LINGUA - a robust architecture for text processing and anaphora resolution in Bulgarian

Hristo Tanev[1]
Department of Computer Science
University of Plovdiv
4000 Plovdiv
Bulgaria
Chritan@pu.acad.bg

Ruslan Mitkov
School of Humanities, Languages and Social Studies
University of Wolverhampton
Wolverhampton WV1 1SB
United Kingdom
R.Mitkov@wlv.ac.uk

Abstract

This paper describes LINGUA – an architecture for text processing in Bulgarian, and focuses on its anaphora resolution component. First, the pre-processing modules for tokenisation, sentence splitting, paragraph segmentation, part-of-speech tagging, clause chunking and noun phrase extraction are outlined. Next, the paper proceeds to describe in more detail the anaphora resolution module. Evaluation results are reported for each processing task.

1. Introduction

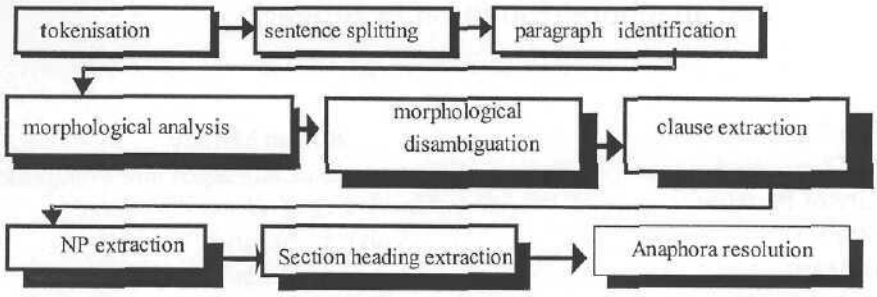
The state of the art of today's full parsing and knowledge-based automatic analysis still falls short of providing a reliable processing framework for robust, real-world applications such as automatic abstracting or information extraction. The problem is especially acute for languages which do not benefit from a wide range of processing programs such as Bulgarian. There have been various projects which address different aspects of the automatic analysis in Bulgarian such as morphological analysis (Krushkov 1997), (Simov et al. 1992), morphological disambiguation (Pascaleva 1995) and parsing (Avgustinova et al. 1989) but no previous work has pursued the development of a knowledge-poor, robust processing environment. This paper reports the development and implementation of a robust architecture for language processing in Bulgarian referred to as LINGUA, which includes modules for POS tagging, sentence splitting, clause segmentation, NP extraction, and anaphora resolution. Our text processing framework builds on the basis of considerably shallower linguistic analysis of the input, thus trading off depth of interpretation for breadth of coverage and workable, robust solution.

2. LINGUA – an architecture for language processing in Bulgarian

LINGUA is a text processing framework for Bulgarian which automatically performs tokenisation, sentence splitting, part-of-speech tagging, NP parsing, clause segmentation, section-heading identification and resolution for third person personal pronouns (Figure 1).

All modules of LINGUA are original and purpose-built, except for the module for morphological analysis which uses Krushkov's morphological analyser BULMORPH (1997). The anaphora resolver is an adaptation for Bulgarian of Mitkov's knowledge-poor pronoun resolution approach (1998).

Figure 1: general structure of LINGUA



2.1 Text segmentation: tokenisation, sentence splitting and paragraph identification

The first stage of processing is the segmentation of text in terms of tokens, sentences and paragraphs. LINGUA operates within an input window of 30 tokens, applying rules for token synthesis, sentence splitting and paragraph identification.

2.1.1 Tokenisation and token stapling

Tokens identified from the input text serve as input to the token stapler. The token stapler forms more complex tokens on the basis of a token grammar. With a view to improving tokenisation, a list of abbreviations has been incorporated into LINGUA.[2]

2.1.2 Sentence splitting

LINGUA's sentence splitter operates to identify sentence boundaries on the basis of 9 main end-of-sentence rules and makes use of a list of abbreviations. Some of the rules consist of several finer sub-rules. The evaluation of the performance of the sentence splitter on a text of 190 sentences reports a precision of 92% and a recall of 99%.

The sentence splitter captures and distinguishes ambiguous cases such as enumerated lists, abbreviations followed by names, sentences ending without a full stop or tokens containing a full stop. Abbreviated names such as "J.S.Simpson" are filtered by special constraints. The sentence splitting and tokenising rules were adapted for English. The resulting sentence splitter was then employed for identifying sentence boundaries in the Wolverhampton Corpus of Business English project.

2.2.2 Paragraph identification

Paragraph identification is based on heuristics such as cue words, orthography and typographical markers. The precision of the paragraph splitter is about 94% and the recall is 98% (Table 3).

2.3 Morphological analysis and part of speech tagging

2.3.1 Morphological analysis

Every token is analysed morphologically. First, precise analysis is performed by the BULMORPH morphological analyser (Krushkov 1997). If precise analysis fails, BULMORPH applies its own procedures for guessing unknown words.

2.3.2 Morphological disambiguation

Morphological disambiguation is performed through robust procedures, which take into account the context of the word and choose only one morphological hypothesis. We used 33 hand-crafted rules for disambiguation. Since large corpora in Bulgarian are not widely available, the development of a corpus-based probabilistic tagger was an unrealistic goal for us. However, as some studies suggest (Voutilainen 1995), the precision of rule-based taggers may exceed that of the probabilistic ones.

Typical examples of lexical ambiguity that LINGUA can solve are син (son) and син (blue), човека (men - plural) and човека (the man - object definite form), as well as работи (works – verb 3 person singular) and работи (works – plural noun). We did not handle the ambiguity arising from the different verb tenses (e.g. present, past)

We used a POS tagging model which considers the context of the word spanning no more than 5 words - up to 2 words before and up to 2 words after the ambiguous word (Only one rule, which solves the construction "*Ако...то...*"("If..then..")[3] makes use of a wider context). A rule is applied if only one of the words in the window is ambiguous.

2.4 NP extraction

This module makes use of noun phrase rules to identify noun phrases in the text. Phrases containing smaller NPs can be analysed (phrases with no limitation on the depth of nested phrases can be parsed), but at the moment the grammar has no provision for NPs containing nested clauses. The NP grammar can also analyse phrases which contain left modifiers, such as adjectives, demonstrative pronouns, numerals etc, prepositional phrases such as *Бащата на Георги* (literally "The father of George") and conjunctions such as *Петър и Иван* ("Peter and Ivan").

The NP extractor is based on a simple unification grammar for NP and AP in Bulgarian which was specially developed for this project. The recall of NP extraction, measured against 352 NPs from software manuals, was 77% and the precision - 63.5%

2.5 Clause chunking

A heuristics-based algorithm for identification of clause boundaries is implemented in LINGUA. The complex sentences are split into clauses in the following way. First, finite verb forms are identified working on the assumption that each clause is built around a finite verb. Next, the boundaries between clauses are identified. This process is divided into several steps. To start with, demarcating expressions such as *за да* (in order to), тъй като (since) as well as к-words (Bulgarian equivalent of wh-words) are located. If no such expressions are found between finite verbs, then conjunctions, adverbs, punctuation signs are searched for.

2.6 Section heading identification

This is a typical processing task for the domain of technical manuals. LINGUA uses two heuristics for recognising section captions: (a) A single sentence in a paragraph without finite verb forms is a section heading (b) A single sentence in a paragraph with capital letters is a section heading. Heuristic (a) covers 90% of the headings in technical-texts.

3. Anaphora resolution in Bulgarian

3.1 Adaptation of Mitkov's knowledge-poor approach for Bulgarian

The anaphora resolution module is implemented as the last stage of the language processing architecture (Figure 1). This module resolves third-person personal pronouns and is an adaptation of Mitkov's robust, knowledge-poor multilingual approach (Mitkov 1998, Mitkov 2000a) whose latest implementation by R. Evans is referred to as MARS[4] (Orasan, Evans and Mitkov 2000). MARS does not make use of parsing, syntactic or semantic constraints; nor does it employ any form of non-linguistic knowledge. Instead, the approach relies on the efficiency of sentence splitting, part-of-speech tagging, noun phrase identification and the high performance of the antecedent indicators; knowledge is limited to a small noun phrase grammar, a list of (indicating) verbs and a set of antecedent indicators.

The core of the approach lies in activating the *antecedent indicators* after filtering candidates (from the current and two preceding sentences) on the basis of gender and number agreement and the candidate with the highest composite score is proposed as antecedent.[5] Before that, the text is pre-processed by a sentence splitter which determines the sentence boundaries, a part-of-speech tagger which identifies the parts of speech and a simple phrasal grammar which detects the noun phrases. In the case of complex sentences, heuristic 'clause identification' rules track the clause boundaries.

LINGUA performs the pre-processing, needed as an input to the anaphora resolution algorithm: sentence, paragraph and clause splitters, NP grammar, part-of-speech tagger, section heading identification heuristics. Since one of the indicators that Mitkov's approach uses is term preference, we manually developed[6] a small term bank containing 80 terms from the domains of programming languages, word processing, computer hardware and operating systems.[7] This bank additionally featured 240 phrases containing these terms.

The antecedent indicators employed in MARS are classified as *boosting* (such indicators when pointing to a candidate, reward it with a bonus since there is a good probability of it being the antecedent) or *impeding* (such indicators penalise a candidate since it does not appear to have high chances of being the antecedent). The majority of indicators are genre-independent and are related to coherence phenomena (such as salience and distance) or to structural matches, whereas others are genre-specific (e.g. *term preference, immediate reference, sequential instructions*). Most of the indicators have been adopted in LINGUA without modification from the original English version (see Mitkov 1998 for more details). However, we have added 3 new indicators for Bulgarian: selectional restriction pattern, adjectival NPs and name preference.

The boosting indicators are

First Noun Phrases: A score of +1 is assigned to the first NP in a sentence, since it is deemed to be a good candidate for the antecedent.

Indicating verbs: A score of +1 is assigned to those NPs immediately following the verb which is a member of a previously defined set such as *discuss*, *present*, *summarise* etc.

Lexical reiteration: A score of +2 is assigned those NPs repeated twice or more in the paragraph in which the pronoun appears, a score of +1 is assigned to those NP, repeated once in the paragraph.

Section heading preference: A score of +1 is assigned to those NPs that also appear in the heading of the section.

Collocation match: A score of +2 is assigned to those NPs that have an identical collocation pattern to the pronoun.

Immediate reference: A score of +2 is assigned to those NPs appearing in constructions of the form "...V_1NP...<CB> V_2 it", where <CB> is a clause boundary.

Sequential instructions: A score of +2 is applied to NPs in the NP_1 position of constructions of the form: "To V_1 NP_1..... To V_2 it.."

Term preference: a score of +1 is applied to those NPs identified as representing domain terms.

Selectional restriction pattern: a score of +2 is applied to noun phrases occurring in collocation with the verb preceding or following the anaphor. This preference is different from the *collocation match* preference in that in operates on a wider range of 'selectional restriction patterns' associated with a specific verb[8] and not on exact lexical matching. If the verb preceding or following the anaphor is identified to be in a legitimate collocation with a certain candidate for antecedent, that candidate is boosted accordingly. As an illustration, assume that 'Delete file' has been identified as a legitimate collocation being a frequent expression in a domain specific corpus and consider the example 'Make sure you save *the file* in the new directory. You can now delete *it*.' Whereas the 'standard' *collocation match* will not be activated here, the *selectional restriction pattern* will identify *delete file* as an acceptable construction and will reward the candidate *the file*.

Adjectival NP: a score of +1 is applied to NPs which contain adjectives modifying the head. Empirical analysis shows that Bulgarian constructions of that type are more salient than NPs consisting simply of a noun.

Name preference: a score +2 is applied to names of entities (person, organisation, product names).

The impeding indicator is *Prepositional Noun Phrases*: NPs appearing in prepositional phrases are assigned a score of –1.

Two indicators, *Referential distance* and *Indefiniteness* may increase or decrease a candidate's score. *Referential distance* gives scores of +2 and +1 for the NPs in the same and in the previous sentence respectively, and –1 for the NPs two sentences back. *Indefiniteness* assigns a score of –1 to indefinite NPs, 0 to the definite (not full article) and +1 to these which are definite, containing the definite 'full' article in Bulgarian.

3.2 Evaluation of the performance of the anaphora resolution module

The precision of anaphora resolution measured on corpus of software manuals containing 337 anaphors, is 72.6%. Given that the anaphora resolution system operates in a fully automatic mode, this result could be considered very satisfactory. It should be noted that some of the errors arise from inaccuracy of the pre-processing modules such as clause segmentation and NP extraction (see Table 3).

We also evaluated the anaphora resolution system in the genre of tourist texts. As expected, the success rate dropped to 63.7% which, however, can still be regarded as a very good result, given the fact that neither manual pre-editing of the input text, nor any post-editing of the output of the pre-processing tools were undertaken. The main reason for the decline of performance is that some of the original indicators such as term preference, immediate reference and sequential instructions of the knowledge-poor approach, are genre specific.

The software manuals corpus featured 221 anaphoric third person pronouns, whereas the tourist text consisted of 116 such pronouns. For our evaluation we used the measures *success rate* and *critical success rate* (Mitkov 2000b). Success rate is the ratio SR = AC/A, where AC is the number of correctly resolved and A is the number of all anaphors. We also compared our approach with the typical baseline model *Baseline most recent* which takes as antecedent the most recent NP matching the anaphor in gender and number. The results are shown in the Table 1.

Table 1: success rates of LINGUA and a baseline model

Text		Pronouns	Weight set		
			Standard	Optimised	Baseline most recent
Software manuals	Success rate	221	75.0%	78.8%	58.0%
	Critical succ. rate		70.0%	73.0%	54.0%
	Non trivial succ.rate		70.0%	78.8%	58.0%
Tourist guides	Success rate	116	68.1%	69.8%	65.0%
	Critical succ. rate		63.3%	64.4%	58.8%
	Non trivial succ.rate		67.2%	69.0%	65.0%
All texts	Success rate	337	72.6%	75.7%	60.4%
	Critical succ. rate		67.7%	70.0%	55.7%
	Non trivial succ.rate		72.3%	75.4%	60.4%

These results show that the performance of LINGUA in anaphora resolution is comparable to that of MARS (Orasan, Evans and Mitkov 2000). An optimised version[9] of the indicator weights scored a success rate of 69,8% on the tourist guide texts, thus yielding an improvement of 6,1%.

The performance of LINGUA on software manuals shows clear superiority over the baseline model (by 17%).

3.3 Evaluation of the antecedent indicators

Our evaluation also covered the measures *decision power* and *indispensability* which characterise the separate factors/indicators employed in the algorithm, as opposed to success rate and critical success rate that relate to the performance of the algorithm or system as a whole.

Decision power is a measure of the influence of each factor (in our case indicator) on the final decision, its ability to "impose" its preference in line with, or contrary to the preference of the remaining indicators (Mitkov 2000b). The decision power (DP_K) of a boosting indicator K is defined in the following way:

$$DP_K = \frac{SI_K}{A_K} \times 100\%$$

where SI_K is the number of successful antecedent identifications (resolutions) when this indicator is applied and A_K is the number of applications of this indicator.

For the penalising indicators *prepositional noun phrase* and *indefiniteness* we calculate this figure as

$$DP_K = \frac{UI_K}{A_K} \text{ x } 100\%$$

where UA_K is the number of unsuccessful antecedent identifications and A_K the number of applications of this indicator.

Table 2: Decision power and indispensability values of the indicators used

	Tourist		Software	
	DP	Indisp.	DP	Indisp.
First Noun Phrases	34.92%	0.00%	21.62%	0.00%
Indicating Verb	75.00%	0.00%	90.00%	0.00%
Lexical Reiteration	24.00%	0.00%	32.65%	2.20%
Section Heading Preference	35.00%	3.10%	23.07%	2.20%
Collocation Pattern Preference	33.33%	0.00%	60.00%	0.00%
Adjectival Noun Phrases	33.59%	2.10%	33.33%	2.20%
Definiteness	73.25%	3.80%	81.44%	0.00%
Non-prepositional Noun Phrases	73.73%	4.70%	71.05%	0.00%
Referential Distance	45.55%	5.20%	51.36%	26.00%
Names Preference	30.76%	0.00%	13.26%	0.00%
Collocation Term Pairs	N/A	N/A	100.00%	0.00%
Term Preference	N/A	N/A	34.40%	0.00%
Immediate Reference	N/A	N/A	62.5%	6.81%

Indispensability shows how vital, indispensable the presence of specific factor is (Mitkov 2000b). We define indispensability for a given indicator K as

$$Ind_K = \frac{SR - SR_{-K}}{SR} \text{ x } 100\%$$

where SR_{-K} is the success rate obtained when the indicator K is excluded, and SR is the success rate (with all the indicators on). In other words, indispensability is a measure for the non-absolute, relative contribution of this indicator to the "collective efforts" of all indicators: this measure shows how much the approach would lose out if the specific indicator were removed. It should be noted that being indispensable does not mean decision-powerful, confident and vice-versa.

Since *indispensability* and *decision power* were calculated on tourist texts and software manuals, genre specific indicators such as term preference, sequential instructions and immediate reference scored 0% on tourist guides (Table 2)

Referential distance, *immediate reference* and *non-prepositional noun phrases* indicators appear to have the highest value of indispensability, similar to the English version of the approach (Orasan, Evans, Mitkov 2000).

4. Conclusion and summary

This paper outlines the development of the first robust and shallow text processing framework in Bulgarian LINGUA which includes modules for tokenisation, sentence splitting, paragraph segmentation, part-of-speech tagging, clause chunking, noun phrases extraction and anaphora resolution (Figure 1). Apart from the module on pronoun resolution which was adapted from Mitkov's knowledge-poor approach for English and the incorporation of BULMORPH in the part-of-speech tagger, all modules were specially built for LINGUA. The evaluation shows promising results for

each of the modules (Table 3). Comparison with other pre-processing tools in Bulgarian was not possible due to the unavailability of evaluation results for these. The comparison with the original anaphora resolution method for English suggests that the Bulgarian version as adapted for LINGUA performs comparably well.

Table 3: Summary of the results

Language processing module	*Precision*	*Recall*	*Evaluation data*
sentence splitter	92.0%	99.0%	190 sentences
paragraph splitter	94.0%	98.0%	268 paragraphs
clause chunker	93.5%	93.1%	232 clauses
POS tagger	95.0%	-	303 POS tags
NP parser	63.5%	77.0%	352 noun phrases
anaphora resolution	71.1%	-	221 anaphors

Acknowledgments

We would like to thank Richard Evans who commented on an earlier version of this paper. Our thanks go also to Constantin Orasan who helped us to employ genetic algorithms in the optimisation phase.

NOTES

[1] Most of the work by the first author was funded by the University of Wolverhampton and was conducted while he was guest researcher of the Research Group of Computational Linguistics, School of Humanities, Languages and Social Studies, University of Wolverhampton.

[2] This list of abbreviations enhances the performance of the sentence splitter as well (see below).

[3] In Bulgarian the word *mo ("then")* is a homograph which can also be the 3-rd person pronoun *it*.

[4] MARS stands for Mitkov's Anaphora Resolution System.

[5] For a detailed procedure how candidates are handled in the event of a tie, see (Mitkov 2000).

[6] This was done for experimental purposes. In future applications, we envisage the incorporation of automatic term extraction techniques.

[7] Note that MARS obtains terms automatically using TF.IDF.

[8] At the moment these patterns are extracted from a list of frequent expressions involving the verb and domain terms in a purpose-built term bank but in generally they are automatically collected from large domain-specific corpora.

[9] The optimisation made use of genetic algorithms in a manner similar to that described in (Orasan, Evans and Mitkov 2000).

References

Avgustinova, T., K. Oliva and E. Paskaleva. 1989. "An HPSG-based Parser for Bulgarian". *International Seminar on Machine Translation "Computer and Translation 89"*, 10-22. Moscow, Russia.

Krushkov, H. 1997. *Modelling and building of machine dictionaries and morphological processors* (In Bulgarian), University of Plovdiv, Ph.D. Thesis, 1997

Mitkov, R. 1998. "Robust pronoun resolution with limited knowledge". *Proceedings of the 18.th International Conference on Computational Linguistics (COLING'98)/ACL'98 Conference*, 869-875. Montreal, Canada.

Mitkov, R. 2000a. "Multilingual anaphora resolution". *Machine Translation*. (forthcoming)

Mitkov R. 2000b. "Towards more consistent and comprehensive evaluation in anaphora resolution". *Proceedings of LREC'2000*, Athens, Greece.

Simov, K., Paskaleva, E., Damova, M. And M. Slavcheva. 1992 " Morpho-Assistant - A Knowledge Based System for Bulgarian Morphology" *Proceedings of the Third Conference on Applied Natural Language Processing* (Demo description), Trento, Italy

Orasan C, Evans R. and Mitkov R. 2000. "Enhancing Preference-Based Anaphora Resolution with Genetic Algorithms", *Proceedings of NLP'2000*, Patras, Greece

Voutilainen, A. 1995. "A syntax-based part-of-speech tagger". Proceedings of the 7th conference of the European Chapter of the Association for Computational Linguistics (EACL'95), 157-164. Dublin, Ireland.