more than one sentence about an important topic, rather than one sentence for each topic. However, the algorithm still requires a method for deciding how many topics to include in the sentence selection criterion, and therefore in the summary. If we take too few, we may lose topics which are important from the summarization point of view. But if we take too many, we end up including less important topics, as Gong and Liu’s algorithm does. In [9] we proposed a way of determining automatically the number of significant topics. In summarization, we know what percentage of the full text the summary should be, and after computing LSA, we know the contribution of each topic. We took the most significant topics until the sum of their contributions exceeded the summarization percentage. We showed that our modification results in a significant improvement over the Gong and Liu’s method.

‘Purely lexical’ LSA determines the main ‘topics’ of a document on the basis of the most common meaning of terms, single words, as usual in LSA. In [8] we showed, however, that anaphoric information can easily be integrated in a mixed lexical / anaphoric LSA representation, by generalizing the notion of ‘term’ used in SVD matrices to include *discourse entities* as well. In the input SVD matrix we can use two types of ’terms’: terms in the lexical sense (i.e. words) and terms in the sense of discourse entities, represented by anaphoric chains. In such a case the representation of sentences specifies not only whether they contain a certain word, but also whether they contain a mention of a discourse entity. With this representation the chain ‘terms’ may tie together sentences that contain the same anaphoric chain, even if they do not contain the same word. The resulting matrix can then be used as input to SVD as before. We used the anaphora resolver GuiTAR, developed at the University of Essex. It is able to identify anaphors that can be further connected to anaphoric chains (e.g.: president Bill Clinton – he – the president – Clinton).

In [9] we compared our algorithm with the existing approaches. In evaluation we used the DUC2002 corpus [13]. In 2002, DUC (Document Understanding Conference) included a

[2] The topic is determined by a linear combination of original terms. If a word combination pattern is salient and recurring in document, this pattern will be captured and represented by a topic.

single-document summarization task, in which 13 systems participated. The test corpus used for the task contains 567 documents from different sources; 10 assessors were used to provide two 100-word human summaries for each document. In addition to the results of the 13 participating systems, the DUC organizers also distributed baseline summaries (the first 100-words of a document). The coverage of all the summaries was assessed by humans. In 2003 the ROUGE measure, the most respected evaluation measure, was introduced. It is able to measure the similarity between human summaries and automatically created abstracts, and it is the top evaluation measure so far. We showed in our ROUGE evaluation that our system performs as well as the best participating system in DUC 2002.

We put the main accent on the multilingualism of our summarizer. LSA is a totally language independent process. The only difference in processing different languages is the stop-word list and lemmatization. In anaphora resolution, the situation is different. So far, we have enriched our summarization method with anaphoric knowledge only for texts written in English. Now, we plan to create an anaphora resolver for the Czech language in which we intend to implement similar resolution algorithms as the ones in GuiTAR. For demonstration of the summarizer functionality, see the following summary of our introduction.

> Most of over 3M online Internet users in the Czech Republic are searching not only Czech pages but English, Slovak, German and others as well. The EuroWordNet thesaurus (EWN) [2] is the core of our multilingual searching approach, and the heart of our summarizer is Latent Semantic Analysis (LSA) [3]. Some of modules, namely language recognition, lemmatization, word sense disambiguation and indexing, were described in [4]. Multilingual searching and user query expansion are possible due to EWN that is also used in lemmatization, indexing and a query conversion into the language independent form. Queries are entered in one of the languages (currently Czech and English). However, it should be noted that the principles remain the same for an arbitrary number of languages.

FIGURE 2 – EXAMPLE SUMMARY

RESULTS

We created a testing corpus which includes Czech and English texts, in particular – press articles from ČTK and Reuters news agencies. The corpus consists of a total number of 82000 Czech and 25000 English articles. They were chosen from 5 classes – weather, sport, politics, agriculture, and health. A 100-word extract was created for each document.

Table 2 shows the influence of query expansion on the retrieved results. In each setup we present a total number of retrieved documents (*all* column) and the number of documents that are relevant in the top 30 (*rel* column). The first column is a basic setup, no extension is applied. The average precision exceeded 90 percent. In the next columns you can read the results when query expansion was used. Subordinate relations preserve satisfactory precision because more specific terms are searched. On the contrary, superordinate relations can introduce some general terms, making results less relevant. The main advantage of query expansion is the enrichment of the result set. Our system achieved a precision level of up to 96% over the first 30 retrieved documents. We compared the retrieval performance of the Google approach, the widely accepted search method, and that of our MUSE system. Our approach and the state-of-the-art Google search engine are compared in table 3. We measured the intersection between MUSE and Google in the first 10 and 30 retrieved documents on the same query. The three right-most columns show the MUSE performance when all possible query expansion levels were used.

Query	**Without expansion**		**Expansion by similar relations**		**Expansion by subordinate relations**		**Expansion by superord. relations**		**Precision with all expansions**		**Precision without expansion**	**Precision with all expansions**
	all	rel	all	rel	all	rel	all	rel	All	rel		
formula & one & champion	88	27	88	27	88	27	465	26	465	26	90,0	86,7
terorismus & útok	265	29	265	29	265	29	300	29	300	29	96,7	96,7
white & house & president	2393	29	2657	28	2393	29	5880	23	6116	23	96,7	76,7
povodeň & škody	126	29	126	29	126	29	126	29	126	29	96,7	96,7
cigarettes & health	366	25	366	25	366	25	393	25	393	25	83,3	83,3
rozpočet & schodek	2102	30	2102	30	2102	30	2174	30	2174	30	100,0	100,0
plane & cash	221	29	221	26	211	29	2306	29	2306	29	96,7	96,7

TABLE 2 – QUERY EXPANSION RESULTS

Query	**MUSE approach**			**MUSE approach with query expansion**		
	Inters. 30	Inters.10	Total number	Inters. 30	Inters. 10	Total number
formula & one	25 (83%)	9 (90%)	351	24 (80%)	9 (90%)	2075
national & park	9 (30%)	3 (30%)	508	9 (30%)	3 (30%)	1198
religion & war	20 (67%)	7 (70%)	73	20 (67%)	7 (70%)	74
water & plant	11 (37%)	7 (70%)	73	6 (20%)	4 (40%)	1489
hockey & championship	20 (67%)	7 (70%)	82	20 (67%)	7 (70%)	85
traffic & jam	18 (64%)	6 (60%)	64	16 (53%)	6 (60%)	165
heart & surgery	16 (53%)	7 (70%)	563	17 (57%)	7 (70%)	703
weather & weekend	19 (63%)	10 (100%)	140	16 (54%)	10 (100%)	158

TABLE 3 – RESULTS COMPARED WITH GOOGLE

We also tested the influence of summarization on the quality of the retrieved results. To verify the influence, we performed the same queries on both the full text and summarized corpus. Searching in summaries improves the response times of the system significantly (see table 5), without any remarkable loss of precision (see table 4). The number of relevant documents in the top 30 retrieved results is basically the same. The intersection of the documents retrieved by searching in both corpuses is approximately 50 %.

Query	**Summary and fulltext intersection in the first 30 retrieved documents**	**Summary relevance in the first 30 retrieved documents**
formula & one	21 (70%)	26 (86%)
national & park	10 (33%)	20 (67%)
religion & war	4 (13%)	26 (86%)
water & plant	7 (23%)	14 (47%)
hockey & championship	16 (53%)	29 (97%)
traffic & jam	11 (36%)	23 (76%)
heart & surgery	16 (53%)	30 (100%)
weather & weekend	5 (16%)	28 (93%)

TABLE 4 – SUMMARY COMPARED WITH FULLTEXT

Query	**Searching time in full text [ms]**	**Searching time in summaries [ms]**
formula & one	6359	984
national & park	8797	1312
religion & war	6172	922
water & plant	8734	1015
hockey & championship	1938	547
traffic & jam	3656	688
heart & surgery	5656	1031
weather & weekend	4125	703

TABLE 5 – SEARCH TIME COMPARISON

CONCLUSION AND FURTHER WORK

Our results show approximately 70 % similarity with the Google approach in the top 30 retrieved documents. However, MUSE has several advantages in comparison with Google. Firstly, our system respects a multilingual environment. If we enter a query in English, Google is not able to find any relevant documents written in another language. On the contrary, MUSE will retrieve both English and Czech documents. Secondly, synonyms are considered equal in the searching process. Moreover, we provide query expansion, and finally, a part of the system is an automatic summarizer. Searching in summaries is reasonably precise and five times faster.

There is a problem related to the actual EWN structure – a missing word's equivalents in non-English languages. This can cause some difficulties in cross-language searching. As EWN is gradually being completed, this problem will disappear.

The system will be tested in our university digital library, which offers large numbers of texts, mostly in Czech and English. We believe that MUSE it will help our students and researchers to gain information more efficiently and quickly.

REFERENCES

1 http://global-reach.biz/globstats/refs.php3

2 http://www.illc.uva.nl/EuroWordNet/

3 T. K. Landauer and S. T. Dumais: *A solution to plato's problem: The latent semantic analysis theory of the acquisition, induction, and representation of knowledge*. In Psychological Review, 104:211–240.

4 Karel Jezek and Michal Toman: *Documents Categorization in Multilingual Environment*. In Proceedings of ELPUB, Leuven, Belgium, 2005.

5 http://fmg-www.cs.ucla.edu/fmg-members/geoff/ispell.html

6 http://quest.ms.mff.cuni.cz/pdt/Morphology_and_Tagging/Morphology/index.html

7 Josef Steinberger and Karel Jezek: *Text Summarization and Singular Value Decomposition*. In Lecture Notes in Comp.Sc.2457 pp.245-254, Springer-Verlag 2004.

8 Josef Steinberger, Mijail A. Kabadjov, Massimo Poesio and Olivia Sanchez-Graillet: *Improving LSA-based Summarization with Anaphora Resolution*. In Proceedings of EMNLP, Vancouver, Canada, 2005.

9 Massimo Poesio, Josef Steinberger, Mijail A. Kabadjov and Karel Ježek: *An Approach to Summarization Combining Anaphoric and Lexical Knowledge within the LSA Framework.* Submitted for ECAI'06, Trento, Italy, 2006.

10 Michal Toman and Karel Ježek: *Modifikace bayesovského disambiguátoru*. In Znalosti 2005, VŠB-Technická univerzita Ostrava, 2005.

11 E. Hovy and C. Lin: *Automated text summarization in summarist. In ACL/EACL Workshop on Intelligent Scalable Text Summarization*, Madrid, Spain, 1997.

12 Y. Gong and X. Liu: *Generic text summarization using relevance measure and latent semantic analysis.* In Proceedings of ACM SIGIR, New Orleans, US, 2002.

13 http://www-nlpir.nist.gov/projects/duc/data.html