Building an Information Retrieval System for Serbian - Challenges and Solutions

Miroslav Martinović [1], *Srđan Vesić* [2], *Goran Rakić* [2]

[1] Department of Computer Science, College of New Jersey, U.S.A.
[2] Faculty of Mathematics, University of Belgrade, Serbia

mmmartin@tcnj.edu, srdjan@matf.bg.ac.yu, grakic@devbase.net

10.21437/Interspeech.2007-437

Abstract

We describe challenges encountered while building an information retrieval system for Serbian language. Approaches designed and adopted to handle them are depicted and illuminated in this paper. As a backbone of our system, we used SMART retrieval system which we augmented with features necessary to deal with specificities of the Serbian alphabet. In addition, morphological richness of the language accentuated implications of the text preprocessing phase. During this phase, we devised two algorithms which increased retrieval precision by 14% and 27%, respectively. Testing was conducted using two gigabyte EBART collection of Serbian newspaper articles.

Index Terms: information retrieval, text processing, alphabet, morphology, precision, recall.

1. Introduction

Over the last few years, design and development of computational linguistic resources for languages spoken by less than ten million people (small languages) is receiving a considerable attention. In addition, ethnolinguistical analysis of the relationship between culture, thought, and language and in particular, a study of a minority language within the context of the majority population has been gaining momentum ([5], [7], [8], [11]). Standardization, linguistic normalization and revitalization of small languages have been initiated and promoted. Speakers of smaller languages have gained awareness that their languages belong to the world's cultural heritage, and are becoming more and more inclined to use their native tongues at a broader scale. The rising number of web-pages in lesser-used languages demonstrates this fact.

Work described in this paper started as part of a Sabbatical research that investigated possibilities and limitations of transfer of NLP technologies from a resource rich and morphologically moderate language like English to a morphologically intricate, significantly less resourced and linguistically only remotely related one like Serbian. A collaborative work has been instigated on a development of human language technology for Serbian language with an ultimate goal of building an open domain question answering system. Along the way and as byproducts, it was envisioned that a number of specific NLP applications for Serbian would also be concocted (e.g. machine readable dictionary, corpora, tagger, morphological analyzer, shallow parser, information retrieval system). As a component of the planned Q&A system, an information retrieval system (SPRETS) has been developed. While there were viable attempts towards information retrieval of Serbian speech and text ([4]), SPretS is a first, novel and unique full blown and complete retrieval system for Serbian language.

2. Serbian Information Retrieval System (Srpski PRETraživački Sistem - SPRETS)

Designing and developing of an information retrieval system for Serbian language was carried out through several phases. Having in mind the original intention of investigating if and to what extent, a transfer of existing technology is possible in this context, we were to attempt to build the system around the widely-used, freely available SMART system ([14]). In fact, we made an effort to follow the steps taken during development of an information retrieval system for English (actually, retrieval component of question answering system QASTIIR ([11])).

In search of a relatively sizeable text collection of open domain, contemporary Serbian language, we opted for and obtained access to the EBART archive of document collections ("http://www.arhiv.co.yu/aa_opis.htm"). This (approximately) two gigabytes collection comprises of articles from fifteen of the most influential Serbian newspapers with highest circulation in the country compiled from the year 2003 on.

2.1. Overall Structure of SPRETS

The overall structure of SPRETS is depicted in Figure 1.

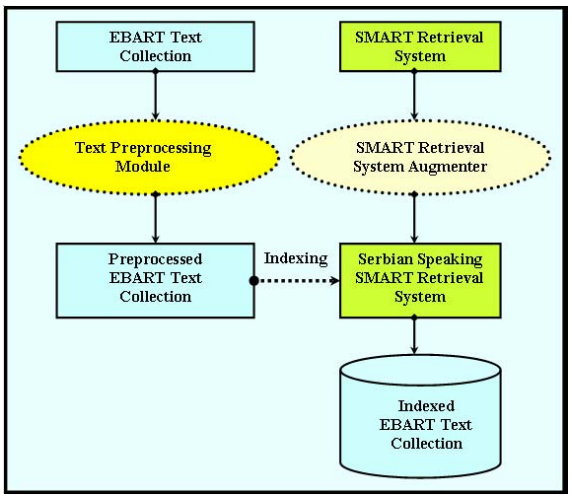

Figure 1: *SPRETS' Overall Structure.*

Both, the text collection and the SMART system are modified and preprocessed in order to make a proper indexing process possible. The preprocessed text collection is then fed into augmented SMART retrieval system enabled for Serbian

language processing. Indexing that follows produces an index of the EBART text collection. An identical preprocessing is done to an expanded query at the time of the actual search.

2.1.1. Preprocessing EBART Text Collection

Preprocessing of the text collection at the indexing time, as well as of an expanded query at the search time comprises of three main phases: (i) conversion of non-ASCII Serbian letters (***ć***, ***č***, ***đ***, ***š***, and ***ž***) into their corresponding ASCII transcripts (*cx*, *cy*, *dx*, *sx*, and *zx*), (ii) stopping, and (iii) word conflation.

While conversion of non-ASCII letters and stopping are rather straightforward operations, design and development of a successful word conflation algorithm is not as much. In addition, it has considerably reaching consequences as far as functioning of the retrieval system is concerned.

2.1.1.1 Word Conflation and Stemming in SPRETS

Serbian language possesses a (almost notoriously) complex morphology: seven noun cases, three noun genders, three noun numbers, six verb tenses, six verb forms, etc., etc. ([5], [15]). It was quite reasonable to assume that the quality of word conflation would have essential implications on retrieval process. This was eventually proved by our subsequent investigation, as well as the assessment and evaluation of the system's performance.

Thus, in addition to development of our retrieval system, we decided to experiment with it and measure the impact of different conflation modules. Thus, we set up a threefold experiment to run the system and measure its performance first without a conflation module, then using a stemmer with an exhaustive set of rules, and finally a stemmer with only a rudimentary stemming tenets. Consequently, we developed two algorithms for conflation modeled on some well known predecessors in the field ([3], [7], [8], [9], [10], [12], [13]).

2.1.1.1.1 Exhaustive Conflation Algorithm (ECA)

This conflation algorithm consists of about fifty rules of transformation that include stemming of word endings but also prefix analysis and letter substitutions in the middle of the word. A typical rule consists of its label, word measure condition (very short words are not transformed), word form description followed by a transformation (format established for STELEMMIN, generic minimal stemmer/lemmatizer ([10])). Since transformation can be done anywhere in a word, position of transformation is needed as well as descriptions of pattern to be replaced and pattern to replace it. All descriptions are regular expressions. An example of a typical word transformation rule would be as follows:
31; M>3; word has *a* at the next to last position preceded by a consonant and not followed by ***ć***, ***č***, ***đ***, ***š***, ***ž***, ***lj***, ***nj***, or ***dž***; remove the *a*; replace it with *''*.

This method attempts to recognize word similarities not only when words differ in gender, number and tense but also when complex sound alterations take place. In Serbian, those include various linguistic phenomena as palatalization, *l* into *o* conversion, disappearing *a*, etc ([5], [15]).

Palatalization rule for instance implements conversion of *k*, *g*, *h* sounds into ***č***, ***ž***, ***š*** when they are found in front of vowels (e.g. nominative case *dru**g*** transforming into vocative *dru**že***).

Rule implementing conversion of *l* into *o* is used when gender of adjectives or nouns changes from feminine to masculine or neutral (e.g. nominative feminine *ce**l**a* and nominative masculine *ce**o***).

The *disappearing a* rule deletes next to last *a* in a word if *a* is preceded by a consonant and is not followed by ***ć***, ***č***, ***đ***, ***š***, ***ž***, ***lj***, ***nj***, or ***dž*** (e.g. genitive singular *manjka* derived from nominative singular of the noun *manj**a**k*).

In another example, word ***matematički*** gets transformed into ***matematik*** by first applying the rule that removes ending ***ki*** and then another rule that replaces ***č*** into ***k***.

Word ***najjači*** is transformed into *jk* by applying the following rules in order: deletion of prefix *naj*, deletion of a vowel at the end of a word, replacement of ***č*** by ***k***, deletion of *a* at the next to last position when surrounded by consonants.

Because of the exhaustive nature of the algorithm, it succeeds in transforming a great majority of words into their corresponding common stems. However and expectedly, cases of excessive normalization were observed, as well. For example, word *pruge* (meaning *railways*) and *prugasti* (meaning *stripy*) get transformed into the same word *pruga* (*railway*). And, because of alike cases of excessive normalization, we decided to experiment with an alternative algorithm. Leaving out rules observed to be responsible for the before mentioned cases lead us to our next algorithm.

2.1.1.1.2 Rudimentary Conflation Algorithm (RCA)

This algorithm was developed by reduction of the previous one to only a small subset of its original rules. Here, rules that deal with previously brought up complex linguistic phenomena are left out together with some other rules that were recognized to cause excessive stemming.

When processing query *pruge Srbije* (*railways of Serbia*), ECA conflated words *pruge* (*railways*) and *prugaste* (*stripy*) into the same root word and consequently produced rather amusing but serious retrieval errors: it returned an article that dealt with popularity of Barbies in *stripy* bathing suites.

In another similar case, given query *kirija* (*rent*), EC algorithm reduced words *kirija*, *Kira* (a first name) and *Kiri* (last name) to a same stem.

New RC algorithm is free of excessive normalization and capable to distinguish between and produce different stemmed forms in cases analogous to our previous examples (*pruge* and *prugaste*; *kirija*, *Kira* and *Kiri*). It overall showed a greater precision as detailed later in the paper.

2.1.2. Enabling SMART System for Serbian

When we first attempted to utilize the generic freeware information retrieval system SMART, problems encountered were of exclusively technical nature. First, in order to adapt SMART to newer versions of GNU/Linux systems on hand, a number of obsolete system calls had to be modified and a few errors in Makefile compilation procedures adjusted.

Subsequently, letters ***ć, Ć, č, Č, đ, Đ, š, Š, ž,*** and ***Ž*** in Latin version of Serbian alphabet put forward an additional challenge. Thus, SMART had to be further adapted to include a subset of ISO-8859-2 ("http://nl.ijs.si/gnusl/cee/iso8859-2.html") definitions. Furthermore, additional modifications were included to designate which of the new letters are considered upper and which lower case.

Moreover, Serbian language also uses Cyrillic alphabet and in fact, a majority of dailies, newspapers and other publications are written in Cyrillic. In order to enable processing of Cyrillic texts, another extension had to be developed for SMART. Fortunately, all but three Cyrillic letters had unique corresponding letters in Latin (e.g. *a*=***a***,

b=***б***, *c*=***ц***, *d*=***д***, *e*=***е***, *f*=***ф***, *g*=***г***, *h*=***х***, *i*=***и***, etc.). The three Cyrillic letters ***љ,*** ***њ*** and ***џ*** that don't have their correspondents in unique Latin letters use instead two letter sequences ***lj, nj,*** and ***dž***, respectively.

3. Evaluation and Assessment

With respect to the search without any word conflation preprocessing (our 'ground zero' procedure), general assessment is that both algorithms showed a fundamental improvement in both recall and precision. While the exhaustive conflation algorithm (ECA) exhibited an essential increase in recall and a good quality growth in precision, the rudimentary conflation algorithm (RCA) revealed a decent rise in recall and a superior increase in precision.

As we mentioned earlier, a two gigabyte EBART assortment of Serbian newspaper articles from years 2003-2007 was used as our document collection. We made an effort to follow the general recommendations of the TREC guidelines for IR systems' performance assessment ("http://www-nlpir.nist.gov/projects/tv2007/tv2007.html", [16]). An evaluation experiment was set up to measure non-interpolated R-precision at EBART document counts (of 5, 10, 15, 20, etc.). Testing was done on 106 different queries compiled from a random sample of users. Precision was recorded for each individual query on each of the 'mod 5' document counts. As a final step, average individual query precisions at each of the document counts were calculated. Table 1 shows the document count precision averages along with an overall average per all document counts and percent increases of EC and RC algorithms with respect the base 'ground zero' search.

Table 1. *R-precision statistic.*

Document Count	**Ground Zero Precision**	**ECA Average Precision**	**RCA Average Precision**
5	0.698113	0.745283	0.822642
10	0.600943	0.676415	0.754717
15	0.520161	0.632704	0.701258
20	0.478338	0.575977	0.643904
...	...	...	...
Query Average	**0.57439**	**0.65759**	**0.73063**
% Increase	**-**	**14.5 %**	**27.2 %**

4. Conclusions and Future Research

As expected in a morphologically rich language as Serbian, word conflation preprocessing resulted in critical improvements for the performance of our search engine. Giving up on some ECA transformation rules that boosted the recall, brought about an increase in precision of the ensuing RC algorithm.

As mentioned in the examples of processing queries *pruge Srbije* (*railways of Serbia*) and *kirija* (*rent*), ECA showed a tendency towards occasional *overstemming*. This ultimately proved liable for the exposed trade off between the recall and precision between EC and RC algorithms.

Though a self contain task in itself, this information retrieval system was envisioned also as a component of an emerging question answering system for Serbian language partially modeled on some acknowledged achievements in the field ([1], [2], [10]). As such, it will be used in that context, as well. In particular, using it for a passage retrieval is expected to provide a new and slightly different testing ground for our foretold assessments about EC and RC algorithms. Their exposure to smaller bodies of text (passages) as opposed to whole documents, along with a significantly decreased collection size is expected to provide an interesting new slant with respect to precision and recall distributions of the two.

5. Acknowledgements

As part of a Sabbatical research investigating possibilities and limitations of transfer of NLP technologies from a resource rich to a less resourced language, the work on this paper has been funded in part by a grant awarded by Studenica foundation, as well as another grant by World University Service.

We would also like to thank the College of New Jersey for providing both logistical and financial opportunity for this research, as well as to Faculty of Mathematics at Belgrade University for being a gracious host of these activities.

In addition, we are grateful to the EBART consulting company for allowing us to use its document collection which proved to be indispensable to our project.

6. References

[1] Cardie, C. "Empirical Methods in Info Extraction", AI Magazine, 18:4, 65-79 1997.

[2] Cardie, C., Ng, V., Pierce, D., and Buckley, C. "Examining the Role of Statistical and Linguistic Knowledge Sources in a General-Knowledge Question Answering System", Sixth Applied Natural Language Processing Conference (ANLP-2000), 2000.

[3] Chrupala, G., "Simple Data-Driven Context Sensitive Lemmatization", Proceedings of SEPLN 2006, 2006.

[4] Hauptmann, A., Scheytt, P., Wactlar, H., and Kennedy, P.E. "Multi-Lingual Informedia: A Demonstration of Speech Recognition and Information Retrieval across Multiple Languages", *Proceedings of the DARPA Workshop on Broadcast News Understanding Systems*, February, 1998.

[5] Hughes, B., "Towards a Web Search Service for Minority Language Communities", Proceedings of Open Road Conference, 2006.

[6] Ivić, P., Pešikan, M., Klajn, I. and Brborić, B., Srpski jezički priručnik, Beogradska knjiga, Beograd, 2007.

[7] Jesperson, O., Language, its nature, origin and development, George Allen & Unwin, London, 1921.

[8] Korenius, T., Laurikkala, J., Jarvelin, K. and Juhola, M. "Stemming and Lemmatization in the Clustering of Finnish Text Documents", Proceedings of the 13th ACM International Conference on Information and Knowledge Management, Session IR-7, pp. 625 - 633, 2004.

[9] Kraaij, W. and Pohlmann, R., "Porter's stemming algorithm for Dutch", Noordman LGM and de Vroomen WAM, eds. *Informatiewetenschap 1994: Wetenschappelijke bijdragen aan de derde STINFON Conferentie*, Tilburg, pp. 167-180, 1994.

[10] Kraaij, W. and Pohlmann, R. "Evaluation of a Dutch stemming algorithm" Rowley J, ed. *The New Review of*

Document and Text Management, Vol. 1, Taylor Graham, London, pp. 25-43, 1995.

[11] Martinovic, M., and Rofrano, L. "SteLemMin - A Generic Minimal Stem Algorithm for Word Conflation and Lemmatization", Proceedings of Workshop on Computational Modeling of Lexical Acquisition, 2006.

[12] Martinovic, M., "Integrating Statistical and Linguistic Approaches in Building Intelligent Question-Answering Systems", Proceedings of the SSGRR 2002 International Conference on Advances in Infrastructure for e-Business, e-Education, e-Science, and e-Medicine on the Internet, 2002.

[13] Paice, C. D., "Another Stemmer", ACM SIGIR Forum, Vol. 24, Issue 3, pp. 56-61, 1990.

[14] Porter, M. F., "An Algorithm for Suffix Stripping", Program Vol. 4, No. 3, pp. 130-137, 1980.

[15] Salton, G., editor, The SMART Retrieval System: Experiments in Automatic Document Processing. Prentice Hall, 1971.

[16] Stevanović, M., Savremeni srpskohrvatski jezik, I, II, Beograd, 1994.

[17] Voorhees, E. M., "Variations in Relevance Judgements and the Measurement of Retrieval Effectiveness", Information Processing and Management, 36(5), pp. 697-716.