Improvement of Queries using a Rule Based Procedure for Inflection of Compounds and Phrases

Ranka M. Stanković

Abstract—**The selection of words chosen for a query, crucial for the quality of results obtained by the query, can be substantially improved by using various lexical resources. Thus, for example, morphological dictionaries enable morphological expansion of queries, which is very important in highly inflective languages, such as Serbian. This paper discusses issues related to improvement of queries using a rule based procedure implemented in WS4LR, a workstation for manipulating heterogeneous lexical resources developed by the Human Language Technology Group at the University of Belgrade. The procedure is used for automatic production of lemmas for a morphological dictionary from a given list of compounds, and its evaluation on several different sets of data is given. Several examples illustrate how this procedure can be used for improvement of queries for web search engines. Results obtained for these examples show that the number of documents obtained through a query by using our approach can be remarkably increased.**

Index Terms—**Electronic dictionary, inflection, compounds, query expansion.**

I. Introduction

The Human Language Technology group from University of Belgrade (HLT) has been developing various lexical resources over quite a long period, reaching a considerable volume to date. HLT group has produced an integrated and easily adjustable tool, a workstation for language resources, labeled WS4LR, which greatly enhances the potential of manipulating each particular resource as well as several resources simultaneously [1]. This tool has already been successfully used for various language processing related tasks including query expansion.

Dictionaries are one of the most important resources in various phases of the automatic analysis of text [2]. The system of morphological electronic dictionaries of Serbian follows the methodology and format (known as DELAS/DELAF) presented in [3]. E-dictionaries of simple word forms have reached a considerable size: approximately 120,000 entries in total [4].

In recent years the interest for multi-word units and compounds is growing rapidly, and this paper focuses on the morphological description of compounds compatible with the methodology used for simple words. At present, the dictionary of compounds has 2633 lemmas covering different parts of speech.

Development of the dictionary of compounds is not an easy task, so automated creation of lemmas for such a dictionary for a given list of compounds is of great importance. Such a procedure, which is based on rules and relies on data from e-dictionaries of simple words is described in Section II. The developed procedure has been evaluated on several different data sets and afterwards included in WS4LR.

Section III of this paper demonstrates how the described procedure can be used for query improvement. WS4LR architecture is described with special attention to compound management system. Usages of various lexical resources for query improvement are given, with integrated module for automatic detection of structure and inflectional characteristics of compounds. The application of the procedure presented is demonstrated on several examples of morphological expansion of key phrases for web search engines.

II. A Rule Based Procedure for Inflection of Compounds and Phrases

A. Compounds Dictionary

Morphological description of compounds, compatible with the methodology used for simple words, relies on the usage of Finite-State Technology [5]. The final aim is to produce the counterpart of DELAS/DELAF dictionaries of simple words for compounds – DELAC/DELACF.

The following example illustrates the content of compound dictionaries and some problems in their development. For example, the compound *beli medved* ‘polar bear’ should be entered in the DELAC dictionary of compounds [6], as follows:

beli(beo.A38:adms1g) medved(medved.N2:ms1v),
NC_AXN+N+Comp+Zool

Information contained in this entry should provide for automatic creation of all inflected forms for the DELACF dictionary, such as:

beloga medveda,beli medved.NC_AXN:ms4v

Manuscript received on May 9, 2008. Manuscript accepted for publication June 20, 2008. The presented work was done within the Human Language Technology group, University of Belgrade, Serbia.

Ranka M. Stanković is with the Faculty of Mining and Geology, University of Belgrade, Đušina 7, 11000 Belgrade, Serbia (phone: +381 11 3219-148; fax: +381 11 3243 978; e-mail: ranka@rgf.bg.ac.yu).

beli medvede,beli medved.NC_AXN:ms5v
belim medvedom,beli medved.NC_AXN:ms6v

The production of a lemma in the DELAC dictionary for a given compound proceeds in several steps:

1) For each compound component determine its lemma in DELAS dictionary with inflectional class code, and grammatical categories from the DELAF dictionary. For instance, for *beli* the lemma is *beo*, its inflectional class code is A38, and grammatical categories of the form *beli* are *:*adms1g;
2) Determine the inflectional class code for the compound (e.g. NC_AXN in the above example);
3) Determine the syntactic and semantic markers for the compound (e.g. +N+Comp+Zool in the above example).

In order to facilitate this task, a special tool within WS4LR [7] has been developed that assists in obtaining some of the necessary information from existing DELAS/DELAF dictionaries. Even with the help of this tool (for instance by reducing the number of errors in DELAC entries), the development of DELAC dictionary for Serbian is very time consuming. This led to the decision to develop a procedure for automatic (or semiautomatic) construction of DELAC type dictionary from a given list of compounds.

B. Rules Design

The procedure for automatic construction of DELAC type dictionary is based on a set of rules. The rule design strategy is a result of expert knowledge on morphology and the analysis of an existing manually created compound dictionary. The task of the rule based procedure is to generate the complete compound lemma for the dictionary of DELAC type based on the strategy. However, the strategy and the procedure are independent, and changes in the strategy, in general do not affect the procedure itself. This system design made experiments with various rule strategies possible – the final strategy used to evaluate the procedure is a result of several iterations.

The rule based strategy presently consists of 53 rules: 19 rules for compounds with 2 components, 20 rules for compounds with 3 components, 8 rules for compounds with 4 components, and 3 rules for compounds with 5 and 6 components. Each rule defines conditions components of a particular compound and/or separators between them must fulfill in order to get a particular inflectional class assigned to them. The rules are applied in the order they are listed.

Conditions defined for each rule are of two types: the first type specifies grammatical categories of compound components and they usually apply to the components that inflect, while the second type specifies additional conditions like semantic and/or syntactic markers. This can best be illustrated by the example of rule number 43, as shown in the table I.

TABLE I
EXAMPLE OF RULE NUMBER 43, CLASS NC_N6X

Class	Gramm. condition	Frequ ency	Additional conditions
NC_ N6X	_:fs1q__	3	(The first component is a noun)
	_:ms1q__	2	AND
	_:ms1v__	2	((The second, the third and fourth
	_:ns1q__	1	component are in genitive) OR
	_:fs1v__	0	(The second word is a preposition
	_:ns1v__	0	and the third word agrees with it))

This rule is applied as follows: if the first component satisfies (according to the dictionary of simple words) the grammatical conditions (which imply that the first component has to be a noun), and if the second and the third component and the separator between them satisfy one of the remaining additional conditions, then the rule class will be suggested for the given compound. The frequency column gives the number of compounds in the existing DELAC dictionary that satisfy the particular rule line. Examples of additional conditions are:

1) *tehnolog održavanja poljoprivredne mehanizacije* 'agricultural equipment maintenance technologist' where *tehnolog* 'technologist' is a noun that satisfies the condition :ms1v, *održavanja* is in the genitive case (from *održavanje* 'maintenance'), *poljoprivredne* is in the genitive case (from *poljoprivredni* 'agricultural')and *mehanizacije* is in the genitive case (from *mehanizacija* 'equipment');
2) *motor sa unutrašnjim sagorevanjem* 'engine with combustion chamber' where *motor* 'engine' satisfies the condition :ms1q, *sa* 'with' is a preposition that requires the instrumental case, *unutrašnjim* is in the instrumental case (from *unutrašnji* 'combustion');

All rule lines are ordered according to the listed frequency in order to prioritize some conditions in case of multiple choices. The total number of lines for 53 rules is 1014. Most inflectional classes have only one rule, with multiple rules defined only for a minority. These rules model different conditions, and they have different order in the strategy, which reflects the probability of their application.

Figure 1 depicts the XSD scheme of rules for automatic detection of the structure and inflectional characteristics of compounds. As an example, the XML form of rule number 43, for inflectional class NC_N6X, is presented in table II.

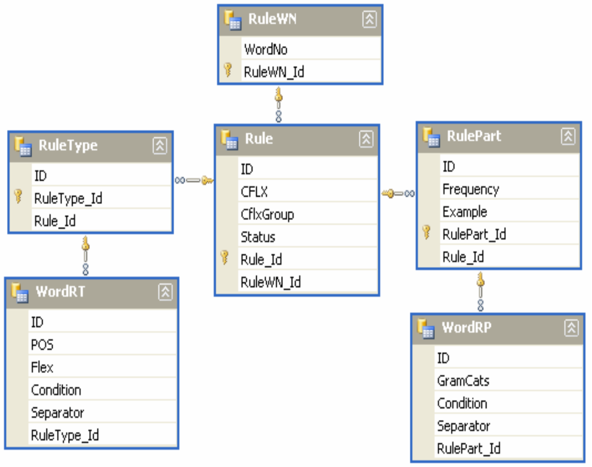


Fig. 1. XSD scheme of rules for the automatic detection of structure and inflectional characteristics of compounds

The pseudo code for automatic construction of a compound lemma goes as follows:

```
predictCFlexLema(Compound)
   // 1) lexical analysis of compound components
  generateDlf(Compund)
  foreach Component in Compound.Components
    Component.findPosLemasFromDlf
    Component.findGramCatsFromDlf
    Component.findLemasFlxCodeFromDelas
    Compund.DataSet.Add(Candidate)
  // 2) Selection of possible rules to be applied
  Rls=Rules.SelectByNumberOfComponents
  Rls.FilterByPosOfComponents
  foreach R in Rls
    dsRt=R.getDataSetRuleType(C.DataSet))
    dsRp=R.getDataSetRulePart(DsRt)
    //3) Construction of compound lemma
    foreach dr in dsRp
      Rule.GenerateCLema(dr)
```

TABLE II
XML FORM OF RULE NUMBER 43, CLASS NC_N6X

```
<Rule ID="43" CFLX="NC_N6X" Status="true">
  <RuleType ID="1">
    <WordRT ID="1" POS="N" Flex="true" />
    <WordRT ID="2" POS="*" Flex="false" Condition="GramCats,2"/>
    <WordRT ID="3" POS="*" Flex="false" Condition="GramCats,2"/>
    <WordRT ID="4" POS="*" Flex="false" Condition="GramCats,2"/>
  </RuleType>
  <RuleType ID="2">
    <WordRT ID="1" POS="N" Flex="true" />
    <WordRT ID="2" POS="PREP" Flex="false" />
    <WordRT ID="3" POS="*" Flex="false" Condition="PrepAgr,2" />
    <WordRT ID="4" POS="*" Flex="false" />
  </RuleType>
  <RulePart ID="1" Frequency="3" Example="princ na belom konju">
    <WordRP ID="1" GramCats="ms1v" />
  </RulePart>
  <RulePart ID="2" Frequency="2"
    <WordRP ID="1" GramCats="ms1q" />
  </RulePart>
  <RulePart ID="3" Frequency="2" >
    <WordRP ID="1" GramCats="ns1q" />
  </RulePart>
  <RulePart ID="4" Frequency="1" >
    <WordRP ID="1" GramCats="fs1q" />
  </RulePart>
  <RulePart ID="5" Frequency="0">
                 <WordRP ID="1" GramCats="ns1v" />
  </RulePart>
  <RulePart ID="6" Frequency="0">
    <WordRP ID="1" GramCats="fs1v" />
  </RulePart>
</Rule>
```

The first part of pseudocode relates to step one of the production of a lemma for DELAC dictionary described in part II section A. The second and third part of pseudocode are related to step two of the production of lemma for DELAC dictionary described in part II section A. Examples of the XML structure of RuleType and RulePart of part two of the pseudocode are given in table II.

C. System Evaluation

The first evaluations of the strategy have been performed using the DELAC dictionary, by comparing results from automatic processing with manually created compound lemmas. Figure 2 shows the success statistics: out of 2135 compound lemmas 219 (10.27%) either couldn't be solved, or the offered solution was incorrect. Further analysis showed that the reason for failure in some cases was the absence of some compound components from DELAS dictionary.

Only for 19 (0.89%) compound lemmas the strategy has not been properly defined, while in 4 (0.19%) cases a rule was missing. Totally or "conditionally" correct results were obtained for 1892 (88.67%) compound lemmas, where "conditionally" correct means that the inflective class code was correctly determined, which is good enough for query expansion.

Since the strategy has been designed to produce all possible lemmas by applying all the rules that meet the criteria defined, in some cases several possibilities have been offered and sorted by previously defined rule priority. Figure 2 shows that out of 1892 correct results, as many as 1667 have been offered as the first answer, 137 as second, 53 as third, 33 as fourth and 2 as fifth.

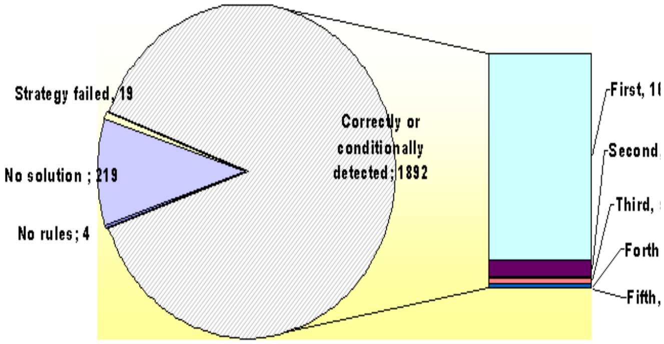


Fig. 2. The Implementation of the Strategy on the test data

The system has been evaluated on three separate sets of data that differ both in content and in structure: compound toponyms, formal names of professions and queries from a search engine. Figure 3 depicts success evaluation for defined strategy. Results have confirmed that the developed strategy can be integrated in morphological query expansion mechanism for compounds and phrases which do not exist in the compounds dictionary.

The evaluation set with queries from search engine was selected from a log file of one of Serbian professional journals that deals with economic issues. The log file used thus gives a good insight in users' queries.

Some of the multi word queries from the log file represent simple lists of key words, for instance *izvoz, uvoz, Beograd, Srbija, 2002* 'import, export, Belgrade, Serbia, 2002'. It is not to be expected that the user would be interested for inflections of such a list as a whole. For many free phrases, especially those with fewer components, the structure was correctly detected and their inflected forms produced, e.g. *udio izvoza u domaćem proizvodu* 'export quota in domestic product'. As a by-product, the analysis of the log file detected some compounds that were not yet in the dictionary of compounds and which were subsequently added to it (the most frequent one being *kursna lista* 'the exchange rate list'). In order to be able to correctly inflect more free phrases some new inflectional transducers had been created.

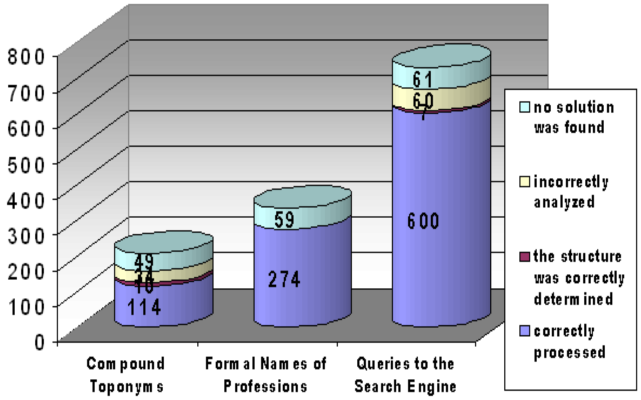


Fig. 3. The implementation of the Strategy on the evaluation data

III. Query Improvement

A. WS4LR

WS4LR handles simultaneously several types of resources, one of them being the system of morphological dictionaries of Serbian simple words and compounds in LADL format. Morphological dictionaries in the same format exist for many other languages, including French, English, Greek, Portuguese, Russian, Thai, Korean, Italian, Spanish, Norwegian, Arabic, German, Polish and Bulgarian.

The system enables concurrent manipulation of a set of dictionaries of lemmas, simple words (DELAS) or compounds (DELAC), distributed in several files. Working with dictionaries of word forms (DELAF, DELACF) type files is not directly supported since this type of files should in general be produced automatically from DELAS and DELAC by applying the appropriate transducers. The organization of dictionaries in separate files is important from the practical point of view since smaller files are easier to manipulate.

An important feature of this system is the ability of retrieving efficiently a subset of lemmas by matching the lemmas, their part of speech (PoS), inflectional class code, syntactic and semantic markers or their Boolean combination. For instance, one can look for all the dictionary entries starting or ending with a search string.

Another important resource handled by WS4LR is the Serbian Wordnet [8]. A Wordnet is composed of synsets, or sets of synonymous words representing a concept, with basic semantic relations between them forming a semantic network. Each synset word or "literal" is denoted by a "literal string" followed by a "sense tag" which represents the specific sense of the literal string in that synset, while interlingual index (ILI) enables the connection of the same concepts in different languages, a feature that can be used, among others, for cross-language information retrieval.

For expansion of queries with proper names WS4LR is using Prolex, a multilingual database of proper names which represents the implementation of an elaborate four-layered ontology of proper names [9] organized around a conceptual proper name that represents the same concept in different languages is used.

WS4LR also handles aligned texts. A pair of semantically equivalent texts in different languages, such as an original text and its translation, that are aligned on a structural level (paragraph, sentence, phrase, etc.) is known as an aligned text or bitext. The standard format for representing aligned texts is the Translation Memory eXchange format (TMX) that is XML-compliant [10].

WS4LR, written in C#, is organized in modules which perform different functions. A Component diagram (Fig. 4.) illustrates the pieces of software that make up the WS4LR system. The diagram on figure 4 demonstrates some components and their inter-relationships. The core of the system *WS4LR_Core* comprises four .Net libraries: *CommonRes.dll*, *NlpQuery.dll, VisualTMX.dll* and *WNDictAuto.dll.* A dependency relationship maps *NlpQuery.dll* to the handled lexical resources.

WS4LR_Core is used by two components: the stand-alone windows application *WS4LR.exe* and the web service *wsQueryExpand.asm.* Web application *WS4QE.asp* manages user query request, than uses web service in order to expand user query, submits the expanded query to Google search engine and finally presents retrieved result.

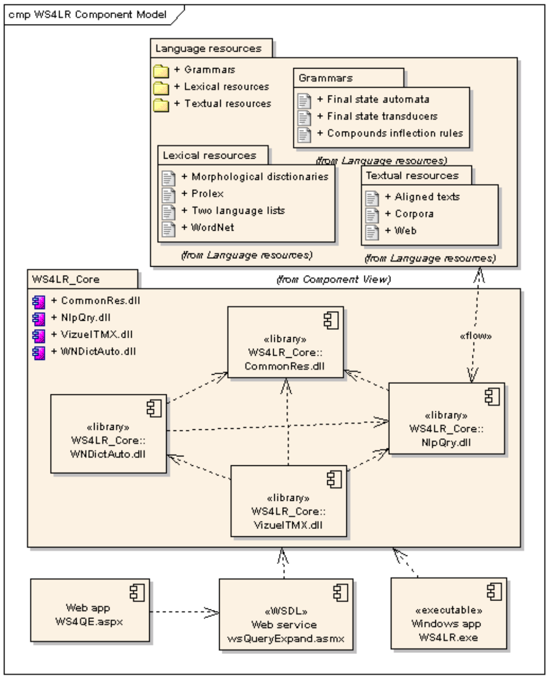

cmp WS4LR Component Model
Language resources
+ Grammars
+ Lexical resources
+ Textual resources
Grammars
+ Final state automata
+ Final state transducers
+ Compounds inflection rules
(from Language resources)
Lexical resources
+ Morphological disctionaries
+ Prolex
+ Two language lists
+ WordNet
(from Language resources)
Textual resources
+ Aligned texts
+ Corpora
+ Web
(from Language resources)
WS4LR_Core
(from Component View)
+ CommonRes.dll
+ NlpQry.dll
+ VizuelTMX.dll
+ WNDictAuto.dll
«library» WS4LR_Core:: CommonRes.dll
«flow»
«library» WS4LR_Core:: WNDictAuto.dll
«library» WS4LR_Core:: NlpQry.dll
«library» WS4LR_Core:: VizuelTMX.dll
Web app WS4QE.aspx
«WSDL» Web service wsQueryExpand.asmx
«executable» Windows app WS4LR.exe

Fig. 4. The components that make up the WS4LR system and their inter-relationships

B. Usage of Various Lexical Resources and Tools for Improvement of Queries

Selection of words chosen for a query, which are of paramount importance for the quality of results obtained by the query, can be substantially improved by using various lexical resources. Morphological dictionaries enable morphological expansion of the query, very important in highly inflective languages, such as Serbian. Wordnets and Prolex support semantic and multilingual expansion of the query.

The WS4LR system for query expansion allows the user to decide how his query will be expanded by choosing one or several of the offered options:

1. Alternate alphabet usage – for instance, the user can submit a keyword in Latin alphabet: *lekar opšte prakse* 'general practitioner' which will be expanded automatically by adding the keyword in Cyrillic: *лекар опште праксе*.

2. The inclusion of inflectional forms, for instance, *lekara opšte prakse, lekaru opšte prakse, lekarima opšte prakse, etc* with support of morphological dictionaries, inflectional transducers and the rule based procedure for Serbian.

3. The addition of synonyms – for instance, the synonym *lekar opšte medicine* 'GP' can be added to the keyword *lekar opšte prakse*. Synonyms are added on basis of the Serbian Wordnet (SWN). All other relations included in SWN can also be used for query expansion, for instance related hyponyms: *porodični lekar, kućni lekar, seoski lekar* 'family doctor, country doctor'.

4. The expansion of proper names using Prolex which offers to the user the option of adding proper name aliases, its synonyms, but also other proper names which are semantically related to the initial proper name through holonym and meronym relations. Thus a query with the word Meksiko 'Mexico' can be expanded with derivation *Meksikanac* '*Mexican man'*, *Meksikanka*, 'Mexican woman' but also with meronymes *Mexico-City* and *Puebla*.

5. The inflection of free phrases by predicting their syntactic structure. Presumption is that many free phrases used for search will have the same syntactic structure as a compound, and that the inflectional transducers for compounds that have already been developed can be applied to inflect them correctly. This type of expansion is implemented with the rule based system described in the second section of this paper. An example is the phrase *prosečna plata u Srbiji* 'average salary in Serbia' which, according to the dictionaries can be analyzed as a phrase of the form adjective+noun followed by any two words. In this particular case the rule 47 for NC_AXN4X is applied for query expansion.

6. The bilingual search – for instance, to the keyword *lekar opšte prakse* and its Serbian synonym keyword *lekar opšte medicine* a corresponding English set of synonyms can be added: {general practitioner, GP}. The bilingual search is, however, done separately and the results are presented in two columns.

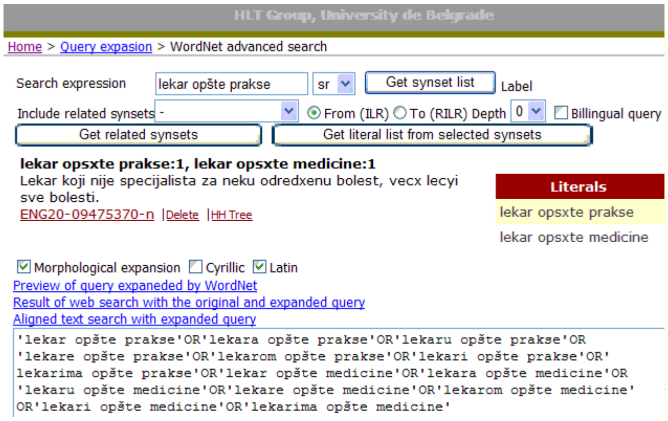
C. WS4QE

The developed web application receives the user query, and subsequently uses the local web service WS4QE to expand the query and forward it to the Google search engine using the Google AJAX Search API. Google AJAX Search API is a Java script library which enables the embedding of Google searches into personal web pages or web applications. This library is composed of simple web objects which perform "inline" search using numerous Google services (Web Search, Local Search, Video Search, Blog Search, News Search and Book Search).

The web service returns the required information in XML form, which is being received and converted to appropriate application structures (string, array, table, etc.). Some of the typical calls are: *getObliciLeme(lema)*, which retrieves all inflective forms of a lemma, *getSinonimiWN_WithFlex(lema)* which retrieves all wordnet synonyms with inflective forms, *getSinonimiWN_NoFlex(lema)* which retrieves all wordnet synonyms without inflective forms, *getProlexTable(rec, jezikSearch, Inflect, ExpandWith)* which retrieves all chosen proper name expansions according to the request specified by the user.

WS4QE also offers functions for aligned text manipulation and search with expanded queries, but some of WS4QE features related to query expansion will be illustrated in web search.

Query expansion is implemented with different possibilities and levels of detail, so the web user can choose from several options (from simple query expansion to complex wordnet advanced search). Figure 5 shows the page with the keyword *lekar opšte prakse* chosen as the initial search string. As semantic expansion was chosen, the appropriate synset was retrieved and synonym the *lekar opšte medicine* appeared in the list of words that can be used for composing the query. In this case morphological expansion was selected, and the query is further expanded only by including both chosen words in all inflected forms.


HLT Group, University de Belgrade
Home > Query expasion > WordNet advanced search
Get synset list
Label
Get related synsets
Get literal list from selected synsets
Literals
Preview of query expaneded by WordNet
Result of web search with the original and expanded query
Aligned text search with expanded query

Search expression: lekar opšte prakse | sr

Include related synsets: - | ● From (ILR) ○ To (RILR) Depth 0 | □ Billingual query

lekar opsxte prakse:1, lekar opsxte medicine:1
Lekar koji nije specijalista za neku odredxenu bolest, vecx lecyi sve bolesti.
ENG20-09475370-n |Delete |HH Tree

lekar opsxte prakse
lekar opsxte medicine

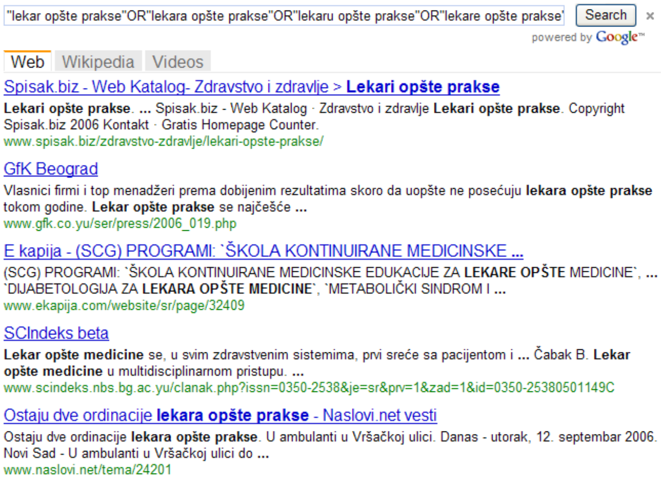
☑ Morphological expansion □ Cyrillic ☑ Latin

```
'lekar opšte prakse'OR'lekara opšte prakse'OR'lekaru opšte prakse'OR
'lekare opšte prakse'OR'lekarom opšte prakse'OR'lekari opšte prakse'OR'
lekarima opšte prakse'OR'lekar opšte medicine'OR'lekara opšte medicine'OR
'lekaru opšte medicine'OR'lekare opšte medicine'OR'lekarom opšte medicine'
OR'lekari opšte medicine'OR'lekarima opšte medicine'
```

Fig. 5. Morphological and semantic expansion of a query

The query, now composed of two Latin and two Cyrillic strings was then submitted by WS4QE to Google and, as a result, documents with different forms of both synonymous compounds were obtained. A thorough inspection of all documents was not performed, for obvious reasons, but it is safe to say that it is most unlikely that any of the documents obtained is irrelevant because all words used are specific in that they are neither homonymous nor polysemous. Part of the results of the expanded query is depicted in Figure 6.

For illustration of recall purposes, three query expansions where performed using the word *političko opredeljenje ‘political preference’* and all results were compared. First query expansion included semantic expansion with synonym *ideologija ‘ideology’*. The expanded query *"ideologija "OR" političko opredeljenje"* was then submitted by WS4QE to Google and, as a result, a total of 245,000 documents were obtained. The same query submitted directly to Google with only the initial string *političko opredeljenje* returned a total of 24,700. Thus the expanded query, without the morphological expansion, obtained almost ten times more documents. In the second case, semantic expansion remained and the query was improved additionally by including all words in Cyrillic alphabet. The result of the expanded query was a total of 320,000 documents. The expanded query once again remarkably increased the number of documents obtained. The third query was performed with morphological and semantic expansion, but the extension to Cyrillic alphabet was omitted. As a result 609,000 documents were obtained, which means that the recall has been extremely improved. Thus it can be concluded that a considerable increase of recall was obtained in all three examples.



"lekar opšte prakse"OR"lekara opšte prakse"OR"lekaru opšte prakse"OR"lekare opšte prakse'

Spisak.biz - Web Katalog- Zdravstvo i zdravlje > **Lekari opšte prakse**
Lekari opšte prakse. **...** Spisak.biz - Web Katalog · Zdravstvo i zdravlje **Lekari opšte prakse**. Copyright Spisak.biz 2006 Kontakt · Gratis Homepage Counter.
www.spisak.biz/zdravstvo-zdravlje/lekari-opste-prakse/

GfK Beograd
Vlasnici firmi i top menadžeri prema dobijenim rezultatima skoro da uopšte ne posećuju **lekara opšte prakse** tokom godine. **Lekar opšte prakse** se najčešće **...**
www.gfk.co.yu/ser/press/2006_019.php

E kapija - (SCG) PROGRAMI: `ŠKOLA KONTINUIRANE MEDICINSKE ...
(SCG) PROGRAMI: `ŠKOLA KONTINUIRANE MEDICINSKE EDUKACIJE ZA **LEKARE OPŠTE** MEDICINE`, **...** `DIJABETOLOGIJA ZA **LEKARA OPŠTE MEDICINE**`, `METABOLIČKI SINDROM I **...**
www.ekapija.com/website/sr/page/32409

SCIndeks beta
Lekar opšte medicine se, u svim zdravstvenim sistemima, prvi sreće sa pacijentom i **...** Čabak B. **Lekar opšte medicine** u multidisciplinarnom pristupu. **...**
www.scindeks.nbs.bg.ac.yu/clanak.php?issn=0350-2538&je=sr&prv=1&zad=1&id=0350-25380501149C

Ostaju dve ordinacije **lekara opšte prakse** - Naslovi.net vesti
Ostaju dve ordinacije **lekara opšte prakse**. U ambulanti u Vršačkoj ulici. Danas - utorak, 12. septembar 2006. Novi Sad - U ambulanti u Vršačkoj ulici do **...**
www.naslovi.net/tema/24201

Fig. 6. Results for expanded query for *lekar opšte prakse*

On the other side, speaking of precision, unexpanded query with compounds and phrases can obtain unrelated results. For example document with “*... srpske politike kroz istoriju* ***političkog*** *ekumenizma,... u kontekstu istorijskog* ***opredeljenja*** *za etiku, etičnost i karakternost, ...”* is obtained, but is not relevant, because the adjective *političkog* is related to the noun *ekumenizma* instead of *opredeljenja.*

References

[1] Krstev, C., Stanković, R., Vitas, D., Obradović, I. (2006). “WS4LR: A Workstation for Lexical Resources”. *In Proceedings of the 5th International Conference on Language Resources and Evaluation, LREC 2006*, Genoa, Italy, May 2006, pp. 1692-1697.

[2] Gelbukh, A., Sidorov G. "Approach to construction of automatic morphological analysis systems for inflective languages with little effort". *LNCS 2588*, 2003, pp. 215-220.

[3] Courtois, B., Silberztein, M. (eds.): *Dictionnaires électroniques du français. Langue française*. 87, Larousse, Paris, 1990.

[4] Krstev C.: *Processing of Serbian – Automata, Texts and Electronic Dictionarie*. Faculty of Philology, University of Belgrade, Belgrade, 2008.

[5] Savary, A., Krstev, C., Vitas, D.: “Inflectional non compositionality and variation of compounds in French, Polish and Serbian, and their automatic processing”. *Bulag - Bulletin de Linguistique Appliquée et Générale*. 32, 73-94, 2007.

[6] Krstev, C., Vitas, D., Savary, A.: “Prerequisites for a Comprehensive Dictionary of Serbian Compounds”. *In: Salakosi, T., Ginter, F., Pyysalo, S., Pahikkala, T. (eds.) FinTAL 2006. LNAI*, vol. 4139, pp. 552--564. Springer, Heidelberg, 2006.

[7] Krstev, C. Stanković, R., Vitas, D., Obradović, I..: “The Usage of Various Lexical Resources and Tools to Improve the Performance of Web Search Engines”. *In: 6th LREC International Conference on Language Resources and Evaluation*, Marrakech, Marocco, 2008.

[8] Krstev C., Pavlović-Lažetić G., Vitas D., Obradović I.: “Using Textual and Lexical Resources in Developing Serbian Wordnet.” *In Romanian Journal of Information Science and Technology*, Romanian Academy, Publishing House of the Romanian Academy, vol. 7, No. 1-2, pp. 147-161, (2004).

[9] Krstev, C., Vitas, D., Maurel, D., Tran, M. (2005). “Multilingual Ontology of Proper Names”. *In Proc. of Second Language & Technology Conference*, Poznań, Poland, April 21-23, Wydawnictwo Poznańskie Sp. z o.o, Poznań.

[10] TMX 1.4b specification, http://www.lisa.org/standards/tmx/tmx.html