KNOWLEDGE- AND LABOR-LIGHT MORPHOLOGICAL ANALYSIS[1]

Jirka Hana
The Ohio State University

Abstract

We describe a knowledge and labor-light system for morphological analysis of fusional languages, exemplified by analysis of Czech. Our approach takes the middle road between completely unsupervised systems on the one hand and systems with extensive manually-created resources on the other. For the majority of languages and applications neither of these extreme approaches seems warranted. The knowledge-free approach lacks precision and the knowledge-intensive approach is usually too costly. We show that a system using a little knowledge can be effective. This is done by creating an open, flexible, fast, portable system for morphological analysis. Time needed for adjusting the system to a new language constitutes a fraction of the time needed for systems with extensive manually created resources: days instead of years. We tested this for Russian, Portuguese and Catalan.

[1]This paper differs only slightly from a paper presented at the Department of Linguistics of The Ohio State University in 2004 and finished in 2005. The morphological analyzer described in this paper was developed as part of a joint project with Anna Feldman and Chris Brew aimed at developing a portable resource-light tagger. I thank Chris Brew, Anna Feldman, Jan Hajič, Brian Joseph and Detmar Meurers for providing valuable feedback and to Steve Evans for helping me in my struggle with the English language.

The work described in this paper was partially supported by NSF CAREER Award 0347799.

Jirka Hana. Knowledge- and labor-light morphological analysis. *OSUWPL*, Volume 58, Fall 2008, pp. 52–84.

1 Introduction

This paper describes a knowledge- and labor-light system for morphological analysis of Slavic languages, namely Czech and Russian. Our approach takes the middle road between completely unsupervised systems à la (Goldsmith 2001) on the one hand and systems with extensive manually-created resources à la (Hajic 2004) on the other. These approaches are scientifically interesting and there are cases when they are also practically justifiable (e.g., the former for analyzing understudied languages and the latter for applications requiring very high precision). However we believe that for the majority of languages and majority of purposes neither of these extreme approaches seem warranted. The knowledge-free approach still lacks precision and the knowledge-intensive approach is usually too costly. We show that a system that uses a little knowledge can be effective. We exploit the 80:20 rule: The part of the work that is easy to do *and* that matters most is done manually or semi-automatically and the rest is done automatically.

Czech this way? We use Czech to test our hypotheses. We do not suggest that morphological analysis of Czech should be designed exactly in the way we do. An excellent high precision system using manual resources[2] already exists (Hajic 2004). The main reason for working with Czech is that we can easily evaluate our system on the Prague Dependency Treebank – a large morphologically annotated corpus (http://ufal.mff.cuni.cz/pdt).

However, no manual resources, including those of (Hajic 2004), can cover arbitrary text – there is an unbounded universe of names (people, products, companies, musical groups, ...) technical terms, neologisms, quotes from other languages; typos, ... We suggest that for languages such as Czech and Russian, morphological analysis should rely on extensive manual resources backed up by a system similar to ours. Less dense languages (e.g., Sorbian, Romany, Czech used in chat-rooms or in any other specialized settings, etc.) can use less of the expensive manual resources and more of the automatic or semiautomatic resources.

The system. For our work, we developed an open, flexible, fast and portable system for morphological analysis. It uses a sequence of analyzing modules. Modules can be reordered, added or removed from the system. And although we provide a basic set of analyzing modules, it is possible to add other modules for specific purposes without modifying the rest of the system. The modules we provide are re-usable for both resource-light and resource-intensive approaches, although the latter option is not explored in detail here.

Nouns only. In the rest of the paper we focus exclusively on nouns. We have several reasons for this:

1. they are hard for the unsupervised systems, because their endings are highly homonymous (at least in Slavic languages);

[2]We use the term *manual resources* to refer to manually-created resources, *automatic resources* to automatically created resources (with possibly some minor manual input) and *semi-automatic resources* to automatic resources manually corrected (fully or partially).

Lemma freq decile	Number of tokens	Corpus coverage (%)	Cumulative coverage (%)	Lemmas not in tr2 (%)
10	164 643	74.1	74	0.2
9	22 515	10.1	84	6.7
8	11 041	5.0	89	22
7	6 741	3.0	92	36
6	4 728	2.1	94	48
5	3 179	1.4	96	61
4	2 365	1.1	97	65
3	2 364	1.1	98	70
2	2 364	1.1	99	75
1	2 364	1.1	100	77

Note: Each decile contains 2364 or 2365 noun lemmas.

Table 1: Corpus coverage by lemma frequency

2. they are the class where the manually-created resources approach fails the most – they are the most open class of all (consider proper names);

3. for practical reasons, we have to limit the scope of our work.

Written language. Finally, it is necessary to stress that we are concerned only with analysis of a written text, not speech.

2 Motivation – Lexical statistics of Czech

To motivate our approach, we provide some statistics about Czech nouns, assuming that nouns in other Slavic languages behave similarly. The statistics are based on the t1 and tr2 corpora (§A.1). The tr1 corpus contains 222 304 noun tokens (out of 619 984 all tokes) corresponding to 42 212 distinct forms (87 321) and 23 643 lemmas (43 056).[3]

Table 1 and Figure 1 break lemmas into deciles by their frequency and compare their corpus coverage. Similarly as the Zipf's law (Zipf 1935; Zipf 1949), they make two things apparent:

- It is quite easy to get a decent coverage of a text with a small number of high frequency lemmas. The 2.4K lemmas in the 10th decile cover 3/4 of noun tokens in

[3]The lemmas in tr1 (and in the whole PDT), distinguish not only between homonyms but often also between words related by polysemy. For example, there are at least four different lemmas for the word *strana*: strana-1 'side (in space)', strana-2 'political party' strana-3 '(contracting) party, (on somebody's) side, ..', strana-4 'page'. All four have the same morphological properties – it is a feminine noun, paradigm *žena*. While this statistics treats them as four distinct entities, our Guesser and automatically acquired lexicons do not distinguish between them. However, the statistics are still valid, because only relatively few lemmas have such distinction.

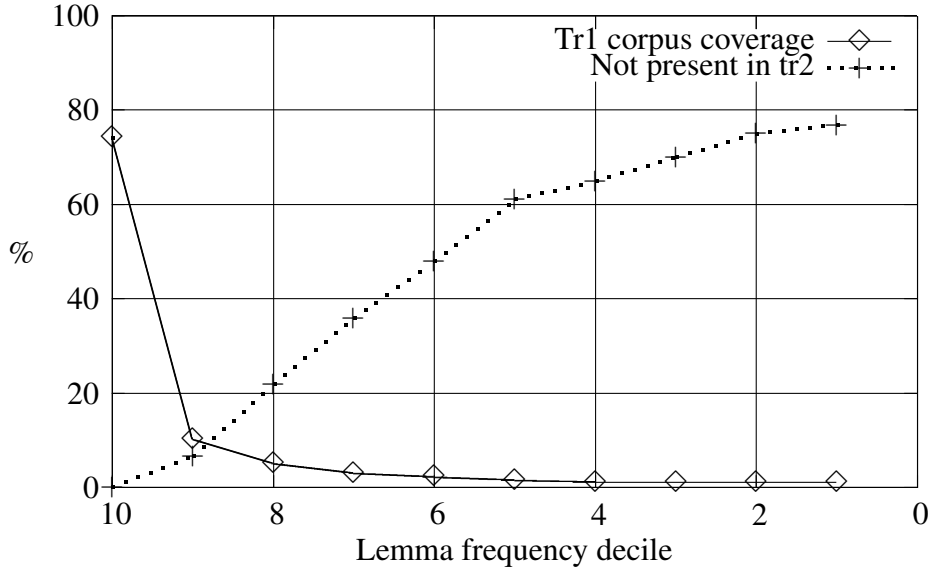

Figure 1: Lemma characteristics by frequency

the corpus, 7.1K lemmas in the top three deciles cover nearly 90% of all noun tokens. That means that even in labor-light systems, it is not necessary to go the way of completely automatically acquired morphology.

- It is very hard, practically impossible, to get a perfect coverage of a running text even with very large lexicons.

  – First, the lemmas in each of the lower deciles add relatively much smaller coverage.

  – Second, infrequent lemmas also tend to be text specific. 77% of the lemmas in the lowest decile of the `tr1` corpus did not occur in the `tr2` corpus – even though the corpora are very similar (they both consists from texts from the same newspapers and magazines). Even when we take the first half of the lemmas (decile 1-5), 70% of the lemmas are text specific!

These facts justify our approach – to provide manually a small amount of information that makes the most difference and let the system learn the rest. This makes it possible to keep the amount of necessary labor close to that of the unsupervised system with quality not much worse than that of the expensive system with manual resources.

3 A Morphological Analyzer of Czech

In this section, we introduce both the general framework for doing and training resource-light morphological analyzes and its instantiation on Czech. Application to other languages is discussed in §4.

In this section, we discuss the analyzer in general (§3.1); the strategy of using it (§3.2); how morphological paradigms are seen by a linguist and how by our system (§3.3); automatic creation of morphological resources – a large lexicon (§3.4) and a list of abbreviations (§3.5). Finally, we evaluate the whole system in §3.6 and suggest several possible enhancements in §3.7.

3.1 Morphological analyzer

Morphological analysis is a function that assigns a set of lemmas (base forms), each with a set of tags, to a form:

(1) MA: form → set(lemma × set(tag))

ženou → { (*žena* 'woman', {noun fem sing inst }),
(*hnát* 'hurry', {verb pres pl 3rd }) }

ženy → { (*žena* 'woman', {noun fem sing gen,
noun fem pl nom,
noun fem pl acc,
noun fem pl voc }) }

Our goal was to design an open, fast, portable and easily configurable morphological analyzer. It is a modular system that queries its analyzing modules in a particular order. Any module can be loaded several times with different parameters (say, different lexicons). A module receives information about the word, its potential prefixes and its context (currently just the preceding word with its analysis, and the following word). The module returns zero or more analyses. An analysis must contain information about a lemma and a tag. Depending on the mode the morphological analyzer is run in, it can also contain additional information, like a paradigm name, ending length, etc.

3.2 General Strategy

We focus our work and knowledge on creating a limited amount of resources that make the most difference and that are easy to create. The rest is done automatically. The system uses a mix of modules with various level of precision and invested effort. The modules are run in a cascading way. Modules that make less errors and overgenerate less are run before modules that make more errors and overgenerate more. Modules on the subsequent level are used for analysis only if the modules from the previous level did not succeed (although this is configurable).

The system contains three types of modules (in addition there are specialized modules for handling numbers, abbreviations, symbols, etc.):

1. Simple word lists – each word form is accompanied by information about its lemma and tags.

2. Guesser – analyzes words using only information about paradigms.

   On the plus side, (1) the Guesser has a high recall and (2) is very labor-light – it is enough to specify the paradigms. However, the disadvantages are that (1) it has a low precision (overgenerates a lot) and (2) it is quite slow – there are too many things to check and perform on too many analyses.

3. Lexicons – analyzes words using a lexicon and a list of paradigms.

   Lexicon-based analysis has just the opposite properties of the Guesser. It requires a lexicon, which is usually very costly to produce. However, (1) only analyses that match the stem in the lexicon and its paradigm are considered; (2) it is very fast, because stem changes, etc. can be computed before hand and be simply listed in the lexicon. The problem of the costly lexicon is partly addressed in §3.4.

Traditional labor-intensive systems use information about paradigms together with a large lexicon, possibly backed up by a guesser (e.g., Hajic 2004; Mikheev & Liubushkina 1995). Word lists are usually used for languages with simple inflectional morphology like English. It might seem obvious that for Czech, a language with 7 cases, 2 numbers and 4 genders, form lists are out of the question. However, in practice only few lemmas occur in a larger number of forms. Table 2 summarizes the distribution of lemma occurrences in the `tr1` corpus in terms of the number of encountered forms. It can be seen that 64% of the lemmas occur only in one form.

	Lemmas		
Nr of forms	Count	Percentage	Cumulative percentage
1	15 192	64.26	64
2	4 155	17.57	82
3	1 807	7.64	89
4	948	4.01	93
5-9	1 523	6.44	100
10-17	18	0.08	100
Total	23 643	100	

Table 2: Noun lemma distribution by the number of forms in the corpus

Entering a lexicon entry is very costly. While it is usually easy (for a native speaker) to assign a lemma to one of the major paradigm groups, it takes considerably more time to select the exact paradigm variant differing only in 1 or 2 forms (in fact, this may be even idiolect-dependent). For example, it is easy to see that *atom* 'atom' does not decline according to the neuter paradigm *město* 'town' but it takes more time to decide to which of the hard masculine inanimate paradigms it belongs (See Table 3). On the other hand, entering possible analyses for individual word forms is usually very straightforward.

Therefore, our system uses a list of manually entered analyses for the most common forms, an automatically acquired lexicon for less common words and finally, the ending-based guesser as a safety net covering the rest.

		hard masculine inanimate paradigms				
	atom 'atom'	hrad 'castle'	ostrov 'island'	rybník 'pond'	zámek 'chateau'	domeček 'small house'
S1	atom-0	hrad-0				
S2	atom-u	hrad-u	ostrov-u/a	rybník-u/a		
S3	atom-u	hrad-u				
S4	atom-0	hrad-0				
S5	atom-e	hrad-e			zámk-u	domečk-u
S6	atom-u	hrad-ě/u		rybníc-e/ík-u	zámk-u	
S7	atom-em	hrad-em				
P1	atom-y	hrad-y				
P2	atom-ů	hrad-ů				
P3	atom-ům	hrad-ům				
P4	atom-y	hrad-y				
P5	atom-y	hrad-y				
P6	atom-ech	hrad-ech			zámc-ích	domečc-ích/čk-ách
P7	atom-y	hrad-y				

Table 3: Forms of *atom* 'atom' and the hard masculine inanimate paradigms

Note that the process of providing the form list is not completely manual – a native speaker selects the correct analyses from those suggested by the ending-based guesser. Analyses of closed-class words can be entered by a non-native speaker on the basis of a basic grammar book. Finally, there is the possibility to manually process the automatically acquired lexicon: a native speaker removes the most obvious errors for the most frequent lexical entries. They remove errors that are easy to identify and that have the highest impact on the results of the system. We did not use this possibility when building the analyzer for Czech, but we did use when annotating development corpora for Portuguese and Russian.

3.3 Czech paradigms

3.3.1 Czech paradigms seen by a linguist

Simply put, in a fusional language like English or Czech, a paradigm is a set of endings with their tags, e.g., *0* – noun singular, *s* – noun plural. The endings are added to stems producing word forms characterized by those tags, e.g., *cat* – noun singular, *cats* – noun plural. However, life is not easy, and the concatenation is often accompanied by various more or less complicated phonological/graphemic processes affecting the stem, the ending or both, e.g., *potato-**e**s, countr**i**-es, kni**v**-es*, etc.

As a more complex illustration, consider several examples of Czech nouns belonging to the *žena* 'woman' paradigm, a relatively 'well-behaved' paradigm of feminine nouns, in Table 4.[4] Without going too deeply into linguistics, we can see several complications:

[4]Abbreviations of morphological categories, e.g., `S1` – singular nominative, are based on Hajic's (2004)

	woman	owl	draft	goat	iceberg	vapor	fly
S1	žen-a	sov-a	skic-a	koz-a	kr-a	pár-a	mouch-a
S2	žen-y	sov-y	skic-**i**	koz-y	kr-y	pár-y	mouch-y
S3	žen-ě	sov-ě	skic-**e**	koz-**e**	k**ř**-**e**	pá**ř**-e	mou**š**-**e**
S4	žen-u	sov-u	skic-u	koz-u	kr-u	pár-u	mouch-u
S5	žen-o	sov-o	skic-o	koz-o	kr-o	pár-o	mouch-o
S6	žen-ě	sov-ě	skic-**e**	koz-**e**	k**ř**-**e**	pá**ř**-e	mou**š**-**e**
S7	žen-ou	sov-ou	skic-ou	koz-ou	kr-ou	pár-ou	mouch-ou
P1	žen-y	sov-y	skic-**i**	koz-y	kr-y	pár-y	mouch-y
P2	žen-0	sov-0	skic-0	koz-0	k**e**r-0	p**a**r-0	m**u**ch-0
P3	žen-ám	sov-ám	skic-ám	koz-ám	kr-ám	pár-ám	mouch-ám
P4	žen-y	sov-y	skic-**i**	koz-y	kr-y	pár-y	mouch-y
P5	žen-y	sov-y	skic-**i**	koz-y	kr-y	pár-y	mouch-y
P6	žen-ách	sov-ách	skic-ách	koz-ách	kr-ách	pár-ách	mouch-ách
P7	žen-ami	sov-ami	skic-ami	koz-ami	kr-ami	pár-ami	mouch-ami

Table 4: Examples of the *žena* paradigm nouns

1. Ending variation: *žen-ě*, *sov-ě* vs. *burz-e*, *kř-e*, *pář-e*; *žen-y* vs. *skic-i*.

   The dative and local sg. ending is *-ě* after alveolar stops (*d, t, n*) and labials (*b, p, m, v, f*). It is *-e* otherwise.

   Czech spelling rules require the ending *-y* to be spelled as *-i* after certain consonants, in this case: *c, č, d', ň, š*. The pronunciation is the same ([ɪ]).

2. Palatalization of the stem final consonant: *kr-a – kř-e*, *mouch-a – mouš-e*.

   The *-ě/e* ending affects the preceding consonant: *ch* [x] → *š*, *g/h* → *z*, *k* → *c*, *r* → *ř*.

3. Epenthesis: *kr-a – ker*.

   Sometimes, there is an epenthesis in genitive plural. This usually happens when the noun ends with particular consonants. There are certain tendencies, but in the end it is just a property of the lexeme; cf. *občank-a – občanek* 'she-citizen, id-card' vs. *bank-a – bank* 'bank' (both end with *nk*, but one epenthesises and the other not). Some nouns allow both possibilities, e.g., *jacht-a – jachet/jacht* 'yacht'

4. Stem internal vowel shortening: *pár-a – par*.

   Often the vowels *á, í, ou* shorten into *a, i/ě, u* in gen. pl. and sometimes also in dat., loc. and ins. pl. If the vowel is followed by multiple consonants in nom. sg, the shortening usually does not happen. In many cases there are both short and long variants (*pár-a – par – pár-ám/par-ám*, *pár-ách/par-ách*, *pár-ami/par-ami* 'vapor'), usually stylistically different.

tagset, the tagset we use, and are discussed in §A.3.

form	lemma	gloss		category
měst-a	město	town	NS2	noun neut sing gen
			NP1 (5)	noun neut pl nom (voc)
			NP4	noun neut pl acc
tém-a	téma	theme	NS1 (5)	noun neut sing nom (voc)
			NS4	noun neut sing acc
žen-a	žena	woman	FS1	noun fem sing nom
pán-a	pán	man	MS2	noun masc anim sing gen
			MS4	noun masc anim sing acc
ostrov-a	ostrov	island	IS2	noun masc inanim sing gen
předsed-a	předseda	president	MS1	noun masc anim sing nom
vidě-l-a	vidět	see	VpFS	verb past fem sing
			VpNP	verb past neut pl
vidě-n-a			VsFS	verb passive fem sing
			VsNP	verb passive neut pl
vid-a			VeMS	verb transgressive masc sing
dv-a	dv-a	two	CYS1	numeral masc sing nom
			CYS4	numeral masc sing acc

Table 5: Homonymy of the *a* ending.

It would be possible to discuss in a similar manner all the Czech (noun) paradigms. Depending on how you count, there are roughly 13 basic paradigms – 4 neuter, 3 feminine and 6 masculine; plus there are nouns with adjectival declension (another 2 paradigms). In addition, there are many subparadigms and subsubparadigms, all of which involves a great amount of irregularity and variation on the one hand and a great amount of homonymy on the other (see Table 5). For a more detailed discussion, see for example (Karlík *et al.* 1996; Fronek 1999).

3.3.2 Czech paradigms seen by an engineer

There are two different ways to address phonological/graphemic variations and complex paradigm systems when designing a morphological analyzer:

- A linguistic approach. Such a system employs a phonological component accompanying the simple concatenative process of attaching an ending. This implies a smaller set of paradigms and morphemes. Two-level morphology (Koskenniemi 1983; Koskenniemi 1984) is an example of such a system and (Skoumalová 1997) is an example for Czech. The problem is that implementing morphology of a language in such a system requires a lot of linguistic work and expertise. For many languages, the linguistic knowledge is not precise enough. Moreover, it is usually not straightforward to translate even a precisely formulated linguistic description of a morphology into the representation recognized by such system.

  In Czech, the forms of the noun *kra* 'iceberg$_{\texttt{FS1}}$', *kře* 'iceberg$_{\texttt{FS36}}$', *ker* 'iceberg$_{\texttt{FP2}}$'

etc. (see Table 4) would be analyzed as involving the stem *kr*, the endings *-a*, *-ě* and *-0* and phonological/graphemic alternations. Forms of the noun *žena* 'woman$_{\texttt{FS1}}$' (*ženě* 'FS36', *žen* 'FP2', etc.) would belong to the same paradigm as *kra*.

- An engineering approach. Such a system does not have a phonological component, or the component is very rudimentary. Phonological changes and irregularities are factored into endings and a higher number of paradigms. This implies that the terms *stem* and *ending* have slightly different meanings than they traditionally do. A stem is the part of the word that does not change within its paradigm, and the ending is the part of the word that follows such a stem.

  Examples of such an approach are (Hajic 2004) for Czech and (Mikheev & Liubushkina 1995) for Russian. The previous version of our system (Hana *et al.* 2004) also belongs to this category. The advantages of such a system are its high speed, simple implementation and straightforward morphology specification. The problems are a very high number of paradigms (several hundreds in the case of Czech) and impossibility to capture even the simplest and most regular phonological changes and so predict the behavior of new lexemes.

  For example, the English noun paradigm above (*0 – s*) would be captured as several other paradigms including, *0 – s*, *0 – es*, *y – ies*, *f – ves*.

  In Czech, the forms of the noun *kra* 'iceberg$_{\texttt{FS1}}$' would be analyzed as involving the stem *k* followed by the endings *-ra*, *-ře* and *-er*. Forms of the nouns *žena* 'woman$_{\texttt{FS1}}$' and *kra* would belong to two different paradigms.

Our current system is a compromise between these two approaches. It allows some basic phonological alternations (changes of a stem-tail[5] and a simple epenthesis), but in many cases our *endings* and *stems* are still different from the linguistically motivated ones. Therefore, many of the paradigms are still technical.

Currently, our system is capable of capturing all of the processes described in §3.3 except the stem internal vowel shortening:

1. Ending variation: A paradigm can have several subparadigms. There are three paradigms corresponding to the linguistic paradigm *žena* (see Table 4): `NFzena`, subparadigm `NFkoza` and its subparadigm `NFskica`.

   – A subparadigm specifies only endings that are different from the main paradigm. `NFkoza` is like `NFzena` but has *-e* in `S3` and `S6`; `NFskica` is like `NFkoza` but has *-i* in `S2`, `P1`, `P4` and `P5`.

   – Each paradigm restricts the possible tails of stems that can decline according to it. For example, `NFskica` requires the stems to end in *c, č, ž, š* or *j*.

2. Palatalization: A paradigm can specify a simple replacement rule for changing stem-tails. For example, the paradigm `NFkoza` says that stem-final *ch* changes to *š* in `S3` and `S6`.

[5]We use the term *tail* to refer to a final sequence of characters of a string. We reserve the word *ending* to refer to those tails that are morphemes (in the traditional linguistic sense or in our technical sense).

lemma	gloss	paradigm	$stem_1$	$stem_2$	$stem_3$
žena	woman	NFzena	žen	=1	—
sova	owl	NFzena	sov	=1	—
chodba	corridor	NFzena	chodb	chodeb	—
skica	draft	NFskica	skic	=1	=1
koza	goat	NFkoza	koz	=1	=1
kra	iceberg	NFkoza	kr	ker	kř
pára	vapor	NFkoza	pár	par	pář
moucha	fly	NFkoza	mouch	much	mouš
váha	weight	NFkoza	váh	=1	váz

Table 6: Examples of lexical entries for some nouns of the *žena* paradigm

3. Epenthesis: An ending can be marked as allowing epenthesis. All the three paradigms allow epenthesis in P2.

The current paradigm module cannot capture stem vowel changes. Therefore, the Guesser analyzes such forms incorrectly. It still provides the correct tags but not the correct lemma. For example, *par* is analyzed as a form of the incorrect lemma *para* instead of the correct *pára*; the tag NNFP2-----A---- is correct.

Our system specifies 64 noun paradigms (still not exploiting all the possibilities) and 14 common paradigms for adjectives and verbs. The choices on what to cover involve a balance between precision, coverage and effort. More work would be somewhat beneficial but our goal is to stop before the return on effort becomes too low.

3.3.3 Paradigms and Lexicons

A lexicon entry contains information about the lemma, its paradigm and stem or stems. The Lexicon-based Analyzer does not use the information about stem changes that Guesser uses, but instead refers to the stems listed directly in the lexicon entry. This not only speeds up the processing but also makes it possible to capture phonological changes or irregularities that the Guesser is currently unable to handle, including the stem vowel changes mentioned above. Table 6 lists several lexicon entries, for most of them the full declensions can be found in Table 4. $Stem_2$ is used in genitive plural (P2) for all paradigms. This stem expresses epenthesis (*chodb* → *chodeb*) and stem vowel shortening (*pár* → *par*). Entries belonging to the NFskica or NFkoza paradigms can specify a third stem used in dative and locative singular (S3, S6). This stem expresses palatalization (*mouch* → *mouš*).

3.4 Lexicon acquisition

The morphological analyzer supports a module or modules employing a lexicon containing information about lemmas, stems and paradigms. There is always the possibility to provide this information manually. That, however, is very costly. In this section we describe how to acquire a lexicon approximation from a large raw corpus.

This approach differs from the work by Mikheev (1997) or Hlaváčová (2001). Mikheev's (1997) algorithm attempts to acquire a lexicon that would cover forms not covered by a large manually created lexicon. Similarly, Hlaváčová (2001) describes a guesser that acquires rules for analyzing unknown words on the basis of a large set known words (it associates tails, usually endings, often preceded by a final part of a stem, with tags). In other words, in both cases it is assumed that a manually created lexicon covers most of the text and the automatically created lexicon or rules are used only as a backup. In our case, it is the main lexicon that is acquired automatically (note that our form lists are significantly smaller than the lexicons used in Mikheev (1997) or Hlaváčová (2001)).

3.4.1 General idea

The general idea is very simple. The ending-based Guesser module overgenerates. Part of the ambiguity is usually real but most of it is spurious. We can use a large corpus to weed the spurious analyses out of the real ones. In such a corpus, open-class lemmas are likely to occur in more than one form. Therefore, if a lemma-stem-paradigm candidate suggested by the Guesser occurs in other forms in other parts of the corpus, it increases the likelihood that the candidate is real and vice versa.

To make it more concrete: if we encounter the word *talking* in an English corpus, using the information about paradigms, we assume that it is either the *-ing* form of the lemma *talk* or that it is a monomorphemic word (such as *sibling*). Based on this single form we cannot really say more. However, if we also encounter the forms *talk*, *talks* and *talked*, the former analysis seems more probable; and therefore, it seems reasonable to include the lemma *talk* as a verb into the lexicon. If we encountered also *talkings*, *talkinged* and *talkinging*, we would include both lemmas *talk* and *talking* as verbs.

3.4.2 Examples & Problems

We can use our Morphological Analyzer to analyze all the words in the corpus and then create all the possible hypothetical lexical entries consistent with these analyses. After that, we would like to run some filtering that would drop most of the bad entries and leave a small number of entries that would include the good ones. In this subsection, we discuss some of the problems associated with such a filtering.

Let's consider for example the lemma *podpora* 'support'. It is a feminine noun belonging to (a variant of) the *žena* paradigm. The `raw` corpus contains 8138 tokens of this lemma in 9 forms – see Table 7.[6] There are 192 (!) ways to assign a lemma and a paradigm to various subsets of these forms (see Table 8). Most of them sound very funny to a native speaker; only a minority sounds funny to an average learner of Czech; none sounded funny to our Guesser. In this case, we are lucky that we got nearly all the forms of the paradigm, only the vocative singular form is missing, which is not very surprising.

We could select the hypothetical entry that has the highest number of forms. While it would be the correct choice in this case, this strategy would not work in all cases. Con-

[6] We ignore all colloquial forms.

forms	possible case	occurences
podpor-a	S1	810
podpor-y	S2, P1, P4, P5	1633
podpoř-e	S3, S6	782
podpor-u	S4	4128
podpor-o	S5	0
podpor-ou	S7	625
podpor-0	P2	123
podpor-ám	P3	11
podpor-ách	P6	20
podpor-ami	P7	6
podporaa	typo	1

Table 7: Forms of the lemma *podpora* in the `raw` corpus.

# of covered forms	# of entries
9	1
6	2
5	2
4	8
3	7
2	3
1	169

Table 8: Candidate entries for *podpora* forms.

sider for example the noun *bezvědomí* 'unconsciousness' and the adjective *bezvědomý* 'unconscious'. Ignoring negation, *bezvědomí* has 4 theoretical forms, but one of them accounts for 70% of the categories, moreover those much more frequent ones (cf. *pondělí* 'Monday', which declines the same way, in Table 16). *Bezvědomý* has potentially more than 20 forms (cf. *mladý* in Table 17). The problem is that the common form of the former is also a form of the latter. So if we considered a simple majority of forms, the nouns similar to *bezvědomí* would usually lose. We could instead compare the realized percentages of the theoretical number of forms. However, this unnaturally penalizes paradigms with the following properties:

- Paradigms with distinct rare forms. There are many rare categories that are not realized even for a common lemma. For example, vocative is extremely rarely found in a written text. However, for certain paradigms, the form is very easy to find because it is simply the same as a form of a frequent category (e.g., *bezvědomí* 'unconsciousness$_{\texttt{S5/1/2/3/4/...}}$', vs. *pane* 'Mister$_{\texttt{S5}}$' (only `S5`)).
- Paradigms with large number of distinct forms in general. One form is enough to see 25% of forms of a word like *bezvědomí*, while 5 forms are necessary for the same percentage of a word like *bezvědomý*.
- Paradigms with alternative forms: The paradigm *hrad* has only one nominative plural, while the paradigm *pán* has two (e.g., *páni* / *pánové* 'gentlemen$_{\texttt{P1}}$'). Should we count those alternative forms as one or as two? What if some (but not all) work also for a different category?

A different problem is presented by "stolen" forms. Consider the word *atom* 'atom', an inanimate noun of the *hrad* paradigm. The `raw` corpus contains 161 tokens of this lemma in 7 forms – see Table 9. Seeing those 7 forms is not enough to decide whether the words belongs to an animate or inanimate paradigm. There are 5 paradigms each covering all 7 forms; see Table 10 listing two of them. If the `raw` corpus contained only those forms, we could simply keep all 5 hypotheses and still be happy to drop the other 122 hypotheses covering smaller number of forms. The problem is that the corpus also contains 208 tokens of the adjective *atomové* 'atomic$_{\texttt{FS2/FP1/..}}$' that however also fit nom. pl. of the animate paradigm *pán*. Therefore the incorrect paradigm *pán* seems to cover more forms than the correct *hrad* paradigm.[7]

For a native speaker of Czech, it is hard to resist mentioning some of the other non-existing lexical entries our algorithm found at various levels of development:

- Neuter noun *bylo* (paradigm *město*; forms *bylo*$_{S14}$, *byla*$_{S2/P14}$, *byl*$_{P2}$, *byly*$_{P7}$). In fact these are past participle forms of the verb *být* 'to be': *bylo*$_{NS}$, *byla*$_{FS/NP}$, *byl*$_{MS}$, *byly*$_{FP/IP}$. The word lists providing analyses for the most frequent word forms fix this particular problem.
- Neuter noun *architektuře* 'baby architect?' (paradigm *kuře* 'chicken'; form *architektuře*$_{S14}$). In fact, it is a form of the feminine noun *architektura* 'architecture' (*architektuře*$_{S36}$).

[7] It does not help that the corpus also contains the name *Atoma* which looks like animate gen/acc. sg.

forms	possible case	occurrences	
atom-0	S1, S4	48	36%
atom-u	S2, S3, S6	28	21%
atom-e	S5	0	0%
atom-em	S7	1	0%
atom-y	P1, P4, P5, P7	22	17%
atom-ů	P2	30	23%
atom-ům	P3	1	0%
atom-ech	P6	1	0%
Total		132	100%

Table 9: Forms of the lemma *atom* in the `raw` corpus.

	masculine inanimate	*atom* in `raw`	masculine animate	*atom* in `raw`
S1	hrad-0	+	pán-0	+
S2	hrad-u	+	pán-a	
S3	hrad-u	+	pán-u/ovi	+/–
S4	hrad-0	+	pán-a	
S5	hrad-e		pan-e	
S6	hrad-ě/u	–/+	pán-u	+
S7	hrad-em	+	pán-em	+
P1	hrad-y	+	pán-i/ové	–/(+)
P2	hrad-ů	+	pán-ů	+
P3	hrad-ům	+	pán-ům	+
P4	hrad-y	+	pán-y	+
P5	hrad-y	+	pán-i	
P6	hrad-ech	+	pán-ech	+
P7	hrad-y	+	pán-y	+
Total		7		7 (8)

Table 10: Fit of the forms of *atom* to the *hrad* and *pán* paradigms.

- Masculine animate noun *papír* (paradigm *pán*; forms: *papír*$_{S1}$, ***papírové***$_{P15}$, *papíru*$_{S6}$, ...). In fact, these are forms of the nonanimate noun *papír* 'paper' (*papír*$_{S14}$, *papíru*$_{S236}$, ...), and the adjective *papírový* 'made from paper' (*papírové*$_{FP145/IP14/MP4}$)

  The set of endings of animate and inanimate declensions are very similar. One of the distinctions is that only animate paradigms can contain the *-ové* ending (P15). However, many adjectives are derived from inanimate nouns by suffix *-ov-* that can be in certain forms followed by *-é*.[8] The simple higher-number-of-forms wins approach would produce systematic errors.

3.4.3 The algorithm

The algorithm has 4 steps:

1. **Morphological analysis of a raw corpus.**

   For this we can use any morphological analysis that provides information not only about lemmas and tags, but also about the paradigms used. We used our MA system configured to provide the necessary information.

2. **Creating all possible hypothetical lexical entries.**

   Every entry has to contain information about its lemma, paradigm and set of forms that occurred in the corpus.

3. **Filtering out bad entries.**

   The general idea is that the entry that covers the highest number of forms wins. However, taking into account the problems mentioned above, we allow several refinements:

   - Certain forms can be excluded from the counting. Used for endings that cause systematic errors. See the example with *papírové* at the end of the previous section.
   - Certain entries are not dropped even when competing entries cover more forms. Used for paradigms with very low number of distinct forms: *stavení* or *jarní*.
   - An entry covering less frequent forms (e.g., instrumental or vocative) need not be considered if it does not cover frequent forms as well (e.g., nominative).
   - Size of the winning crust can be specified, in relative or absolute terms. A crust of, say, 15% means that not only the entries with the highest number of forms, but also entries with the number of forms 15% smaller are kept. This decreases the precision of the lexicon but increases the recall (i.e., leads to a higher ambiguity and a lower error rate).
   - Minimal number of tokens and/or forms for an entry can be specified. This allows limiting the algorithm to entries with statistically reliable number of forms/tokens.

[8] See Table 10 for an example of animate and inanimate paradigms and Table 17 for adjectival paradigms.

4. **Creating a lexicon.**

   This step is quite uninteresting – it is necessary to create appropriate lexical entries for items that survived all the filtering. For that we need information about the lemma and paradigm which we have and about stem(s) which we can easily derive.

3.5 All caps abbreviation acquisition

The purpose of this subsection is to show that simple methods that do not rely on language-dependent work can be effective. Our goal is to acquire a list of usual all-caps abbreviations (e.g., *BMW*, *ČR* 'Czech Republic', *OSN* 'United Nations', *USA*, *OECD*) from a large unannotated corpus. This is different from the goals of many papers dealing with finding definitions of highly specific abbreviations in scientific tests (e.g., Chang *et al.* 2002; Larkey *et al.* 2000; Yeates 1999; Yeates *et al.* 2000). Here, we are interested in frequent abbreviations and we do not attempt to find their meaning (since they are common, they are usually not defined in the text as the papers cited above require).

3.5.1 A naïve approach and its problems

The solution might look very straightforward: It seems to be enough to extract simply all words that occur only in all caps. However, there are at least three problems with this approach (all frequencies are relative to the `raw` corpus, see §A.1):

1. For various reasons (typographical conventions, errors, etc.) all-caps abbreviations are sometimes written in lower case. For example, *ČTK* 'Czech News Agency' (about 550 occurrences, or 18% are in lower case, abbreviation of article sources are often written in lower case; similarly *AP*, *ITAR*, etc.), *ECU* 'European Currency Unit' (20, 3%), *DIK* a certain kind of investor (30, 13%), etc.

2. Many of the abbreviations are homographs (ignoring case) with normal words. For example, *JE* 'nuclear plant' vs. *je* 'is' (310,000 or 99.8%), *OF* 'a political movement' vs. *of* 'English preposition in institution names, etc.' (1300, 63%), *ME* 'European Cup' vs. *me* 'English or French pronoun in song names, etc.' (about 260 occurrences, or 8% are in lower case); *NATO* vs. *nato* 'with that' (600, 12%), etc. Other examples are *VŠE* 'School of Economics' vs. *vše* 'all', *DNA* vs. *dna* 'bottom', etc.

3. Many non-abbreviations are often written in caps (in titles, for emphasis, typographical conventions, etc.). For example, *PRAHA* 'Prague' (about 7400 occurrences, or 17% are in caps; especially in position when marking the source of a message; similarly other towns), *Jiří* 'a male's name' (1708, 13%; caps are used especially when the name stands as a name for a paragraph – a paragraph about that person; similarly other personal names), *DNES* 'Today – a newspaper' (300, 1%; for some reason that newspaper name is often written in caps), *EKOFLÓRA* 'company name' (3, 100%; for some reason that company writes its name in caps), etc.

And of course these cases are not independent: a non-abbreviation from 2) can be in a title and therefore written in all caps, an abbreviation from 2) can be spelled with a lower case, etc.

3.5.2 Using several heuristics

We decided to identify abbreviations using several non-strict heuristics. These heuristics were found after *quick* inspections of the results obtained by the above simple criterion and of the results of roughly 5 refinements (recall that we intentionally do not use annotated corpora). All-caps abbreviations tend:

1. to occur in all-caps.
2. to be relatively short. Most abbreviations have 2 (*ČR* 'Czech Republic', *OH* 'Olympic Games', *EU*), 3 (*BMW*, *ODS* 'Civic Democratic Party', *ČNB* 'Czech National Bank') or 4 (*OECD*, *ČSSD* 'Czech Socially Democratic Party') characters. Long abbreviations are possible (*UNPROFOR*, *UNICEF*) but very uncommon; moreover, they often decline like normal words (abl. sg. *UNPROFORu* or *UNPROFOR*).
3. to occur not only in all-caps contexts. However, some of them are often accompanied by other abbreviations (*RB – RB OSN* 'Security Council of the United Nations', *ÚV – ÚV KSČ* 'Central Committee of the Communist Party of Czechoslovakia').
4. to contain consonant clusters that would be phonologically impossible under normal pronunciation. Of course, (1) many abbreviations consist of a usual sequence of graphemes (*OS* 'Operating System', *ODA* 'Civic Democratic Alliance'); (2) a text can contain many foreign words that contain 'weird' sequences of consonants (*ss* or *tt* in *Massachusetts*, or *ss* in *Gross*). However, a word that is all-caps and consists exclusively of non-syllabic consonants is nearly 100% an all-cap abbreviation (or a typo).

Naturally, these criteria give very unreliable results when applied to low-frequency tokens.

3.5.3 The algorithm

The algorithm considers all words that ever occur in all caps and collects some basic statistics about them – their frequency, the frequency of their non-all-caps variants, frequency of occurrence in all-caps contexts, their length and whether they have the all-consonant property.

The algorithm has a set of rules each corresponding to one of the above tendencies. Each rule increases or decreases a word's score for various degree of compliance with the tendency. The words are punished for a low frequency or going against the (1)–(3) tendencies; and rewarded for having the all-consonant property. Words with a score above a specified threshold are considered to be all-caps abbreviations. In addition, we ignored common roman numerals (arbitrarily I–XXX).

3.5.4 Evaluation

Training data. The all-caps acquistion algorithm was trained on the non-annotated `raw` corpus described in A.1.

Testing data. We tested the algorithm on the `te` corpus. The corpus is annotated with lemmas, and most abbreviation lemmas are marked by a *:B* code. For example, *BMW* is assigned the lemma *BMW-1_:B_;K* (the company) or *BMW-2_:B_;R* (the car)[9]; *atd* 'etc' is assigned *atd-1_:B*. Some abbreviations are also marked with '8' in the variant slot of their tags.

However, some abbreviations are not marked in either way.[10] For example, *USA* is assigned the lemma *USA_;G* and tag `NNIPX-----A----`. Therefore we decided to identify all-caps abbreviations as words that have all-caps lemmas, are at least two characters long and are not annotated as roman numerals. We manually checked those that do not contain *:B* code in the lemma or '8' in the tag. Out of the 1375 all-caps words in the corpus, we identified 769 as all-caps abbreviations.

We ran the algorithm in three ways:

1. In the first evaluation (the baseline), the algorithm simply checked whether an all-caps word it was among the abbreviations learned from the `raw` corpus or not.

2. In the second evaluation, the algorithm checked whether a word was among the abbreviations learned from the `raw` corpus or not.

3. In the third evaluation, we inspected and corrected the 50 most frequent all-caps words that were accepted as abbreviations and the 50 most frequent all-caps words that we refuted as abbreviations. There were no errors in the accepted list, but there were 6 errors in the refuted list. The errors exemplify the problems of the method and the rules we selected:

   (a) *UNPROFOR* – refuted, because too long (Rule 2). This may be okay, because this is not a classical abbreviation but an acronym, it is also often declined (*UNPROFORu*, *UNPROFOR-u*, *UNPROFORU*).

   (b) *ÚV* 'Central Committee' – refuted because mostly followed by an all-caps word (Rule 3), e.g., *ÚV KSč* 'Central Committee of the Communist Party of Czechoslovakia'.

   (c) *OF* 'Civic Forum' – refuted because often occurs in lower case as English *of* (Rule 1); however nearly none *OF* occurred in all-caps environment.

   (d) *KAN* 'Club of Active Non-partisans' – refuted because often occurs in lower-case as a lower-case abbreviation in names of water works and sewage companies.

[9] *;K* stands for a company, *;R* stands for product. I omit the lemma comments that are in parens and are in Czech. See (Hajic 2004) for more details.

[10] According to (Hajic 2004), eventually all abbreviations should be marked by the '8' in their variant slot.

(e) *RB* 'Security Council' – refuted because mostly followed by all caps *OSN* 'U.N.' (Rule 3).

(f) *OV* 'County Committee' – the same story as *ÚV* in (b).

Experiment	Recall	Precision	F-measure
1 – baseline	100.0%	55.1	71.1
2 – unsupervised	97.3%	91.2	94.1
3 – High frequency hypotheses manually corrected	98.1%	91.4	94.6

Table 11: Evaluation of the abbreviation learner

3.5.5 Possible Enhancements

There are many ways to improve the suggested methods for acquiring a list of abbreviations (and non-abbreviations). It would be worth focusing on the abbreviations that have the same form (ignoring case) as normal words (§3.5.1 item 2). We believe that a more subtle implementation of Rule 2 (§3.5.2) would help – currently only the case of the immediately preceding and following words is considered. Similarly, if a token of an abbreviation candidate occurs several times in a few adjacent sentences or paragraphs, it would be good to compare those tokens: do they all use all caps? (then it is probably an abbreviation), do only some of them use all caps? (then it is probably a normal word in a title or emphasis). It is also possible to use a relatively small annotated tuning corpus to tune the criteria and their parameters.

3.6 Evaluation of the whole system

We evaluated our Morphological Analyzer against the `te` corpus manipulating two parameters:

- Whether a lexicon automatically acquired from the `raw` corpus is used.
- Size of a word list capturing analyses of the most frequent word forms (top forms list, or TFL). The lists were created on the basis of the `raw` corpus.

The results are summarized in Table 12. It is worth repeating that we are concerned only about nouns. The TFLs help without a question – they lower both error-rate (they help with irregular words that are not covered by our paradigms) and ambiguity. The automatic lexicon lowers ambiguity (by pruning incorrect lexical entries), but also increases error-rate (by pruning correct lexical entries). Without TFL, ambiguity decreases by 40% and error rate increases by 38%. With 10K-TFL, ambiguity decreases by 32% and error rate increases by 25%. Depending on what the results will be used for, it may or may not make sense to use an automatic lexicon. The quality of the results is worse than the quality of

Lexicon	–	–	–	+	+	+	Hajič[11]
Top forms list	0K	5K	10K	0K	5K	10K	
Error rate	3.6	2.9	2.7	5.8	3.9	**3.6**	1.3
Ambiguity tag/w	19.6	13.1	11.5	11.7	8.5	**7.8**	3.8
Speed w/s[12]	3000	3500	4800	4500	6500	8200	

Table 12: Evaluation of the Czech morphological analyzer (on nouns)

(Hajic 2004), a system with a large manually created lexicon: Our recall error is roughly three times as large and precision error twice as large.

As mentioned before, the Guesser is relatively slow, therefore using a TFL and/or lexicon increases the speed of analysis.

3.7 Possible enhancements

Currently, the main effort is focused on improving lexicon acquisition: (i) considering frequencies and contexts of word forms when eliminating incorrect hypotheses; (ii) replacing sequential application of heuristics with their weighted parallel combination; (iii) using information about common derivation patterns to extend the algorithm over several lemmas related by derivation and eliminating some of the systematic errors mentioned above. We are also exploring the possibilities of combining our approach with various machine learning techniques. Finally, we are in the process of improving our tools used by native (or informed) speakers to provide the limited amount of information needed by the analyzer in fast and effective way.

4 Application to Russian

To test the portability of our approach to other languages, we created a similar morphological analyzer for Russian (Hana *et al.* 2004), Portuguese (Hana *et al.* 2006) and Catalan (Feldman *et al.* 2006). Here, as an example, we discuss the modification of the system for Russian.

4.1 Russian versus Czech

A detailed comparative analysis of Czech and Russian is beyond the scope of this paper. However, we would like to mention a number of the most important facts. Both languages are Slavic (Czech is West Slavonic, Russian is East Slavonic). Both have extensive inflectional morphology whose role is important in determining the grammatical functions of phrases. In both languages, the main verb agrees in person and number with the subject;

[11] 300K lexicon (Hajič, p.c.)

[12] Running on Sun Java RE 1.5.0.01 with HotSpot, MS Windows XP on Pentium Celeron 2.6 GHz, 750MB RAM. The time need to initialize the system (load and compile lexicons, paradigms etc.) is not included.

adjectives agree in gender, number and case with nouns. Both languages are free constituent order languages. The word order in a sentence is determined mainly by discourse. It turns out that the word order in Czech and Russian is very similar. For instance, old information mostly precedes new information. The "neutral" order in the two languages is Subject-Verb-Object. Here is a parallel Czech-Russian example from our development corpus:[13]

(2) a. [Czech]

Byl jasný, studený dubnový den
was$_{Masc.Past}$ bright$_{Masc.Sg.Nom}$ cold$_{Masc.Sg.Nom}$ April$_{Masc.Sg.Nom}$ day$_{Masc.Sg.Nom}$
i hodiny odbíjely třináctou.
and clocks$_{Fem.Pl.Nom}$ stroke$_{Fem.Pl.Past}$ thirteenth$_{Fem.Sg.Acc}$

b. [Russian]

Byl jasnyj, xolodnyj aprel'skij den'
was$_{Masc.Past}$ bright$_{Masc.Sg.Nom}$ cold$_{Masc.Sg.Nom}$ April$_{Masc.Sg.Nom}$ day$_{Masc.Sg.Nom}$
i časy probili trinadtsat'.
and clocks$_{Pl.Nom}$ stroke$_{Pl.Past}$ thirteen$_{Acc}$

'It was a bright cold day in April, and the clocks were striking thirteen.' [from Orwell's '1984']

Of course, not all utterances are so similar. However, most of the differences are on syntactic levels and in the level of usage. (Hana *et al.* 2004) and (Hana & Feldman 2004), discuss some ways how to address some of those differences.

On the level of morphology, the languages are very close. The order and function of morphemes are nearly identical. Obviously, there are some differences. For example, Russian does not have vocative; Russian marks reflexivity by a verb suffix, while Czech by a reflexive clitic; Russian verb negation is marked by a separate particle, while Czech verb negation is marked by a prefix; Russian adjectives and participles do not distinguish gender in plural; etc. Naturally, the morphemes have different shapes (and are written in different scripts), but even from this point of view, they are also often similar.

4.2 Data

Tag system. We adapted the Czech tag system (see §A.2). It has about 900 tags which is significantly less than the Czech tagset with about 4300 tags. There are two reasons for this: (i) a theoretical one – some Russian categories have fewer values (case; many Czech morphemes distinguish various levels of colloquiality); (ii) the Czech tag system is very elaborate and specially designed to serve multiple needs, while our tagset is designed solely to capture the core of Russian morphology and demonstrate the portability of our techniques for morphological processing. See (Hana *et al.* 2004) for more details.

[13] All Russian examples in this paper are transcribed in the Roman alphabet. Our system is able to analyze Russian texts in several Cyrillic encodings and various transcriptions.

Testing data. For evaluation purposes, we selected and morphologically annotated (by hand) a small portion from the Russian translation of Orwell's *1984*. This corpus contains 4011 tokens and 1858 distinct forms.

Development data. For development testing, we used another part of *1984*. Since we want to work with minimal language resources, the development corpus is intentionally small – 1788 tokens. We used it to test our hypotheses and tune the parameters of our tools.

Raw corpus. For lexicon acquisition (cf. §3.4), we used a large raw corpus – Uppsala Russian Corpus.[14] The corpus contains about 1M tokens (roughly 35 times smaller than the Czech `raw` corpus).

Future data. In the near future, we would like to increase the size of the testing corpus to roughly 10K tokens. These tokens will come from newspaper texts. We plan to include some newspaper texts into the development data as well, however, we still want to keep the size of the development data very small, probably somewhere around 3K tokens.

4.3 Evaluation

We evaluated our system against the 4K tokens of the testing corpus manipulating two parameters:

- Whether a lexicon automatically acquired from the `raw` corpus is used.
- Whether the longest ending filtering (LEF, see (Hana *et al.* 2004)[15]) is used.

For practical reasons, we did not use the top-frequency lists, although we believe they would help significantly. We plan to employ them in a near future.

The results are summarized in Table 13. Again, we are concerned only with nouns; the results on all tokens are shown only for reader's information and comparison with our previous work (Hana & Feldman 2004; Hana *et al.* 2004). The results are not directly comparable with the results for Czech (§3.6) because of the different nature of the testing corpora (newspapers/magazines for Czech vs. fiction for Russian).

In comparison with (Hana *et al.* 2004), we significantly decreased the recall error (9.6% → 6.0%, i.e., 38% relative reduction; on all tokens with both lexicon and LEF), but at the same time ambiguity increased, though about half the relative size (3.1% → 4.0; i.e.,

[14] The corpus is freely available from Uppsala University at http://www.slaviska.uu.se/ryska/corpus.html

[15] This is a simple heuristic to decrease the number of analyses. The heuristic assumes the correct ending is usually one of the longest candidate endings. A similar approach was used by Mikheev (1997). In English, it would mean that if a word is analyzed either as having a zero ending or an *-ing* ending, we would consider only the latter; obviously, in the vast majority of cases that would be the correct analysis. In addition, we specify that few long but very rare endings should not be included in the maximum length calculation. To stay within the labor-light paradigm, we capture only the few most common systematic errors the LEF does.

	Lexicon	–	+	–	+
	LEF	no	no	yes	yes
All	recall error	**3.6**	5.6	5.7	6.0
	ambiguity (tag/w)	12.9	4.6	9.5	**4.0**
Nouns	recall error	**1.7**	5.3	1.7	5.3
	ambiguity (tag/w)	24.5	6.9	18.1	**5.9**

Table 13: Evaluation of the Russian morphological analyzer

23% relative increase). We believe that for most applications this is an improvement. For example, the tagging error of the (Hana *et al.* 2004) tagger ran on the results of this analyzer decreased from 26.5% to 23% on all tokens, i.e., 13% relative reduction. (Feldman 2006) reports further improvement of the tagger – her error rate is 18.6%.

5 Conclusion

We have shown that a morphological system with a small amount of manually created resources can be successful.

Time needed for adjusting the system to a new (inflectional) language constitutes a fraction of the time needed for systems with extensive manually created resources: days instead of years. Two things are required – a reference grammar (for information about paradigms and closed class words) and a large amount of text (for learning a lexicon; e.g., newspapers from the internet). It is also advisable to have access to a native speaker. First, because reference grammars are often too vague, and second, because a quick glance at results (i.e., at an automatically acquired lexicon or at an analyzed text) can provide feedback leading to a significant increase of accuracy. Also, as Tables 2 and 12 show, providing (manually or semi-automatically) correct analyses for the most frequent words helps a lot. However, all of these require only limited linguistic knowledge.

The quality of the results is worse than that of systems with manual resources (roughly tripling recall error and doubling precision error). However, we believe that the approach still has a large space for improvement and that eventually the results will be very similar. Some of such enhancements were mentioned in §3.7.

In the near future, we plan to compare effectiveness (time and price) of our approach and the standard resource-intensive approach when annotating a medium-size corpus (e.g., 100K tokens). The resource-intensive system has lower ambiguity and error rate and therefore an annotator can work faster (less things to select from, less things to add). On the other hand, creation of such a system is very time consuming.

A Data

A.1 Corpora

During our work on Czech , we used several corpora for various purposes:

- a large raw corpus (`raw`) to train and tune our tools;
- annotated corpora (`te`) to test our tools;
- a small annotated corpus (`tu`) a to tune our tools;
- annotated corpora (`tr`, `tr1`, `tr2`) to report some statistics about Czech texts.

All of these corpora are either part of the Prague Dependency Treebank 1.0 (PDT, Böhmová *et al.* 2001) or are part of the PDT distribution. Let's discuss them in more detail:

- `Raw` consists of all the texts labeled as *Raw texts* in the PDT distribution.[16] The texts come from a Czech daily newspaper Lidové Noviny from the years 1991-1995. It contains over 39M tokens or nearly 2.4M sentences.
- `Te` consists of all the annotated texts labeled as evaluation data in PDT.[17] It consists of about 125K tokens or 8K sentences. The texts come from two daily newspapers, a business weekly and a popular scientific magazine.
- `Tr` consists of all the annotated texts labeled as training data in PDT. It consist of about 1.5M tokens or 95K sentences. The texts come from the same sources as the `Te` texts. To allow evaluation of how particular statistics transfers from one corpus to another, we split the corpus into two parts, each with about 620K tokens.[18] These smaller corpora are referred to as `tr1` and `tr2`. It is worth noting that we analyzed the `tr` corpus for the sake of this paper, and only after finishing our work on the tools. That means we did not use a source that would not be available for some other Slavic languages. The results are reported in §2.
- `Tu`. We need some annotated data to tune the parameters of our modules. The data should (1) be as close as possible to the data that would be obtained by morphologically annotating a large corpus; but (2) they should be also labor cheap. We decided to manually annotate a very small amount of data reflecting frequency of words in a corpus.

  From the `raw` corpus, we extracted word forms[19] and their frequencies. We split these words into groups by their frequency percentiles. From each of these groups

[16] See http://ufal.mff.cuni.cz/pdt/Corpora/Raw_Texts/index.html for more details.

[17] See http://ufal.mff.cuni.cz/pdt/Corpora/PDT_1.0/Doc/PDT10_data.html for more details.

[18] The remaining tokens are not used in this paper. PDT is organized by sources and date of publication. To prevent differences between the two corpora caused by such organization, we split the corpus into 40 pieces and put all the odd pieces into `tr1` and all the even pieces `tr2`.

[19] We ignore capitalization, forms differing only in capitalization are considered to be the same forms. It would be hard and probably unnecessary to address this issue properly. We also exclude all forms containing digits.

we randomly selected 10 noun forms. To favor more frequent words, i.e., words with a greater impact, the groups are between the following percentiles 0, 50, 75, 90, 95, 99, 100.[20] We morphologically annotated these forms. This is done semi-automatically – we select (in few cases add) the right analyses from the output of our Guesser. Naturally, we do not consider the context of those forms – so for example *hrad* 'castle' would be annotated as both nominative and accusative.

The number of occurrences of each form in the tuning corpus is the same as its frequency in the `raw` corpus.

A.2 Tagset

We used the Czech tag system employed in PDT (Hajic 2000). Every tag is represented as a string of 15 symbols each corresponding to one morphological category. We refer to the positions in such a string as slots. Two slots are not used (13,14); the slot 2 (detailed POS) uniquely determines the slot 1 (POS). For example, the word *viděl*o 'saw$_{\texttt{neut.sg}}$' is assigned the tag `VpNS---XR-AA---` because it is a verb (`V`), past participle (`p`), neuter (`N`), singular (`S`), does not distinguish case (`-`), possessive gender (`-`), possessive number (`-`), can be any person (`X`), is past tense (`R`), not gradable (`-`), affirmative (`A`), active voice (`A`), and is the basic stylistic variant (the final hyphen).

No.	Description	Abbr.	No. of values
1	POS	p	12
2	SubPOS – detailed POS	s	75
3	Gender	g	11
4	Number	n	6
5	Case	c	9
6	Possessor's Gender	f	5
7	Possessor's Number	m	3
8	Person	e	5
9	Tense	t	5
10	Degree of comparison	d	4
11	Negation	a	3
12	Voice	v	3
13	Unused		1
14	Unused		1
15	Variant, Style	i	10

Table 14: Overview of the Czech positional tagset

The tagset uses about 4300 tags. Thus, it is much larger than the Penn Treebank tagset, which uses only 36 non-punctuation tags (Marcus *et al.* 1993). It is also larger than

[20] We ignore the words with only one token (i.e., most of the forms below the median) because it would be hard to sort out spelling errors, etc.

case	
1	nominative
2	genitive
3	dative
4	accusative
5	vocative
6	locative
7	instrumental

number	
S	singular
P	plural
gender	
M	masculine animate
I	masculine inanimate
F	feminine
N	neuter

Table 15: Explanation of glosses.

the tagset we developed on its basis for Russian, a language similar to Czech – about 900 tags (see also Hana *et al.* 2004:§4.2).

A.3 Morphological Glosses

The morphological glosses in this paper are based on the tagset above. The noun glosses have the structure gender-number-case, for possible values see Table 15. For example, `FS2` stands for feminine singular genitive. When not relevant or obvious from the context we leave some of the slots out, e.g., `S2` – singular genitive. If a word is ambiguous we separate glosses by slashes. If case is ambiguous we simply list all the relevant case numbers, e.g., `S14` singular nominative or accusative.

B Czech

The Czech language is one of the West Slavic languages. It is spoken by 10+ million speakers mostly in Czechia. In this section, we discuss properties of morphology and syntax of the language relevant to our work. For a more detailed discussion, see for example (Karlík *et al.* 1996; Fronek 1999; Petr 1987).[21]

B.1 Morphology

Like other Slavic languages, Czech is a richly inflected language. The morphology is important in determining the grammatical functions of phrases. The inflectional morphemes are highly ambiguous (see Table 5). There are three genders: neuter, feminine and masculine. The masculine gender further distinguishes the subcategory of animacy. Sometimes, it is assumed that there are four genders: neuter, feminine, masc. animate and masc. inanimate; we follow that practice. In addition to singular and plural, some dual number forms survive in body part nouns and modifiers agreeing with them.[22] There are seven cases:

[21] Alas, there is no recent detailed Czech grammar in English. The dictionary (Fronek 1999) provides a basic overview. A short overview is also in an appendix of (Hana 2007).

[22] In colloquial Czech, there is no dual. The colloquial plural forms are the same as the official dual forms. For example, official: *velkýma rukama* 'big$_{\texttt{FD7}}$ hands$_{\texttt{FD7}}$' vs. *velkými lžícemi* 'big$_{\texttt{FP7}}$ spoons$_{\texttt{FP7}}$' (there is

	N Monday	F song	F fly	M Jirka	M brother	I castle
S1	pondělí	píseň	moucha	Jirka	bratr	hrad
S2	pondělí	písně	mouchy	Jirky	bratra	hradu
S3	pondělí	písni	mouše	Jirkovi	bratru/ovi	hradu
S4	pondělí	píseň	mouchu	Jirku	bratra	hrad
S5	pondělí	písni	moucho	Jirko	bratře	hrade
S6	pondělí	písni	mouše	Jirkovi	bratru/ovi	hradu
S7	pondělím	písní	mouchou	Jirkou	bratrem	hradem
P1	pondělí	písně	mouchy	Jirkové	bratři/ové	hrady
P2	pondělí	písní	much	Jirků	bratrů	hradů
P3	pondělí	písním	mouchám	Jirkům	bratrům	hradům
P4	pondělí	písně	mouchy	Jirky	bratry	hrady
P5	pondělí	písně	mouchy	Jirkové	bratři	hrady
P6	pondělích	písních	mouchách	Jircích	bratřích	hradech
P7	pondělími	písněmi	mouchami	Jirky	bratry	hrady

Table 16: Examples of declined nouns.

nominative (1), genitive (2), dative (3), accusative (4), vocative (5), locative (6), instrumental (7). It is a common practice to refer to the cases by numbers. Only nouns, only in singular, and only about half of the paradigms have a special form for vocative, otherwise the vocative form is the same as nominative.

There is a significant difference in morphology and lexicon between the official and colloquial levels of Czech. The official variant is a semi-artificial language. Sometimes it is claimed, with some exaggeration, that the official Czech is the first foreign language Czechs learn. Since we analyze written texts where the official language is predominant we largely ignore the colloquial language and its forms here.

Nouns. Traditionally, there are 13 basic noun paradigms – 4 neuter, 3 feminine, 4 animate and 2 inanimate; plus there are nouns with adjectival declension (another 2 paradigms). In addition, there are many subparadigms and subsubparadigms. All of this involves a great amount of irregularity and variation. As an illustration, Table 16 shows the declension of a few nouns. For discussion on noun paradigms see §3.3.

Adjectives. Adjectives follow two paradigms: *hard* and *soft*. Both of them are highly ambiguous, filling the 60 (4 genders × 2 numbers × 7 cases + 4) non-negated first grade categories with only 12, resp. 8 forms. See Table 17.[23]

no 'hands$_{\texttt{FP7}}$' or spoons$_{\texttt{FD7}}$); colloquial: *velkýma rukama* 'big$_{\texttt{FP7}}$ hands$_{\texttt{FP7}}$' vs. *velkýma lžícema* 'big$_{\texttt{FP7}}$ spoons$_{\texttt{FP7}}$'.

[23] In colloquial Czech, the hard declension is slightly different: in endings *ý* → *ej*, *é/í* → *ý*. Moreover, neuter in plural uses feminine forms, which in turn can be the same as masculine forms. There is no dual, and the inst. pl. has the same ending as the official dual.

	M	I	N	F	M	I	N	F
S1	mladý		mladé	mladá	jarní			
S2	mladého			mladé	jarního			jarní
S3	mladému			mladé	jarnímu			jarní
S4	mladého	mladý	mladé	mladou	jarního	jarní		
S5	mladý		mladé	mladá	jarní			
S6	mladém			mladé	jarním			jarní
S7	mladým			mladou	jarním			jarní
P1	mladí	mladé	mladá	mladé	jarní			
P2	mladých				jarních			
P3	mladým				jarním			
P4	mladé		mladá	mladé	jarní			
P5	mladí	mladý	mladá	mladé	jarní			
P6	mladých				jarních			
P7	mladými				jarními			
D7	mladýma				jarníma			

Table 17: Adjectival paradigms.

Negation and comparison forms are expressed morphologically. Negation by the prefix *ne-*, comparative by the suffix *-(e)jší-* and superlative by adding the prefix *nej-* to the comparative. The comparative and superlative forms are declined as soft adjectives.

Pronouns. Pronouns have either a noun or adjectival declension.

Numerals. Only *jeden* '1', *dva* '2', *tři* '3', and *čtyři* '4' fully decline, all of them distinguishing case and *jeden* and *dva* also gender. The inflection of the other cardinal numerals is limited to distinguishing oblique and non-oblique forms. Numerals expressing hundreds and thousands have in certain categories a choice between an undeclined numeral form or a declined noun form (*sto dvaceti*, *sta dvaceti* '120.genitive'). Ordinal complex numerals have all parts in the ordinal form[24] and fully declining (*dvacátý pátý* '25th'). Two-digit numerals may have an inverted one-word form (*pětadvacátý* '25th', lit: five-and-twentieth).

Verbs. Verbs distinguish three tenses. Only the forms of the present tense are marked inflectionally, distinguishing number and person. Inflection is also used to mark infinitive, past participles, passive participles and imperatives.

As in all Slavic languages, verbs also distinguish aspect – perfective and imperfective. Aspect is usually marked by prefixes, sometimes suffixes or by suppletion. Change of aspect is usually accompanied by a change, often subtle, in lexical meaning. For exam-

[24] Again, this is the case of the official language, complex numerals in colloquial Czech usually have only their tens and units in ordinal forms.

ple, *psát* 'write$_{\texttt{imp}}$', *napsat* 'write$_{\texttt{perf}}$', *dopsat* 'finish writing$_{\texttt{perf}}$', *sepsat* 'write up$_{\texttt{perf}}$', *sepisovat* 'write up$_{\texttt{imp}}$', etc.

Five main conjugational types are recognized. They are discriminated on the basis of the third person singular endings: (1) *-e*; (2) *-n-e*; (3) *-j-e*; (4) *-í*; (5) *-á*. Each class has several, quite similar, paradigms (6, 3, 2, 3, 1; 15 in total).

Certain categories are expressed analytically; various forms of the verb *být* serve as the auxiliary:

- past tense: present tense aux + past participle; auxiliary is omitted in 3rd person. E.g., *psal jsem* 'I wrote/was writing$_{\texttt{masc}}$'[25]
- future tense: future aux + infinitive. E.g., *budu psát* 'I will write'
- passive: present tense aux + pass. participle. E.g., *jsem obdivován* 'I am adored$_{\texttt{masc}}$'
- conditional: conditional aux + past participle. E.g., *psala bych* 'I would write$_{\texttt{fem}}$'
- past conditional: conditional aux + aux in past participle + past participle. E.g., *byla bych psala* 'I would have written$_{\texttt{fem}}$'

B.2 Syntax

Word order. Czech, like most other Slavic languages, has an exceptionally free word order. Unlike English, word order in Czech is used to express topic-focus structure (cf. Sgall *et al.* 1986) and definiteness. Thus for example, the words in sentence (3) can be rearranged in all 24 possible ways. Each of the sentences has a different topic-focus structure, but all of them are grammatically correct. Prototypically, old information precedes new information.

(3) Včera Petr viděl Marii.
yesterday Peter$_1$ saw Mary$_4$
'Yesterday, Peter saw Mary.'

More precisely, Czech word order is very free as regards the possibility of moving entire phrases; virtually any scrambling is possible. However, scrambling resulting in discontinuous phrases is much less common. It is limited to certain syntactic constructions, to many sentences involving a contrastive theme and to many sentences involving clitics (see, for example, Hana 2007.)

Agreement. Finite verbs agree with their subjects in number and person, participles in gender and number. Attributes agree with the nouns they modify in case, number and gender.

[25] The different position of the auxiliaries in these examples is due to the fact that some are clitics and some not.

Numeral expressions. Numerals expressions with *jeden* '1', *dva* '2', *tři* '3', *čtyři*, '4', *oba* 'both' behave in a "normal" way: a numeral agrees with its noun in case; *jeden*, *dva* and *oba* also in gender. However, numerals *pět* '5' and above in nominative or accusative positions are followed by nouns in genitive plural. The other cases behave the usual way.

Negation. Sentence negation in Czech is formed by the prefix *ne-* attached to the verb. As in the other Slavic languages, multiple negation is the rule, and negative subject or object pronouns, adjectival pronouns and adverbs combine with negative verbs.

(4) Nikdy nikomu nic neslibuj.
never to-nobody$_3$ nothing$_4$ not-promise$_{\texttt{imper}}$
'Never promise anything to anybody.'

References

BÖHMOVÁ, ALENA, JAN HAJIC, EVA HAJIČOVÁ, & BARBORA HLADKÁ. 2001. The Prague Dependency Treebank: Three-Level Annotation Scenario. In *Treebanks: Building and Using Syntactically Annotated Corpora*, ed. by Anne Abeillé. Kluwer Academic Publishers.

CHANG, JEFFREY T., HINRICH SCHÜTZE, & RUSS B. ALTMAN. 2002. Creating an Online Dictionary of Abbreviations from MEDLINE. *JAMIA* .

FELDMAN, ANNA. 2006. *Portable Language Technology: A Resource-light Approach to Morpho-syntactic Tagging*. The Ohio State University dissertation.

——, JIRKA HANA, & CHRIS BREW. 2006. A cross-language approach to rapid creation of new morpho-syntactically annotated resources. In *Proceedings of the fifth international conference on Language Resources and Evaluation (LREC 2006). Genoa, Italy*.

FRONEK, JOSEF. 1999. *English-Czech/Czech-English Dictionary*. Praha: Leda. Contains an overview of Czech grammar.

GOLDSMITH, JOHN. 2001. Unsupervised Learning of the Morphology of a Natural Language. *Computational Linguistics* 27.153–198.

HAJIC, JAN. 2000. Morphological Tagging: Data vs. Dictionaries. In *Proceedings of ANLP-NAACL Conference*, 94–101, Seattle, Washington, USA.

——. 2004. *Disambiguation of Rich Inflection: Computational Morphology of Czech*. Praha: Karolinum, Charles University Press. In press.

HANA, JIRI. 2007. *Czech Clitics in Higher Order Grammar*. The Ohio State University dissertation.

——, & ANNA FELDMAN. 2004. Portable language technology: The case of Czech and Russian. In *Proceedings from the Midwest Computational Linguistics Colloquium, June 25-26, 2004*, Bloomington, Indiana.

——, ——, & CHRIS BREW. 2004. A resource-light approach to Russian morphology: Tagging Russian using Czech resources. In *Proceedings of EMNLP 2004*, ed. by Dekang Lin & Dekai Wu, 222–229, Barcelona, Spain. Association for Computational Linguistics.

HANA, JIRKA, ANNA FELDMAN, LUIZ AMARAL, & CHRIS BREW. 2006. Tagging Portuguese with a Spanish tagger using cognates. In *Proceedings of the Workshop on Cross-language Knowledge Induction, 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL-2006), Trento, Italy*.

HLAVÁČOVÁ, JAROSLAVA. 2001. Morphological Guesser or Czech Words. In *Text, Speech and Dialogue*, ed. by V. Matoušek, Lecture Notes in Computer Science, 70–75. Berlin: Springer-Verlag.

KARLÍK, PETR, MAREK NEKULA, & Z. RUSÍNOVÁ. 1996. *Příruční mluvnice češtiny [Concise Grammar of Czech]*. Praha: Nakladatelství Lidové Noviny.

KOSKENNIEMI, KIMMO. 1983. Two-level model for morphological analysis. In *IJCAI-83*, 683–685, Karlsruhe, Germany.

——. 1984. A general computational model for word-form recognition and production. In *COLING-84*, 178–181, Stanford University, California, USA. Association for Computational Linguistics.

LARKEY, LEAH S., PAUL OGILVIE, M. ANDREW PRICE, & BRENDEN TAMILIO. 2000. Acrophile: an automated acronym extractor and server. In *DL '00: Proceedings of the fifth ACM conference on Digital libraries*, 205–214. ACM Press.

MARCUS, MITCHELL, BEATRICE SANTORINI, & MARY ANN MARCINKIEWICZ. 1993. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics* 19.313–330.

MIKHEEV, ANDREI. 1997. Automatic rule induction for unknown word guessing. *Computational Linguistics* 23.405–423.

——, & LIUBOV LIUBUSHKINA. 1995. Russian Morphology: An Engineering Approach. *Natural Language Engineering* 3.235–260.

PETR, JAN. 1987. *Mluvnice češtiny [Czech Grammar]*. Praha: Academia.

SGALL, PETR, EVA HAJIČOVÁ, & JARMILA PANEVOVÁ. 1986. *The Meaning of the Sentence and Its Semantic and Pragmatic Aspects*. Prague, Czech Republic/Dordrecht, Netherlands: Academia/Reidel Publishing Company.

SKOUMALOVÁ, HANA. 1997. A Czech morphological lexicon. In *Proceedings of the Third Meeting of the ACL Special Interest Group in Computational Phonology, Madrid*, 41–47. ACL.

YEATES, STUART. 1999. Automatic extraction of acronyms from text. In *New Zealand Computer Science Research Students' Conference*, 117–124.

——, DAVID BAINBRIDGE, & IAN H. WITTEN. 2000. Using compression to identify acronyms in text. In *DCC '00: Proceedings of the Conference on Data Compression*, p. 582. IEEE Computer Society.

ZIPF, GEORGE K. 1935. *The Psychobiology of Language*. Houghton-Mifflin.

—— 1949. *Human Behavior and the Principle of Least-Effort*. Addison-Wesley.