Disambiguation of Serbian sentences with Unitex

Ivan Todorović
cs08it9, Lund University, Sweden
mirimiri66@gmail.com

January 16, 2009

Abstract

In this paper we describe a way to resolve word ambiguities in sentences in the Serbian language. For this purpose we use Unitex which is a tool for the analysis of texts in natural languages. It facilitates the creation of rules in a graphical environment, which makes it possible to eliminate inapplicable word meanings.

1 Introduction

Ambiguities, i.e. multiple meanings, in words occur frequently, but as human beings we are not always aware of it when participating in conversation or reading texts. The sense or words often falls natural from their context or prevalence. If they don't, we may have the opportunity to ask our interlocutor to elaborate or we could reconsider the phrase in another context. For example, the sentence *The girl moved the chair* would usually not be ambiguous to a human being, but some of the words in the sentence may have more than one meaning, as we will see below.

In automated analysis however, the above mentioned methods are limited or might not be viable, e.g. if automatically analysing a text. Other means of resolving ambiguities are necessary.

As an example of word ambiguity consider the word *chair*. It may, among others, have the following dictionary definitions:

1. *noun*, a separate seat for one person, typically with a back and four legs.
2. *noun*, the person in charge of a meeting or organisation.
3. *verb*, act as chairperson of or preside over (an organisation, meeting, or public event).

As a consequence, a sentence constructed, partly or in whole, of ambiguous words will itself also be ambiguous.

In this paper we will show how Unitex[1] can be used to resolve such ambiguities. As the following methods may be applicable to other languages, we will first consider the topic more generally and then follow up by focusing on sentences in Serbian. In section two we will give a brief description of the resources at hand. In section three will look into the grammar tools included with Unitex that are relevant to our topic as well as the construction of rules for resolving ambiguities. In section four we will show some examples of resolving ambiguities using rules. In section five we will comment briefly on work related to this paper. Section six provides a summary and outlines directions for future work.

2 Unitex

We will use the text analysis software Unitex to construct rules in order to achieve the desired results. Unitex, a corpus processing system developed at LADL (Laboratoire d'Automatique Documentaire et Linguistique), under the direction of its director, Maurice Gross, is a collection of tools and resources with a graphical user interface written in Java[2] and

[1] http://www-igm.univ-mlv.fr/~unitex/

[2] Sun Microsystems, Inc. http://www.sun.com/java/

external programs written in C/C++. Unitex is free software and is distributed under the LGPL[1] license with some exceptions (Paumier, 2008). The data distributed with it is under the LGPLLR[2] license. Unitex conforms to the Unicode[3] 3.0 standard.

We encountered some problems with the processing of the Serbian corpus and fell back to using Unitex version 1.2, although the latest version if Unitex was version 2.0. For out purpose, it proved to make no significant difference.

2.1 Resources

The Unitex resources consist of electronic dictionaries, grammars and corpora in a several languages.

The Serbian dictionary provided with Unitex is an extract of the Serbian morphological electronic dictionary developed by Krstev and Vitas at the University of Belgrade. It consists of approximately 0.7% of the available Serbian dictionary and covers the lexicon of the Serbian translation of Voltaire's *Candide*, which is also the Serbian corpus supplied with Unitex. In addition the first three chapters of *Candide* are supplied in a disambiguated, lemmatised and feature tagged form. Unitex also includes a few example disambiguation rules.

A state-of-the-art dictionary and grammar rule set has been developed by Krstev and Vitas but is not freely available.

Further details about the Serbian resources and other features of Unitex may be found in Vitas et al. (2003), Krstev and Vitas (2005), Obradović and Stanković (2008) and the Unitex 2.0 Users Manual (Paumier, 2008).

3 Grammars

3.1 Unitex grammars

Unitex grammars are based on *recursive transition networks* (RTN), also called *syntax diagrams*, which itself is an extension of context-free grammars. Its formalism is related to *finite state automata* (FSA), which is suitable for linguistic problems like morphology and syntax (Paumier, 2008).

To give some examples, consider that in context-free grammars every production rule is on the form:

$$V \rightarrow w$$

where V is a single nonterminal symbol, and w is a string of terminals and/or nonterminals (possibly empty, indicated by ε). For example, the following grammar matches any number of a characters.

$$S \rightarrow aS$$
$$S \rightarrow \varepsilon$$

RTN extends this by allowing the right side of a rule to be not only a sequence of symbols, but also to be a *regular expression*. The following rule, in RTN grammar, has the same matching characteristics as the previous two rules:

$$S \rightarrow a*$$

Unitex however, extends RTN, by enabling a grammar to produce output. This in effect, allows Unitex grammars to function as *transducers*, a term derived from the field of FSA.

3.2 Graphic representation

In Unitex grammars are represented by graphs, in a form similar to finite state machine diagrams, that can be edited by the user. The arrow symbol represents the *initial state* and the round symbol containing a square represents the *final state* (*Figure 1*). The grammar only recognises expressions that are matched along the paths between initial and final states. *Figure 1* shows an example of a graphical representation of the sentence *The girl moved the chair*.

Inside a box is the word *lemma* or *canonical form*. If the word is inflected, the inflected form is also shown on top.

The codes underneath each box denotes the part-of-speech (POS) tags (Nugues, 2006), followed by features, e.g. grammatical, inflectional and semantic codes. For example, the last word in the sentence, *chair*, has the following features:

[1] GNU Lesser General Public License
http://www.gnu.org/licenses/lgpl.html

[2] Lesser General Public License for Linguistic Resources
http://infolingu.univ-mlv.fr/DonneesLinguistiques/Lexiques-Grammaires/lgpllr.html

[3] Unicode Consortium
http://www.unicode.org

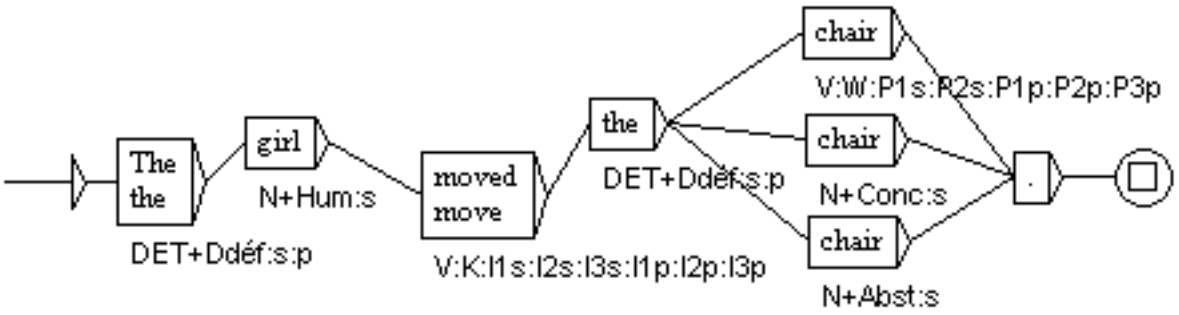

Figure 1: *The girl moved the chair.*

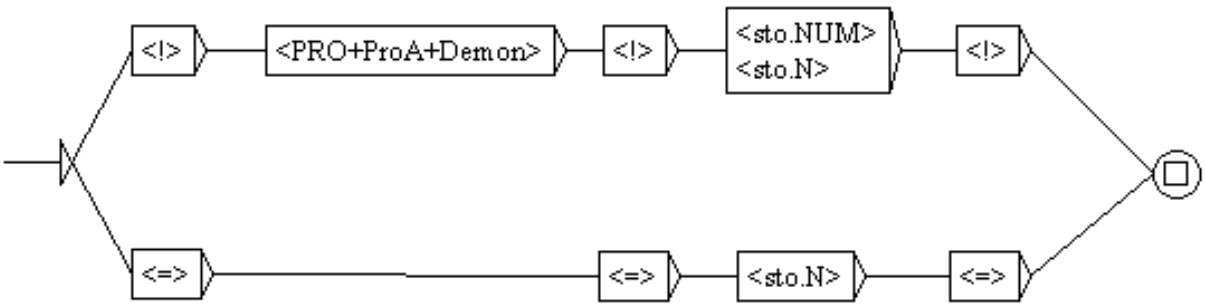

Figure 2: Elimination of the POS tag numeral for the word *sto* when it follows a pronoun. *4.1 [3]*

V:W:P1s:P2s:P1p:P2p:P3p

which denotes that it is a verb (*V*), may be in infinitive form (*W*) or present tense, (e.g. *P1s*, present tense, first person and singular).

3.3 ELAG grammars

In order to remove ambiguities, Unitex allows for the use of ELAG grammars (Laporte and Monceaux, 1998).

ELAG grammars have a different syntax compared to the preceding grammars. They consists of two parts, an *if* part and a *then* part. These grammars work as follows. If a path in the *if* part is recognised, then it must be recognised by the *then* part of the grammar. Otherwise the grammar will be withdrawn from the text automaton.

The *if* part is divided into two parts which are delimited by three boxes containing the <!> symbol. Likewise, the *then* part is divided in the same way using the <=> symbol.

Figure 2 shows an example of an ELAG grammar. In this example, if the Serbian word *sto*, which can be a numeral or a noun, is preceded by a pronoun (*PRO*) as an adjective pronoun (*ProA*) or a demonstrative pronoun (*Demon*) then the sequence is recognised by the *if* part. It must therefore be recognised by the *then* part. In this case, the *then* part eliminates the numeral sense of the word *sto*.

3.4 ELAG delimiters

In the example in *Figure 2*, the word *sto* in both parts of the ELAG grammar are aligned. By having three delimiters, instead of two, Unitex permits sequences of words to be non-aligned, which allow for broader matching possibilities. This feature is known as *synchronisation point*.

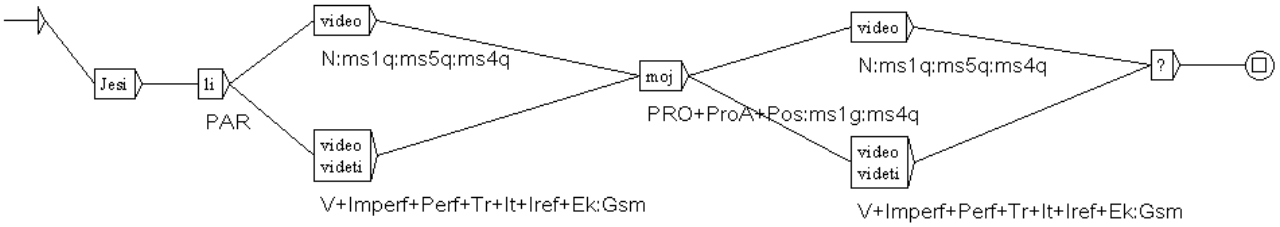

Figure 3: *Jesi li video moj video?*

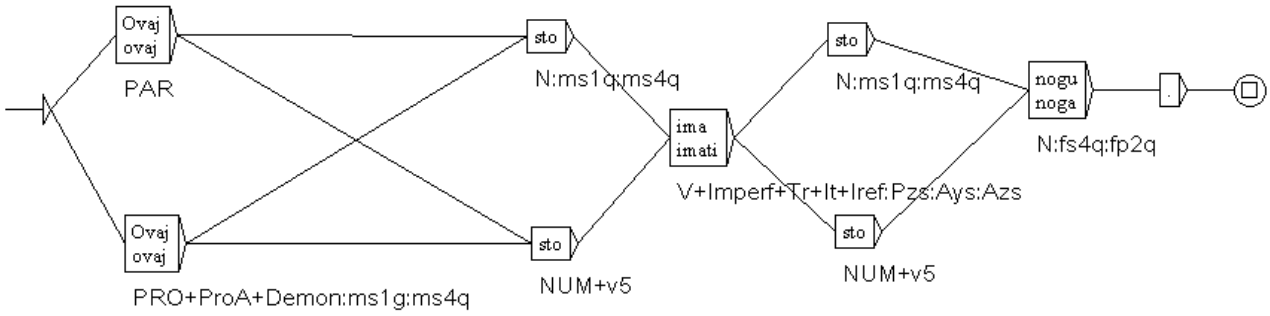

Figure 4: *Ovaj sto ima sto nogu.*

4 Disambiguation examples

We will now show how ELAG grammars can be used to disambiguate the following two Serbian sentences:

1. *Jesi li video moj video?*
   have you PAR saw.V my.PRO video.N
   (*Have you seen my video?*)
2. *Ovaj sto ima sto nogu.*
   this.PRO table.N has.V one hundred.NUM legs.N
   (*This table has one hundred legs.*)

Each sentence is represented in Unitex by an automaton whose paths represent all possible interpretations, as can be seen in *Figures 3* and *4*. For example, in *Figure 3*, the word *video* is interpreted as both a noun and a verb.

4.1 Disambiguation rules

In a tentative attempt to disambiguate our sentences we created the following rules:

1. *An* adjective *or* demonstrative pronoun *may not follow a* noun. (Figure 5)
2. *If a* possessive pronoun *is followed by a* verb *or a* noun*, discard the* verb *tag.* (Figure 6)
3. *If an* adjective *or* demonstrative pronoun *is followed by the word* sto *which has both the POS tags* noun *and* numeral*, discard the* numeral *tag.* (Figure 2)
4. *The word* ovaj *with the* particle *POS tag must be followed by a comma.* (Figure 7)

Applying rules [1] and [2] to the sentence *Jesi li video moj video?* produces the result in *Figure 8*.

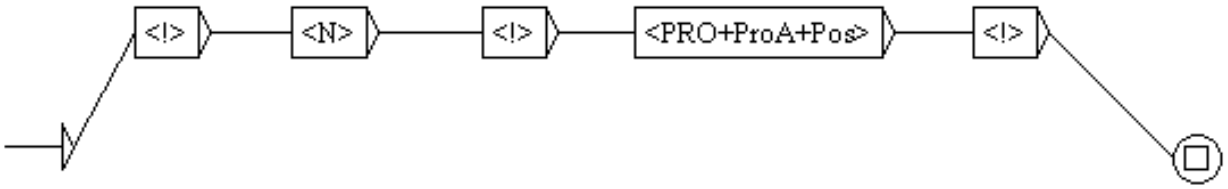

Figure 5: An adjective or demonstrative pronoun may not follow a noun. *4.1 [1]*

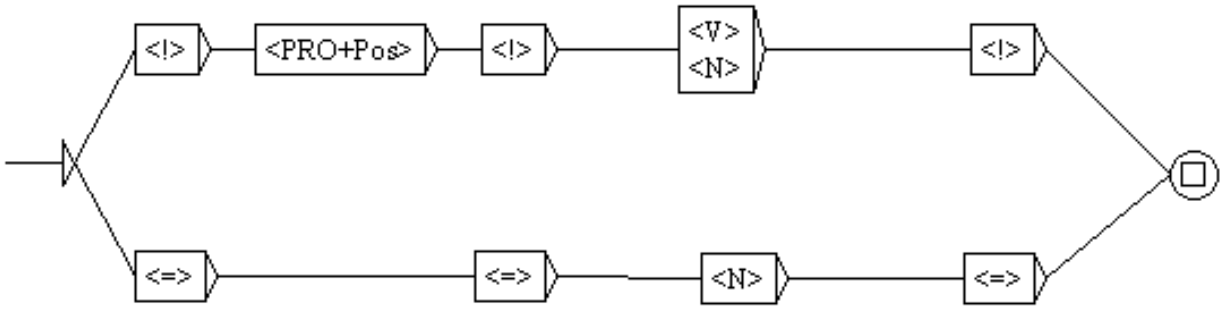

Figure 6: If a possessive pronoun is followed by a verb or a noun, discard the verb. *4.1 [2]*

Correspondingly, applying rules [3] and [4] to the sentence *Ovaj sto ima sto nogu.* produces the result in *Figure 9*.

In the later case the sentence is not completely disambiguated. In fact, while the sentence only makes sense in the case where the POS tag of *sto* is a numeral, the rules used are not suitable for the second instance of the word *sto*. An additional rule is required for the word *sto* to resolve the last ambiguity.

5 Related work

Krstev and Vitas wrote an introductory text that proposes different methods for finding and eliminating invalid paths in Unitex sentence graphs, so called *false ambiguities*.

Obradović and Stanković discusses software tools available for working with Serbian texts, in particular the refinement of Serbian dictionaries and texts that have previously been aligned[1] (Obradović and Stanković, 2008).

6 Summary

The purpose of the ELAG rules created in this paper is to demonstrate some possibilities of Unitex to resolve ambiguities. It is likely that some of these rules are overly constraining.

These examples show how one could go about starting to build an extensive set of ELAG rules, thereby providing the means for the disambiguation

[1] Alignment refers to splitting up parallel texts, into segments of words, sentences or paragraphs, with the goal of connecting equivalent segments.

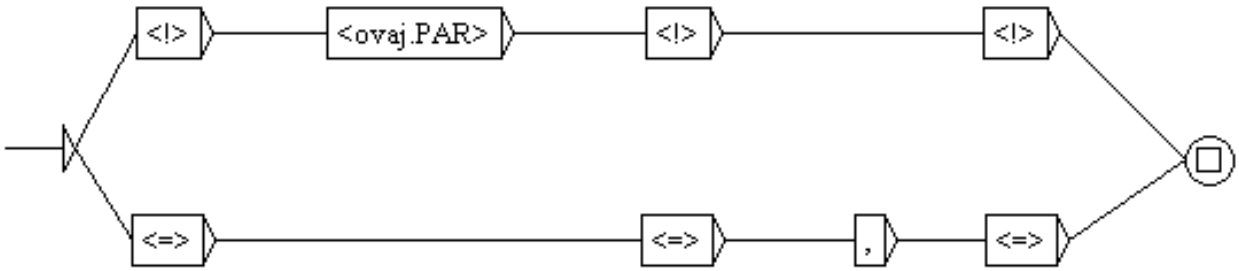

Figure 7: The word *ovaj* with the POS tag particle must be followed by a comma. *4.1 [4]*

of an increasing number of sentences, similar to the work done by Krstev and Vitas.

A future project could be to compare the results from applying an extended number of rules to the Unitex supplied corpus and then compare that to the disambiguated and annotated corpus, with the goal of comparing the success rate with other disambiguation methods.

7 Acknowledgements

I would like to thank Pierre Nugues and Richard Johansson who offered valuable input and helpful advice in the discussions on the subject matter and the contents of this paper.

References

Cvetana Krstev. *(Serbian dictionary for Unitex)*. Faculty of Philology, University of Belgrade, Studentski trg 3, 11000 Belgrade, Serbia. `http://www.matf.bg.ac.yu/~cvetana`.

Cvetana Krstev and Duško Vitas. *How to Find the Right Path? (On the Morphological Disambiguation of Sentence in Serbian)*. University of Belgrade, Serbia.

Cvetana Krstev and Duško Vitas. 2005. *Corpus and Lexicon - Mutual Incompleteness*. Proceedings of the Corpus Linguistics Conference, 14–17 July 2005, Birmingham. ISSN 1747-9398. `http://www.corpus.bham.ac.uk/PCLC/`.

Duško Vitas. *(Serbian dictionary for Unitex)*. Faculty of Mathematics, University of Belgrade, Studentski trg 16, 11000 Belgrade, Serbia. `http://www.matf.bg.ac.yu/~vitas`.

Duško Vitas, Cvetana Krstev, Ivan Obradović, Ljubomir Popović and Gordana Pavlović-Lažetić. 2003. *Processing Serbian Written Texts: An Overview of Resources and Basic Tools*. Workshop on Balkan Language Resources and Tools, 21 November 2003, Thessaloniki, Greece, pp. 97–104.

Eric Laporte and Anne Monceaux. 1998. *Elimination of Lexical Ambiguities by Grammars: The ELAG System*. Lingvisticæ Investigationes. John Benjamins Publishing Company, Amsterdam-Philadelphia, pp. 341–367.

Ivan Obradović and Ranka Stanković 2008. *Software Tools for Serbian Lexical Resources*. Faculty of Mining and Geology, University of Belgrade, Serbia.

Pierre M. Nugues. 2006. *An Introduction to Language Processing with Perl and Prolog*. Springer-Verlag, Berlin, Germany.

Sébastien Paumier. October 2008. *Unitex 2.0 Users Manual*. `http://www-igm.univ-mlv.fr/~unitex/UnitexManual2.0.pdf`. Institut Gaspard-Monge, University of Paris-Est Marne-la-Vallée, France.

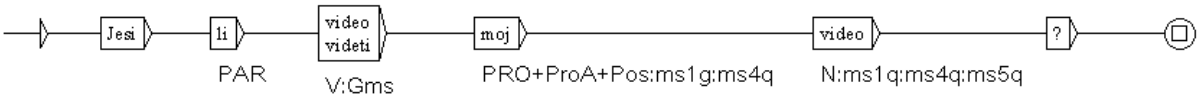

Figure 8: The result after applying disambiguation rules [1] and [2] to the sentence *Jesi li video moj video?*

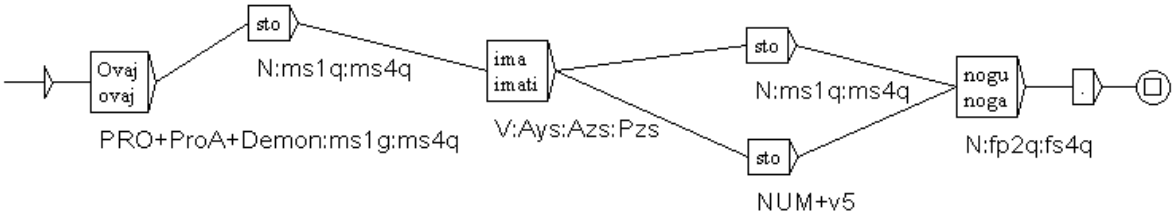

Figure 9: The result after applying disambiguation rules [3] and [4] to the sentence *Ovaj sto ima sto nogu*.