The proper place of men and machines in language technology Processing Russian without any linguistic knowledge

Serge Sharoff (s.sharoff@leeds.ac.uk) University of Leeds, UK
Joakim Nivre (joakim.nivre@lingfil.uu.sv) Uppsala University, Sweden

Abstract

The paper describes several experiments aimed at designing tools for processing Russian texts, namely for Part-Of-Speech tagging, lemmatisation and syntactic parsing, exploiting exclusively statistical approaches without coding any linguistic rules specifically for Russian. While not claiming any new ground for machine learning research, the results demonstrate the possibility to create state-of-the-art tools for Russian in very short time using only machine learning and no hard-coded linguistic knowledge. One of the results of this study is a set of publicly available resources which can be used in standard pipelines for processing Russian. However, they also demonstrate hidden costs associated with the use of purely statistical methods and the need to integrate linguistic parameters into statistical procedures.

1 Introduction

The title of this paper refers to a famous research report produced by Martin Kay in the 1980s, "The proper place of men and machines in language translation", finally published in (Kay, 1997), in which Kay argued for the proper distribution of labour between the human translators and the Computer-assisted Translation systems. Another reference appropriate to the topic of the paper presented here is a statement attributed to Fred Jelinek "Every time I fire a linguist the results of speech recognition go up", i.e. explicit linguistic knowledge is dispensable.[1] This sentiment is related to a paradigmatic shift that happened in the computational linguistics in the beginning of the 1990s: with more and more data available and with the advance in the methods of machine learning, more approaches switched from careful encoding of linguistic phenomena to finding statistical correlations in texts (either annotated or raw). The vast majority of publications at major conferences on computational linguistics belong to this paradigm. However, to the best of our knowledge relatively few attempts have been made to apply entirely statistical methods to building tools for processing Russian, e.g., (Sokirko and Toldova, 2005; Nivre et al., 2008; Sharoff et al., 2008). Purely statistical approaches to language processing are also very infrequent in the proceedings of Russian conferences (like this one).

The paper describes three experiments on designing Russian NLP tools, respectively for Part-Of-Speech (POS) tagging, for lemmatisation and for syntactic parsing. Thus, they cover the basic tools needed for doing NLP and corpus linguistics in Russian. The experiments did not exploit any prior knowledge of the Russian language, i.e. we did not use any rules for dealing with any specific Russian phenomenon. Each experiment can be described in the following lines:

[1] However, this story is not entirely correct, see (Jelinek, 2005).

1. take an annotated Russian corpus;
2. design a simplified representation of annotations to convert the corpus into the format suitable for the learning tool to be used;
3. learn a model in several iterations to tune the learning parameters.

In this approach the human efforts are invested into creating annotated corpora, representing data and designing machine learning algorithms, while the machine is able to learn the links between the data. In the end, linguistic knowledge is induced from annotated corpora rather than explicitly hand-crafted by linguists. In a similar way, development of corpora is possible without manual selection of texts from a range of sources. It can be facilitated by crawling or using the API of a search engine and automatically annotating them with respect to their domains and genres (Baroni et al., 2009; Sharoff, 2010).

The automatically induced rules also do not take the form of hard constraints, separating the possible from the impossible, but rather as graded constraints, distinguishing the more probable from the less probable. This makes the automatically acquired models more robust to noise.

In the sections below we briefly outline the statistical methods used in each of the three tasks (Section 2), ways of representing corpus phenomena (Section 3) and the results obtained using our tools (Section 4)

2 Methods used

2.1 Statistical Part-Of-Speech Tagging

POS tagging is aimed at assigning a POS label (tag) to each word in the input stream. Until the end of the 1980s this task had been usually performed by sets of carefully crafted rules for disambiguating the contexts, e.g., for detecting contexts in which the form стали is a noun (‘steel$_{gen,sg}$’) or a verb (‘become$_{past,pl}$’), cf. one of the earliest descriptions of this sort (Nikolaeva, 1958). Ken Church was one of the first researchers to show the possibility of abandoning the rules and relying exclusively on POS-annotated data (Church, 1988). This led to proliferation of statistical approaches to tagging, either using automatic derivation of decision trees, e.g., TreeTagger (Schmid, 1994), Hidden Markov Models (HMM), e.g., TnT (Brants, 2000), or machine learning, e.g. SVMTool (Giménez and Màrquez, 2004).

Probably, the most widely used approach is based on HMM for estimating the probability of a tag from the distribution of words over tags (which tag is more likely for this word), as well as over $N-1$ adjacent tags, with N often fixed at 3 (a trigram model). For example, given a sentence like:

(1) *Это была гравюра на стали*
this was engraving on steel
‘It was a steel engraving’,

the sequence of tags *Noun Preposition Verb* is much less likely than the one for *Noun Preposition Noun*, hence the word стали in this sentence receives the tag *Noun*. Still the probability of the sequence *Noun Preposition Verb* in Russian is greater than zero because of such constructions as шутки ради позвонили...

This study uses the TnT tagger (Brants, 2000). In addition to standard HMM tagging it employs several useful methods for approximating the probabilities of unseen tag sequences (smoothing) as well as for guessing possible tags of unseen words. The latter is done by computing the probability of the last m characters of an unseen word form co-occurring with a given tag. For example, when such forms as *vociferation, votazione*, конъюгация, 自由主义 are missing in respective training corpora, they are still more likely to receive the noun tag on the basis of POS tags for words with the same ending.

2.2 Learning lemmatisation rules

Lemmatisation rules can be also derived automatically from a list of word forms paired with their possible lemmas and POS tags obtained from an annotated corpus (Erjavec and Džeroski, 2004; Jongejan and Dalianis, 2009). The CST lemmatiser used in our experiments tries to find for each pair the longest shared part, e.g., for the pair близкий-поближе the inner part is бли, this leads to the rule *зкий←по*же (the asterisk indicates any character). The training process then tries to apply the new rule across all pairs with the same POS tag. If lemmatisation is successful, nothing needs to be done, e.g., for низкий-пониже. However, if an applicable rule from the rule base produces incorrect lemmatisation, e.g., for the pair плохой-похуже, the rule *зкий←по*же produces хузкий, which does not match the target lemma, then a new lemmatisation rule is generated to cover more specific cases (there is a special strategy to determine which rules are retained as more general and which cover specific cases). The rule generated in this case *лохой←*охуже, since п is shared. Even though the rule is not entirely correct, it is quite unlikely to cause problems in processing real texts, since it fires only when we have a form ending with охуже which gets the tag of a comparative adjective. The training stage runs until all forms in the training set are successfully mapped to their lemmas.

2.3 Syntactic parsing

Syntactic parsing aims at computing a complete hierarchical representation of an input sentence. Statistical methods for parsing has until recently focused on phrase structure parsing for English, resulting in a series of increasingly accurate parsers trained on the Penn Treebank (Magerman, 1995; Collins, 1997; Charniak, 2000; Charniak and Johnson, 2005). However, dependency parsing has emerged as an interesting alternative, especially for languages with more flexible word order than English, as seen in the CoNLL shared tasks on dependency parsing (Buchholz and Marsi, 2006; Nivre et al., 2007). In fact, for decades dependency parsing was the standard approach in the Soviet/Russian linguistic tradition (Mel'čuk, 1988).

Most recent approaches to statistical dependency parsing can be characterized as either *graph-based* or *transition-based* (McDonald and Nivre, 2007). A graph-based parser learns a model for scoring entire dependency graphs and performs exhaustive search for the highest-scoring graph at parsing time; a typical example is MSTParser (McDonald, 2006). A transition-based parser instead learns a model for predicting the next parser action – or transition – and performs greedy search for best transition sequence at parsing time; a typical example is MaltParser (Nivre et al., 2006). Both approaches can give state-of-the-art accuracy, but the transition-based method is potentially much more efficient, which is useful when parsing large amounts of data. The transition-based MaltParser system has previously been applied to Russian with promising empirical results (Nivre et al., 2008).

Table 1: Annotated corpora used in this study

Disambiguated RNC			SynTagRus		
Tokens	Orth words	Sentences	Tokens	Orth words	Sentences
5801316	5115016	432611	719957	635524	41186

3 Russian corpora and their representation

3.1 Annotated corpora used for training

Information about the training corpora is given in Table 1. The Russian National Corpus contains a component with morphosyntactic annotation (Plungian, 2005), which is commonly known as снятник (disambiguated). Originally it contained only fiction, but it has been expanded to cover a range of genres, such as newspapers, informal communication (jokes and forums), scientific&technical texts, etc. For training the parsing tool, we used SynTagRus, a Russian corpus with dependency annotation for every sentence (Boguslavsky et al., 2000). This has been produced by using the output of ETAP (Apresian et al., 2003) with manual correction of incorrect analyses.

3.2 Adapting the Russian tagset

Zalizniak's Grammatical dictionary (Zalizniak, 1977) is a formalisation of Russian morphology, which is commonly used in NLP tools for automatic morphological analysis, e.g., (Segalovich, 2003; Sokirko, 2004). The tagset used in the disambiguated RNC is also largely based on the Zalizniak categories (with few expansions, such as the use of the vocative case).

The problem with using statistical taggers is that they usually operate with atomic labels, e.g., `NNS` in the English Penn tagset stands for 'plural common noun', `NP` stands for 'singular proper noun', while the output of morphological analysis is traditionally represented by a set of features, e.g., for mystem (Segalovich, 2003):

(2) шлепают: шлепать=V,несов=непрош,мн,изъяв,3-л,пе

which corresponds to 'to slap=Verb,imperfective=nonpast,plural,indicative,3rd person,transitive'.

It is possible to produce a tagset by concatenation of the feature set for each word. However, this results in a fairly large number of tags, for example, concatenation of features for all words in the disambiguated RNC produces 4,592 tags, which is too much for trigram tagger learning on a corpus of five million words. The total number of tags reported in (Sokirko and Toldova, 2005) in an experiment, which also used the disambiguated RNC, is 829 tags. This indicates some kind of tagset design, though it is not described in the report.

MTE is a project aiming at standardising the tagset for a range of language (Erjavec, 2010), it covers many other Slavonic languages, so the added advantage of using it was the possibility to create a unified tagset.

The tagset is positional, i.e., for a major POS (Noun, Verb, etc) there are fixed positions with values for features. For example, `Ncfsgn` stands for 'Noun, common, feminine, singular, genitive, inanimate', while `Vmis-sfp` stands for 'Verb,main,indicative,past,-,singular,feminine,perfective', with the

Code	Explanation	Error rate	Relative error	Coverage
N	Nouns	2.08%	7.21%	28.80%
A	Adjectives	0.86%	9.05%	9.51%
P	Pronouns	0.65%	7.82%	8.28%
V	Verbs	0.50%	4.89%	10.16%
C	Conjunctions	0.14%	2.37%	5.84%
R	Adverbs	0.13%	4.69%	2.81%
S	Prepositions	0.13%	0.89%	14.62%
M	Numerals	0.13%	4.60%	2.81%
Q	Particles	0.10%	4.03%	2.59%
I	Interjections	0.01%	26.42%	0.02%

Table 2: Incorrectly assigned POS tags

hyphen occupying the place of the person value (which is not detected for the Russian verbs in the past tense). The prepositions are marked for the case of the noun phrase they govern. Example (1) receives the following analysis:

(3) *Это* *была* *гравюра* *на* *стали*
P–nsnn Vmis-sfa Ncfsnn Sp-l Ncfsln

SynTagRus is also a part of the Russian National Corpus, but because of the differences in its morphological categories, it uses a separate query interface. The SynTagRus tagset has been also mapped to a subset of MTE. Given that SynTagRus does not contain the category of pronouns (the personal pronouns in it are coded as nouns, possessive pronouns as adjectives, etc), its mapping to MTE produces a smaller tagset in comparison to the RNC. So the extra task in this case was to map the RNC-based output of the tagger to the SynTagRus-based set of tags.

4 Results

4.1 Tagging

Out of the 5 million orthographic words of the disambiguated RNC 10% was kept in the held-out portion used for evaluation. The tagger was trained on the remainder of the disambiguated RNC, and the overall accuracy on the held-out portion was 95.28% (with punctuation excluded).

We also measured the performance of TnT on a reduced tagset of Russian (only codes in Table 2). The accuracy reached 97.09%, which is only slightly better than the performance of the tagger on the detailed tagset, while the detailed tagset is more beneficial for many NLP tasks.

The types of errors produced by the tagger on the full tagset are illustrated in Table 2 and Table 3. The error rate in Table 2 refers to the total count of errors for this category, this is a measure of how important this type of errors is for tagging a text (the table is sorted by this column). It is also interesting to know the amount of word forms *within* each category tagged incorrectly. This is the relative error rate, which reflects how difficult the category is for the tagger, e.g. 7.21% rate for nouns means one out of 14 nouns gets a tag which is incorrect in at least one position, while only one out of 112 prepositions (0.89%) gets a wrong tag (the preposition is not recognised or the case is not assigned correctly). The coverage refers to the total amount of such POS tags in the held-out portion of the RNC, this indicates the relative importance of the category.

The evaluation on individual categories reveals that the most difficult POS category is the category of nominals, which includes adjectives and nouns, as well as pronouns, which is a fringe member, including nominal pronouns (P-----n) and attributive pronouns (P-----a) with nominal inflection, as well as adverbial pronouns (P-----r). The apparently high relative error rate for interjections is explained by the fact that the two most common interjections are ‘a’ and ‘o’ (ambiguous with a common conjunction and preposition respectively), and their low frequency does not influence the overall error rate much.

A more detailed look at the sources of errors presented in Table 3 reveals the following problems:

1. distinguishing between closely related POS classes, such as pronouns and conjunctions (как, когда, что, то), similarly for particles (же, ли);
2. dealing with long-distance dependencies, especially in distinguishing between the nominative and accusative cases (все, право, это);
3. domain mismatch, when the training corpus and the held-out one referred to different domains (судов, masculine or neuter, лиц, animate or inanimate);
4. guessing the full tag for abbreviations (ЭВМ, which was plural genitive in the held-out portion of the RNC, but got the tag of singular genitive in the absence of other indicators of plurality);
5. distinguishing between adverbs and short adjectives (e.g., удобно).

In spite of the number of problems in statistical tagging, a recent comparison of several Russian disambiguation tools in (Ljashevskaja et al., 2010) demonstrated its reasonable performance against other disambiguation and lemmatisation tools (our tagger and lemmatiser are reported there under the names of Peru and Pine). The accuracy of POS tagging achieved on that corpus was 97.3%, which was considerably better than the majority of other (rule-based) systems. In addition to this, the worst performing component of the tagger was the rule-based tokeniser, which incorrectly identified token boundaries and thus decreased the overall performance.

4.2 Lemmatisation

These are the rules generated for the tag `Ncmsgy` for nouns ending in -ц:

ец	еца
иц	ица
заяц	зайца
ец	йца
я-муромец	и-муромца
ринц	ринца
ртц	ртца
ец	ьца
ец	ца

The model for Zalizniak’s Index 5 (masculine nouns ending in -ц) is well-represented, including the regular forms with and without morphological alternation (кузнец-кузнеца, фриц-фрица,

0.0932%	TnT	как	C
0.0920%	RNC	как	P-----r
0.0788%	TnT	что	C
0.0682%	TnT	ЭВМ	Ncfsgn
0.0682%	RNC	ЭВМ	Ncfpgn
0.0507%	RNC	что	P--nsnn
0.0444%	TnT	это	P--nsnn
0.0438%	TnT	как	P-----r
0.0413%	TnT	судов	Ncnpgn
0.0413%	RNC	судов	Ncmpgn
0.0413%	RNC	как	C
0.0363%	TnT	все	P--nsnn
0.0357%	RNC	это	Q
0.0350%	RNC	все	R
0.0338%	RNC	что	P--nsan
0.0325%	TnT	его	P-3msan
0.0300%	RNC	их	P-3-pgn
0.0288%	TnT	то	P--nsnn
0.0288%	RNC	когда	P-----r
0.0288%	TnT	когда	C
0.0269%	RNC	то	C
0.0263%	TnT	же	Q
0.0263%	RNC	же	C
0.0244%	RNC	что	C
0.0244%	RNC	лиц	Ncnpgy
0.0244%	TnT	лиц	Ncnpgn
0.0238%	TnT	что	P--nsnn
0.0238%	TnT	ли	Q
0.0238%	RNC	ли	C
0.0219%	TnT	право	Ncnsan
0.0219%	TnT	их	P-----a
0.0206%	RNC	право	Ncnsnn
0.0188%	TnT	его	P-----a
0.0181%	RNC	все	P--nsan

Table 3: Most common incorrectly tagged words

европеец-европейца, принц-принца, владелец-владельца, чеченец-чеченца), as well as some exceptions, including the irregular заяц-зайца and the occasional forms артц-артца (used in Vasily Grossman's "Life and fate") and Ильи-Муромца, which came from the inability of the lemmatiser to deal with the hyphenated nouns.

The statistical lemmatiser depends on the output of tagging, but it is moderately tolerant to

	LAS	**UAS**
SynTagRus tags, poly-SVM	83.4	89.4
MTE tags, poly-SVM	82.8	88.8
MTE tags, linear SVM	82.2	88.0

Table 4: Parsing results on development set of SynTagRus; labeled attachment score (LAS) and unlabeled attachment score (UAS).

tagger errors. For example, irrespectively of the error in getting the animacy of лиц in Table 3 it still gets the right lemma. However, the error in getting the gender of судов leads to incorrect lemmatisation.

4.3 Syntactic parsing

Because of the need to tune the parameters during parsing, SynTagRus was split into three parts, the training set (507986 words), the development set for tuning the parameters (64196 words) and the test set for the final evaluation (63342 words). Table 4 shows results on the development set for three different settings with the standard evaluation metrics: labeled attachment score (LAS), the proportion of words that are assigned the correct head *and* dependency label, and unlabeled attachment score (UAS), the proportion of words that are assigned the correct head (regardless of label).

The first experiment replicates the settings from (Nivre et al., 2008) exactly, using the original part-of-speech tags from the SynTagRus treebank and using SVMs with a polynomial kernel to predict the next parser transition.[2] The results obtained are slightly better than the ones reported by (Nivre et al., 2008) (LAS 82.3, UAS 89.0), which is probably due to a larger training set. The second experiment uses the same features and the same type of classifier (poly-SVM) but replaces the SynTagRus part- of-speech tags with the MTE tags. This results in slightly lower parsing accuracy, about 0.6 percentage points for both metrics.

Using SVMs with a polynomial kernel is rather inefficient during both training and parsing. For example, parsing the development set of 68,314 tokens takes about three hours. In the third experiment, we therefore used a linear SVM, together with a slightly extended set of features to compensate for the lack of the polynomial kernel. The result is a much faster parser, which parses the development set in under two minutes, although with slightly lower accuracy. This parsing model will be applied to the Russian Web corpus of about 3 billion words, and it is expected to complete parsing in under two months.

5 Conclusions

This paper presents a fairly radical stance: it is redundant to encode linguistic knowledge explicitly; a completely automatic machine learning procedure can quickly produce a fast and reliable NLP

[2] Besides part-of-speech tags, the parser uses word forms, lemmas and morphosyntactic features as a basis for prediction; see (Nivre et al., 2008) for more details.

component, which rivals (and in some cases exceeds) the performance of hard-coded linguistic rules requiring the efforts of many person-months (if not years). Hence, the efforts of linguists need to be spent on creating data rather than writing rules.

Nevertheless, this claim needs to be taken with a pinch of salt. First, the approach was reasonably successful since it implicitly utilised some information about the language. The methods for unknown word guessing as well as lemmatisation used in this study rely on the fact that Russian is a flective language. Statistical tagging and lemmatisation are known to be more difficult for agglutinative languages, like Turkish (Dincer et al., 2008). For an isolating language, like Chinese, there is no problem with lemmatisation, but the greater average ambiguity of the POS tags for known words and the lack of reliable prediction of the POS tag for unknown words makes the accuracy of knowledge-free methods considerably lower.

Second, data representation in terms of tag labelling is sufficiently simple and efficient, but a tag label lacks information about the internal structure of linguistic phenomena. For example, when the system learns the structure of Russian noun phrases, it does not take into account the agreement in case, number and gender. It only learns the fact that `Afpmsg` is normally followed by `Ncmsgn`, `Ncmsgy` or `Npmsgy`, while `Afpfsd` is followed by `Ncfsdn`, etc. However, if the set of training examples does not contain a proper masculine *inanimate* noun (`Npmsgn`) in this sequence, the tagger will fail to treat the sequence of `Afpmsg Npmsgn` as a noun phrase, even if the concept of animacity is not relevant to the noun phrase construction.

Yet another problem in using purely statistical methods is the reliance on patterns present in training data. Each training set has its own peculiarities, which do not necessarily match the peculiarities of the application domain. For example, the impressive accuracy of 97-98% for HMM tagging is obtained on well-controlled newspaper texts (*The Wall Street Journal* for English and *Frankfurter Rundschau* for German), but the accuracy of taggers trained on these corpora drops dramatically on other text genres, down to 85.7% on Internet forums, i.e., every seventh word is tagged incorrectly (Giesbrecht and Evert, 2009). This does not indicate any inferior status of Internet forums, just the fact that the trigram model trained on newspaper texts does not approximate them well. Annotating texts in the application domain to obtain more training data is expensive, so the tools are often used in new domains without formal evaluation of their accuracy, e.g., ukWac (Baroni et al., 2009) has been tagged and lemmatised with the default TreeTagger model. This problem is partly addressed by new approaches to machine learning using domain adaptation, which uses a training corpus from the source domain (with available annotated data), a small number of annotated examples from the target domain and a large number of unlabelled examples from the target domain (Daumé III et al., 2010).

In addition to the known problem of unknowns in the domain mismatch, there is a problem of unknown knowns, namely when peculiarities inherent in the annotated set are not obvious, while machine learning is likely to emphasise them for making classification decisions. In the end, the system might achieve reasonably good accuracy on the held-out portion of the annotated set (since it is drawn from the same distribution), while this accuracy could be irrelevant outside of the annotated set alone. For example, in the field of automatic genre classification it has been shown that a large number of texts on a particular topic within a genre heading can considerably affect the decisions made by the classifier, e.g., by treating texts on hurricanes and taxation as belonging to FAQs (Wu et al., 2010). At the same time, a classifier based on POS trigrams is much less successful, but it suffers less from the transfer from one annotation set to another (Petrenz and Webber, 2010).

Finally, there are problems with correcting the results. An error produced by a rule-based tagger can be corrected by debugging, finding the incorrectly fired rule, modifying it and testing the performance again. A statistical model can be amended by modification of the learning parameters or by providing more data, but this is only indirectly related to the performance of the system in the case of an individual problem.

In either case, the main contribution of the paper is two-fold. First, we describe the baseline for natural language processing for Russian using only statistical methods and minimal adjustment to the representation of source data. In spite its minimalism, the baseline outperforms the majority of the rule-based systems (Ljashevskaja et al., 2010). Second, the tools reported in this paper are available for linguistic research.[3] This defines the entire pipeline, which starts with POS tagging of pre-tokenised texts, proceeds to lemmatisation and ends with syntactic parsing.

Acknowledgements

Research reported in this paper was partly funded by European Community's Seventh Framework Programme (FP7/2007-2013) under Grant Agreement no 248005 (TTC)[4] and partly by European Community's Life Long Learning Programme (project Kelly, Keywords for Language Learning for Young and adults alike).[5]

References

Apresian, J., Boguslavsky, I., Iomdin, L., Lazursky, A., Sannikov, V., Sizov, V., and Tsinman, L. (2003). ETAP-3 linguistic processor: a full-fledged NLP implementation of the MTT. In *First International Conference on Meaning-Text Theory*, pages 279–288, Paris, Ecole Normale Superieure.

Baroni, M., Bernardini, S., Ferraresi, A., and Zanchetta, E. (2009). The WaCky wide web: a collection of very large linguistically processed web-crawled corpora. *Language Resources and Evaluation*, 43(3):209–226.

Boguslavsky, I., Grigorieva, S., Grigoriev, N., Kreidlin, L., and Frid, N. (2000). Dependency treebank for russian: concept, tools, types of information. In *Proceedings of the 18th conference on Computational linguistics - Volume 2*, COLING '00, pages 987–991.

Brants, T. (2000). TnT - a statistical part-of-speech tagger. In *Proc. of 6th Applied Natural Language Processing Conference*, pages 224–231, Seattle.

Buchholz, S. and Marsi, E. (2006). CoNLL-X shared task on multilingual dependency parsing. In *Proceedings of the 10th Conference on Computational Natural Language Learning (CoNLL)*, pages 149–164.

Charniak, E. (2000). A maximum-entropy-inspired parser. In *Proceedings of the First Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 132–139.

[3] They can be downloaded from `http://corpus.leeds.ac.uk/tools`

[4] `http://www.ttc-project.eu`

[5] `http://su.avedas.com/converis/contract/321`

Charniak, E. and Johnson, M. (2005). Coarse-to-fine n-best parsing and MaxEnt discriminative reranking. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 173–180.

Church, K. (1988). A stochastic parts program and noun phrase parser for unrestricted text. In *Proceedings of the second conference on Applied natural language processing*, pages 136–143, Austin, Texas.

Collins, M. (1997). Three generative, lexicalised models for statistical parsing. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics (ACL) and the 8th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 16–23.

Daumé III, H., Kumar, A., and Saha, A. (2010). Frustratingly easy semi-supervised domain adaptation. In *Workshop on Domain Adaptation for Natural Language Processing at ACL2010*, Uppsala.

Dincer, T., Karaoglan, B., and Kisla, T. (2008). A suffix based part-of-speech tagger for turkish. In *Third International Conference on Information Technology: New Generations*, pages 680–685, Los Alamitos, CA.

Erjavec, T. (2010). Multext-east version 4: Multilingual morphosyntactic specifications, lexicons and corpora. In *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*, Valletta, Malta.

Erjavec, T. and Džeroski, S. (2004). Machine learning of morphosyntactic structure: Lemmatizing unknown Slovene words. *Applied Artificial Intelligence*, 18(1):17–41.

Giesbrecht, E. and Evert, S. (2009). Part-of-Speech (POS) Tagging - a solved task? An evaluation of POS taggers for the Web as corpus. In *Proceedings of the Fifth Web as Corpus Workshop (WAC5)*, pages 27–35, Donostia-San Sebastián.

Giménez, J. and Màrquez, L. (2004). SVMTool: A general pos tagger generator based on support vector machines. In *Proceedings of the Forth Language Resources and Evaluation Conference, LREC 2004*, Lisbon.

Jelinek, F. (2005). Some of my best friends are linguists. *Language Resources and Evaluation*, 39(1):25–34.

Jongejan, B. and Dalianis, H. (2009). Automatic training of lemmatization rules that handle morphological changes in pre-, in- and suffixes alike. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, Suntec, Singapore.

Kay, M. (1997). The proper place of men and machines in language translation. *Machine Translation*, 12(1-2):3–23.

Ljashevskaja, O., Astaf'eva, I., Bonch-Osmolovskaja, A., Garejshina, A., Ju., G., D'jachkov, V., Ionov, M., Koroleva, A., Kudrinskij, M., Litjagina, A., Luchina, E., Sidorova, E., Toldova, S., Savchuk, S., and Koval', S. (2010). Ocenka metodov avtomaticheskogo analiza teksta: morfologicheskie parsery russkogo jazyka. In *Trudy konferencii Dialog10*, pages 318–326.

Magerman, D. M. (1995). Statistical decision-tree models for parsing. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 276–283.

McDonald, R. (2006). *Discriminative Learning and Spanning Tree Algorithms for Dependency Parsing*. PhD thesis, University of Pennsylvania.

McDonald, R. and Nivre, J. (2007). Characterizing the errors of data-driven dependency parsing models. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 122–131.

Mel'čuk, I. (1988). *Dependency Syntax: Theory and Practice*. State University of New York Press.

Nikolaeva, T. (1958). Soviet developments in machine translation: Russian sentence analysis. *Mechanical Translation*, 5(2):51–59.

Nivre, J., Boguslavsky, I. M., and Iomdin, L. L. (2008). Parsing the SynTagRus treebank of russian. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 641–648.

Nivre, J., Hall, J., Kübler, S., McDonald, R., Nilsson, J., Riedel, S., and Yuret, D. (2007). The CoNLL 2007 shared task on dependency parsing. In *Proceedings of the CoNLL Shared Task of EMNLP-CoNLL 2007*, pages 915–932.

Nivre, J., Hall, J., and Nilsson, J. (2006). Maltparser: A data-driven parser-generator for dependency parsing. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC)*, pages 2216–2219.

Petrenz, P. and Webber, B. (2010). Stable classification of text genres. *Computational Linguistics*, 34(4).

Plungian, V. A. (2005). Zachem nuzhen nacionalny korpus russkogo yazyka. In *Nacionalny korpus russkogo yazyka*, pages 6–20. Indrik, Moscow. (in Russian).

Schmid, H. (1994). Probabilistic part-of-speech tagging using decision trees. In *International Conference on New Methods in Language Processing*, Manchester.

Segalovich, I. (2003). A fast morphological algorithm with unknown word guessing induced by a dictionary for a web search engine. In *Proc. of MLMTA-2003*, Las Vegas.

Sharoff, S. (2010). In the garden and in the jungle: Comparing genres in the BNC and Internet. In Mehler, A., Sharoff, S., and Santini, M., editors, *Genres on the Web: Computational Models and Empirical Studies*. Springer, Berlin/New York.

Sharoff, S., Kopotev, M., Erjavec, T., Feldman, A., and Divjak, D. (2008). Designing and evaluating a Russian tagset. In *Proceedings of the Sixth Language Resources and Evaluation Conference, LREC 2008*, Marrakech.

Sokirko, A. (2004). Morphologicheskie moduli na sajte `www.aot.ru`. In *Proc. DIALOG'04*. In Russian.

Sokirko, A. and Toldova, S. (2005). Sravnenie effektivnosti dvuh metodik snyatiya lexicheskoy i morfologicheskoy neodno znachnosti dlya russkogo yazyka. In *Internet-matematika*. In Russian.

Wu, Z., Markert, K., and Sharoff, S. (2010). Fine-grained genre classification using structural learning algorithms. In *Proc. of ACL 2010*, Uppsala.

Zalizniak, A. (1977). *Grammaticheskiyj Slovar' Russkogo Jazyka*. Russkij Jazyk.