Applied Artificial Intelligence, 18:17–41, 2004
Copyright © Taylor & Francis Inc.
ISSN: 0883-9514 print / 1087-6545 online
DOI: 10.1080/08839510490250088

Taylor & Francis
Taylor & Francis Group

□ MACHINE LEARNING OF MORPHOSYNTACTIC STRUCTURE: LEMMATIZING UNKNOWN SLOVENE WORDS

TOMAŽ ERJAVEC and SAŠO DŽEROSKI
Department of Intelligent Systems,
Jožef Stefan Institute, Ljubljana, Slovenia

Automatic lemmatization is a core application for many language processing tasks. In inflectionally rich languages, such as Slovene, assigning the correct lemma (base form) to each word in a running text is not trivial, since for instance, nouns inflect for number and case, with a complex configuration of endings and stem modifications. The problem is especially difficult for unknown words, since word-forms cannot be matched against a morphological lexicon. This paper discusses a machine learning approach to the automatic lemmatization of unknown words in Slovene texts. We decompose the problem of learning to perform lemmatization into two subproblems: learning to perform morphosyntactic tagging of words in a text, and learning to perform morphological analysis, which produces the lemma from the word-form given the correct morphosyntactic tag. A statistics-based trigram tagger is used to learn morphosyntactic tagging and a first-order decision list learning system is used to learn rules for morphological analysis. We train the tagger on a manually annotated corpus consisting of 100,000 running words. We train the analyzer on open-class inflecting Slovene words, namely nouns, adjectives, and main verbs, together being characterized by more than 400 different morphosyntactic tags. The training set for the analyzer consists of a morphological lexicon containing 15,000 lemmas. We evaluate the learned model on word lists extracted from a corpus of Slovene texts containing 500,000 words, and show that our morphological analysis module achieves 98.6% accuracy, while the combination of the tagger and analyzer is 92.0% accurate on unknown inflecting Slovene words.

Lemmatization is a core functionality for various language processing tasks. It represents a normalization step on textual data, where all inflected forms of a lexical word are reduced to its common headword form, i.e., the lemma. This normalization step is needed in analyzing the lexical content of texts, e.g., in information retrieval, term extraction, machine translation, etc.

Thanks are due to Amebis, d.o.o. for ambiguously lemmatizing the ELAN corpus. Thanks also to Suresh Manadhar for earlier cooperation on learning morphology and for providing us with CLOG, and to Thorsten Brants for providing TnT.

Address correspondence to Tomaž Erjavec, Department of Intelligent Systems, Jožef Stefan Institute, Jamova 39, SI-1000 Ljubljana, Slovenia. E-mail: tomaz.erjavec@ijs.si

Lemmatization is relatively easy in English, especially if we are not interested in the speech part of a word. So called stemming can be performed with a lexicon which lists the irregular forms of inflecting words, e.g., *oxen* or *took*, while the productive ones, e.g., *wolves* or *walks*, can be covered by a small set of suffix stripping rules. The problem is much more complex for inflectionally rich languages, such as Slovene.

A precondition for correct lemmatization in inflectionally rich languages is determining the part-of-speech together with various other morphosyntactic features of the word-form. Adjectives in Slovene, for example, inflect for gender (3), number (3), and case (6), and, in some instances, also for definiteness and animacy. This, coupled with various morphophonologically induced stem and ending alternations, gives rise to a multitude of possible relations between a word-form and its lemma.

It should be noted that we take the term *lemma* to mean a word-form in its canonical form, e.g., infinitive for verbs, nominative singular for regular nouns, nominative plural for pluralia tantum nouns, etc. And while in English lemmas are almost invariably simply the stem of the word-form, this is in general not the case in Slovene. For example, the feminine gender noun form *postelje*$_{[pt,nom]}$ has *postelja*$_{[sg,nom]}$ (*bed*) as its lemma and this will also be its headword in a dictionary. However, the stem is *postelj-*, as the *-a* is already the inflectional morpheme for the singular nominative of (some) feminine nouns. Although determining the lemma from an inflected form is in this paper referred to as morphological analysis, it could also be viewed as a combination of analysis to identify the ending and isolate the stem, and morphological synthesis to join to it the appropriate canonical ending.

Using a lexicon with coded paradigmatic information, it is possible to lemmatize known words reliably but, in general, ambiguously. Unambiguous lemmatization of words in running text is only possible if the text has been tagged with morphosyntactic information, a task typically performed by so-called part-of-speech taggers (van Halteren 1999), which determine the part-of-speech (and other mophosyntactic information) of words in a text.

Much more challenging is the lemmatization of unknown words. This task, also known as "unknown word guessing," involves both morphological analysis and a part-of-speech tagging, where the two can be combined in various ways. One option is for the morphological analyzer to first try to determine the ambiguity class of the word, i.e., all its possible tags (and lemmas), which are then passed on to a part-of-speech tagger, which disambiguates from among the options. Alternatively, the analyzer can work in tandem with a tagger to directly determine the context dependent unambiguous lemma.

While results on open texts are quite good with hand-crafted rules (Chanod and Tapanainen 1995), there has been less work done with automatic induction of unknown word guessers. Probably the best known system of this kind is described in Mikheev (1997). It learns ambiguity classes from a lexicon and a raw (untagged) corpus. It induces rules for prefixes, suffixes and endings: The paper gives detailed analysis of accuracies achieved by combining these rules with various taggers. The best results obtained for tagging unknown words are in the range of 88%. However, the tests are performed on English language corpora and it is unclear what the performance as applied to lemmatization would be with inflectionally richer languages.

In this article, we discuss a machine learning approach to the automatic lemmatization of unknown words in Slovene texts. For this, we first learn to tag the words in a text with a tagger, where tags are morphosyntactic descriptions (MSDs), and then learn rules for morphological analysis, which produce the lemma from the word-form given its MSD.

A statistics-based trigram tagger, TnT (Brants 2000), is used to learn to perform MSD tagging. Crucially, the TnT tagger incorporates an unknown word-guessing module, which enables it to determine the tags (but not the lemmas) of unknown words with a fair amount of accuracy. The tagger is trained on a medium-sized manually annotated Slovene corpus, comprising 100,000 words, and its performance improved with a backup lexicon and various heuristics.

A system for learning first-order decision lists, named CLOG (Manandhar et al. 1998), is used to learn rules for morphological analysis. These rules cover nouns, adjectives, and verbs, which constitute the open class inflecting words and, hence, have the potential to morphologically analyze an unknown Slovene word. We do not need rules for the other word classes, as they are either closed, i.e., can be reliably covered by the lexicon (e.g., pronouns), or do not have productive inflections, i.e., their lemma form is always identical to the word-form (e.g., adverbs). For training the analyzer, we use a medium-sized Slovene lexicon, comprising 15,000 lemmas and their full inflectional paradigms.

Once we have trained the tagger and the morphological analyzer, unknown word-forms in a new text can be lemmatized by first tagging the text, then giving the word-forms and their corresponding MSDs to the morphological analyzer.

The work presented here builds on our previous experiments with lemmatizing unknown words in Slovene and extends them in several ways. The previous experiment (Džeroski and Erjavec 2000) was limited to a subclass of nouns and adjectives, and used a much smaller training as well as testing set; in this paper, we also show several ways to improve the tagging performance on new texts. If the previous results showed that our approach was valid in theory, we here produce a fully functional Slovene lemmatizer and evaluate it on an open domain.

THE LEXICAL DATA AND MORPHOSYNTACTIC DESCRIPTIONS

The training set we used for our experiment was extracted from a medium-sized lexicon of Slovene (Erjavec 1998), which had been produced in the scope of the MULTEXT-East project (Dimitrova et al. 1998; Erjavec 2001). The lexicon, together with the hand-annotated corpus discussed later is freely available for research purposes and can be obtained from http://nl.ijs.si/ME/V2/. It contains language resources not only for Slovene, but also for English, Romanian, Czech, Bulgarian, Estonian, and Hungarian.

A MULTEXT-East lexicon contains one entry per line. A lexical entry has three fields, separated by the tabulator character:

word-form ⟨TAB⟩ *lemma* ⟨TAB⟩ *MSD*

The *word-form* is the word, as it appears in the running text, modulo sentence initial capitalization, e.g., `diskreditirajmo`, `Moloha`. In the word-forms, as in the lemmas, SGML entities are used for the representation of non-ASCII characters, e.g., `samov&scaron;e&ccaron;ne&zcaron;` for *samovšečnež*. The *lemma* is the unmarked form of the word, e.g., `diskreditirati`, `Moloh`. In cases where the word-form is the lemma itself, the lemma is entered as "=."

The *MSD* is the morphosyntactic description of the word-form. The MSDs are provided as strings, using a linear encoding. In this notation, the first character denotes the part-of-speech; for the other characters in the string, the position corresponds to the part-of-speech determined attribute, and specific characters in each position indicate the value for that attribute. So, for example, the MSD `Vmmp1p` expands to `PoS`: `Verb`; `Type`: `main`; `VForm`: `imperative`; `Tense`: `present`; `Person`: `first`; and `Number`: `plural`. If a certain attribute does not apply either to a language, to a combination of attribute-values, or the specific lexical item, then the value of that attribute is a hyphen. For example, the `Person` attribute of `Verb` is not relevant for `Type`: `participle`, hence, `Vmps-sma` stands for `Verb main participle past` (no `Person`) `singular masculine active`. By convention, trailing hyphens are not included in the lexical MSDs.

To illustrate these points we give, in Figure 1, the lexical paradigm for the verb *gledati* (*to 1ook*) in MULTEXT-East format.

The syntax and semantics of the MULTEXT-East MSDs are given in the morphosyntactic specifications (Erjavec 2001), which have been developed in the formalism and on the basis of specifications for six Western European languages of the EU MULTEXT project (Bel et al. 1995); the MULTEXT project produced its specifications in cooperation with EAGLES, Expert Advisory

gleda	gledati	Vmip3s--n
gleda&scaron;	gledati	Vmip2s--n
gledajo	gledati	Vmip3p--n
gledal	gledati	Vmps-sma
gledala	gledati	Vmps-dma
gledala	gledati	Vmps-pna
gledala	gledati	Vmps-sfa
gledale	gledati	Vmps-pfa
gledali	gledati	Vmps-dfa
gledali	gledati	Vmps-dna
gledali	gledati	Vmps-pma
gledalo	gledati	Vmps-sna
gledam	gledati	Vmip1s--n
gledamo	gledati	Vmip1p--n
gledat	gledati	Vmu
gledata	gledati	Vmip2d--n
gledata	gledati	Vmip3d--n
gledate	gledati	Vmip2p--n
gledati	=	Vmn
gledava	gledati	Vmip1d--n
glej	gledati	Vmmp2s
glejmo	gledati	Vmmp1p
glejta	gledati	Vmmp2d
glejte	gledati	Vmmp2p
glejva	gledati	Vmmp1d

FIGURE 1. Sample lexical entries: Paradigm of "*gledati*" ("*to look*").

Group on Language Engineering Standards (Calzolari and McNaught 1996). The morphosyntactic specifications provide the grammar for the MSDs of the MULTEXT-East languages and are an attempt to standardize morphosyntactic encodings across languages. In addition to encompassing seven typologically very different languages, the structure of the specifications and of the MSDs makes them readily extensible to new languages.

To give an impression of the information content of the Slovene MSDs and their distribution, Table 1 gives, for each category, the number of attributes in the category, the total number of values for all attributes in the category and the number of different MSDs (i.e., combinations of attribute values) in the lexicon.

The Training Set

The Slovene MULTEXT-East lexicon contains about 15,000 lemmas chosen on the basis of one of their word-forms occurring in a 300,000 word corpus. This gives a lexicon of medium size, which displays good coverage of high and medium frequency words.

TABLE 1 Slovene Morphosyntactic Distribution

PoS	Attributes	Values	Lexicon
Noun	5	16	99
Verb	9	28	128
Adjective	7	23	279
Pronoun	11	38	1,335
Adverb	2	5	3
Adposition	3	8	6
Conjunction	2	4	3
Numeral	7	23	226
Interjection	0	0	1
Residual	0	0	1
Abbreviation	0	0	1
Particle	0	0	1
All	46	145	2,083

Rather than including only the word-forms encountered in the corpus, the more informative option of giving the complete inflectional paradigms of the lemmas was chosen. Since this generative approach depends on an internally given morphological model, the paradigms are not fully validated in practice, and, in some cases, tend to overgenerate. Spelling out full paradigms for a language as inflectionally rich as Slovene also leads to a large lexicon. However, this approach (rather than using some sort of morphological compression) has the advantage of keeping the underlying model as simple and explicit as possible. This makes it better for hand corrections, various experiments, and interchange.

In the work presented here, we are interested primarily in lemmatizing unknown, i.e., out of vocabulary words. For the training set, we therefore retained from the lexicon only those entries which inflect and belong to an open class part of speech. In other words, we discarded the grammatical words, namely prepositions, conjunctions, particles, pronouns, and the auxiliary verb, as well as uninflecting open-class words, i.e., adverbs (these inflect for degree, but this process is not productive), interjections, and abbreviations.

The training set thus retains the following three categories:

1. *Noun*: Either common or proper, which belongs to one of three genders and inflects for number (3, includes dual) and case (6); some forms also distinguish the (Boolean valued) category of animacy.
2. *Adjective*: Either qualificative, possessive, or ordinal, some of which inflect for degree (3), and all for gender (3), number (3), and case (6); some forms are further distinguished by the (Boolean valued) categories of animacy and definiteness.

3. *Main verb*: Which inflects for verb form (infinitive, supine, indicative, imperative, past, and passive participle) and, depending on the verb form, for person (3), gender (3), and number (3).

Table 2 gives quantitative data on the lexical training set. For each category, and for the full training set, we give the number of lexical entries, the number of word-forms (i.e., orthographically distinct strings) of different lemmas (i.e., entries marked with =), and the number of distinct MSDs.

MORPHOLOGICAL ANALYSIS

This section describes how the lexical training set was used to learn rules for morphological analysis. For this purpose we used an inductive logic programming (ILP) system that learns first-order decision lists, i.e., ordered sets of rules. We first explain the notion of first-order decision lists on the problem of synthesis of the past tense of English verbs, one of the first examples of learning morphology with ILP (Mooney and Califf 1995). We then lay out the ILP formulation of the problem of learning rules for morphological analysis of Slovene and describe how it was addressed with the ILP system Clog. The induction results are illustrated for an example MSD, and the sizes of the rule sets are given.

Learning Decision Lists

The ILP formulation of the problem of learning rules for the synthesis of past tense of English verbs considered in Mooney and Califf (1995) is as follows. A logic program has to be learned defining the relation `past` (`PresentVerb`, `PastVerb`), where `PresentVerb` is an orthographic representation of the present tense form of a verb and `PastVerb` is an orthographic representation of its past tense form. `PresentVerb` is the input and `PastVerb` the output argument. Given are examples of input/output pairs, such as `past` ([b,a,r,k], [b,a,r,k,e,d]) and `past` ([g,o], [w,e,n,t]). The program for the relation past uses the predicate `split` (A,B,C) as background knowledge: This predicate splits a list (of letters) `A` into two lists `B` and `C`, e.g., `split` ([b,a,r,k,e,d], [b,a,r,k], [e,d]).

TABLE 2 The Lexical Training Set

PoS	Entries	WForms	Lemmas	MSDs
Noun	124,998	60,133	7,278	99
Adjective	306,746	63,764	4,551	279
Main Verb	110,295	77,533	3,682	43
All	542,029	194,142	15,479	421

Given examples and background knowledge, Foidl (Mooney and Califf 1995) learns a first-order decision list defining the predicate `past`. A decision list is an ordered set of rules: Rules at the beginning of the list take precedence over rules below them and can be thought of as exceptions to the latter. An example decision list defining the predicate past is given in Figure 2.

The general rule for forming past tense is to add the suffix *-ed* to the present tense form, as specified by the default rule (last rule in the list). Exceptions to these are verbs ending on *-e*, such as *skate*, where *-d* is appended, and verbs ending in *-ep*, such as *sleep*, where the ending *-ep* is replaced with *-pt*. These rules for past tense formation are specified as exceptions to the general rule, appearing before it in the decision list. The first rule in the decision list specifies a lexical exception over which no generalization (using `split`) can be made: The past tense form of the irregular verb *go* is *went*.

Our approach is to induce rules for morphological analysis in the form of decision lists. To this end, we use the ILP system Clog (Manandhar et al. 1998). Clog shares a fair amount of similarity with Foidl: Both can learn first-order decision lists from positive examples only, an important consideration in NLP applications. Clog inherits the notion of *output completeness* from Foidl to generate implicit negative examples (see Mooney and Califf [1995]). Output completeness is a form of closed world assumption which assumes that all correct outputs are given for each given combination of input arguments' values present in the training set. Experiments show (Manandhar et al. 1998) that Clog is significantly more efficient than Foidl in the induction process. This enables Clog to be trained on much more realistic datasets, and therefore to attain higher accuracy.

Formulating the Problem and Background Knowledge

We formulate the problem of learning rules for morphological analysis of Slovene inflecting open class words in a similar fashion to the problem of learning the synthesis of past tense of English verbs. We use the triplets from the training lexicon, presented earlier, where each triplet is an example of analysis of the form `msd(orth, lemma)`. Within the learning setting of inductive logic programming, `msd(Orth, Lemma)` is a relation or predicate that consists of all pairs (word-form, lemma) that have the same morphosyntactic

```
past([g,o],[w,e,n,t]) :- !.
past(A,B) :- split(A,C,[e,p]), split(B,C,[p,t]), !.
past(A,B) :- split(B,A,[d]),   split(A,_,[e]), !.
past(A,B) :- split(B,A,[e,d]).
```

FIGURE 2. A first-order decision list.

description. `Orth` is the input and `Lemma` the output argument. A set of rules has to be learned for each of the `msd` predicates.

Encoding-wise, the MSD's part-of-speech is decapitalized and hyphens are converted to underscores. The word-forms and lemmas are encoded as lists of characters, with non-ASCII characters encoded as the names of SGML entities. In this way, the generated examples comply with PROLOG syntax. For illustration, the triplet *članki*/*članek*/`Ncmpn` gives rise to the following example:

```
ncmpn([ccaron, l, a, n, k, i], [ccaron, l, a, n, e, k]).
```

As shown in Table 2, we need to learn 421 different target predicates to cover all the open class inflecting words of Slovene.

Instead of the predicate `split/3`, the predicate `mate/6` (Figure 3) is used as background knowledge in Clog. `mate` generalizes `split` to deal also with prefixes (useful for analyzing superlative forms of Slovene adjectives, e.g., $najlepši_{[sup,m,sg,nom]}$ → $lep_{[pos,m,sg,nom]}$), and allows the simultaneous specification of the affixes for both input arguments.

The Induced Rules

The rules for morphological analysis were learned from the training set quantified in Table 2. In Table 3 we give, for each part of speech and overall, the number of MSDs, the number of all rules learned, and then split into the number of lexical exceptions and the number of generalizations using `mate/6`. The last column gives the CPU time necessary to induce the rules; the platform used was an HP UX server B.10.20 A 9000/780.

The rule sets vary substantially in size and complexity over different MSDs; the largest turn out to be the five imperative forms of verbs with 354 rules each, of these 201 exceptions and 153 generalizations. The smallest are the 56 rule sets for the inflections of ordinal adjectives, which are very regular and so have just one rule each.

As a specific example, consider the set of rules induced by Clog for analyzing the genitive singular of Slovene common feminine nouns. The training set for this concept contained 2,646 examples, from which Clog learned 22 rules of analysis. Nine of these were lexical exceptions, and are not interesting in the context of unknown word lemmatization. We list the generalizations in Figure 4.

From the bottom up, the first rule describes the analysis for nouns of the canonical first feminine declension, where the genitive singular ending *–e* is replaced by *–a* to obtain the lemma, e.g., *mize* → *miza*. The second rule deals with the canonical second feminine declension where *–i* is removed from the genitive to obtain the lemma, e.g., *peruti* → *perut*. The third rule attempts to

```
% split requires non-empty lists
split([X,Y|Z],[X],[Y|Z]).
split([X|Y],[X|Z],W) :- split(Y,Z,W).

% suffix remove
% word = stem+Y1 ; lemma = stem
mate(W1,W2,[],[],Y1,[]):-
    split(W1,W2,Y1).

% suffix add
% word = stem ; lemma = stem+Y2
mate(W1,W2,[],[],[],Y2):-
    split(W2,W1,Y2).

% suffix replace
% word = stem+Y1 ; lemma = stem+Y2
mate(W1,W2,[],[],Y1,Y2):-
    split(W1,X,Y1),
    split(W2,X,Y2).

% transfix
% word = P1+stem+Y1 ; lemma = P2+stem+Y2
mate(W1,W2,P1,P2,Y1,Y2):-
    split(W1,P1,W11),
    split(W2,P2,W22),
    split(W11,X,Y1),
    split(W22,X,Y2).

% total suppletion
% word = Y1 ; lemma = Y2
mate(W1,W2,[],[],W1,W2).
```

FIGURE 3. The definition of CLOG's `mate/6`.

cover nouns of the second declension that exhibit a common morpho-phonological alteration in Slovene, the schwa elision. Namely, if a schwa (weak *−e−*) appears in the last syllable of the word when it has the null ending, this schwa is dropped with non-null endings: *bolezni* → *bolezen*. The fourth rule

TABLE 3 Rule Sets Induced by CLOG

PoS	MSDs	Rules	Excpt	General	Time
Noun	99	4,973	3,275	1,698	9:39
Adjective	279	11,768	7,573	4,195	18:40
Main Verb	43	5,822	2,940	2,882	27:48
All	421	22,563	13,788	8,775	56:07

```
ncfsg(A,B):-mate(A,B,[],[],[n,o,v,e],[n,o,v,a]),!.
ncfsg(A,B):-mate(A,B,[],[],[e,v,e],[e,v,a]),!.
ncfsg(A,B):-mate(A,B,[],[],[a,v,e],[a,v,a]),!.
ncfsg(A,B):-mate(A,B,[],[],[r,v,e],[r,v,a]),!.
ncfsg(A,B):-mate(A,B,[],[],[i,v,e],[i,v,a]),!.
ncfsg(A,B):-mate(A,B,[],[],[e,s,n,i],[e,s,e,n]),!.
ncfsg(A,B):-mate(A,B,[],[],[i,s,l,i],[i,s,e,l]),!.
ncfsg(A,B):-mate(A,B,[],[],[v,e],[e,v]),!.
ncfsg(A,B):-mate(A,B,[],[],[z,n,i],[z,e,n]),!.
ncfsg(A,B):-mate(A,B,[],[],[i],[]),!.
ncfsg(A,B):-mate(A,B,[],[],[e],[a]),!.
```

FIGURE 4. Rules for the analysis of Slovene common feminine nouns in the singular genitive.

models a similar case with schwa elision, coupled with an ending alternation, which affects only nouns ending in *−ev*, e.g., *bukve* → *bukev*. The fifth and sixth rules again model schwa elision, but applied to second declension nouns (*misli* → *misel*; *dlesni* → *dlesen*), The last five rules all cover cases of first declension nouns, which would otherwise be incorrectly subsumed by the fourth *−ve* → *−ev* rule, e.g., *njive* → *njiva.*

As can be seen, the rules exhibit explanatory adequacy, as they can be easily linked to linguistic explanations. However, due to the limited background knowledge, the rules specify phonological generalizations (e.g., schwa elision) in a clumsy and repetitive manner.

THE TRAINING AND TESTING CORPORA

While we earlier discussed the lexical dataset, we turn now to the two corpus datasets we have used in our experiments: the Slovene part of the MULTEXT-East corpus, used for training the tagger, and the Slovene part of the IJS-ELAN corpus, used to evaluate our system. Both corpora are available via the WWW, the MULTEXT-East one for research purposes, and the IJS-ELAN without any restrictions.

The MULTEXT-East Corpus

The greatest bottleneck in the induction of a quality tagging model for Slovene is the lack of training data. The only available hand-validated tagged corpus is the MULTEXT-East corpus (Dimitrova et al. 1998; Erjavec 2001). This corpus consists of the novel *1984* by George Orwell, in the English original and in the Bulgarian, Czech, Estonian, Hungarian, Romanian, and Slovene translations. The corpus is sentence aligned and annotated with validated context disambiguated morphosyntactic descriptions and lemmas.

```
<s id="Osl.1.2.3.4">
  <w lemma="Winston" ana="Npmsn">Winston</w>
  <w lemma="se" ana="Px------y">se</w>
  <w lemma="biti" ana="Vcip3s--n">je</w>
  <w lemma="napotiti" ana="Vmps-sma">napotil</w>
  <w lemma="proti" ana="Spsd">proti</w>
  <w lemma="stopnica" ana="Ncfpd">stopnicam</w>
  <c>.</c>
</s>
```

FIGURE 5. Annotation in the MULTEXT-East corpus.

This makes it a good dataset for experiments in automatic tagging and lemmatization, even more so as it was the first such corpus for many of the languages involved. Despite its small size, it has been used for experiments on Romanian (Tufiş 1999), Hungarian (Varadi and Oravecs 1999), and Slovene (Džeroski and Erjavec 2000), or used for testing various approaches to tagging (Hajič 2000).

The corpus is encoded according to the recommendation of the Text Encoding Initiative, TEI P3 (Sperberg-McQueen and Burnard 1999). To illustrate the information contained in the corpus, we give the encoding of an example sentence in Figure 5. To give an indication of the properties of this training corpus, as well as the difference between Slovene and English, we give in Table 4 the number of word tokens in the corpus, the number of different word types in the corpus, i.e., of word-forms regardless of capitalization or their annotation, and the number of different context disambiguated lemmas and MSDs. The inflectional nature of Slovene is evident from the larger number of distinct word-forms and MSDs used in the corpus.

Since the time of its release on CD-ROM (Erjavec et al. 1998), errors and inconsistencies were discovered in the MULTEXT-East specifications and data, which were subsequently corrected. Most importantly, the English part, which had been only automatically tagged in the first release, was manually corrected. But because this work was done at different sites and in different manners, the corpus encodings had begun to drift apart. The EU project CONCEDE (Consortium for Central European Dictionary Encoding, 1998-2000) comprised most of the MULTEXT-East partners, offered the

TABLE 4 Inflection in the *1984* Corpus

	Slovene	English
Words	90,792	104,286
Forms	16,401	9,181
Lemmas	7,903	7,059
MSDs	1,010	134

possibility to bring the versions back on a common footing. The new version of the *1984* corpus and associated resources has been recently released (Erjavec 2001) and is freely available for research purposes. This version has been also used for training our tagger, as will be further explained in the next section.

The IJS-ELAN Corpus

In this section, we introduce the bilingual IJS-ELAN corpus, the Slovene part of which we have used as the target corpus in our experiment. This corpus was compiled in the scope of the EU ELAN project in order to serve as a widely distributable dataset for language engineering and for translation and terminology studies. IJS-ELAN is composed of fifteen recent terminology-rich texts and their translations; it contains one million words, about half in Slovene and half in English.

The first version of IJS-ELAN (Erjavec 1999) is sentence aligned and tokenized, but not tagged with morphosyntactic descriptions or lemmatized. Table 5 gives some quantitative measures from the corpus: The first line gives the number of translation segments, which, for the most part, correspond to sentences; the second line the number of punctuation tokens, <c> and the third of word tokens, <w>. The fourth line indicates the lexical stock of the corpus, as it gives the number of (decapitalized) word types. Finally, the fifth line excludes all the word types with a digit in them, thus removing numbers and similar "words." Given the richer inflection, Slovene, of course, exhibits a much large number of word types than does English.

Performing morphosyntactic annotation is relatively easy for English, as publicly available taggers and services exist to tag and lemmatize the corpus to a high level of accuracy. However, adding similar annotations to the Slovene half of the corpus is significantly more difficult.

As manual annotation would have been prohibitively expensive, we thus had a real need to investigate automatic means of annotating the Slovene part of the corpus, and of lemmatizing it. The next section reports on the tagging process, which was also a prerequisite for our lemmatization experiment. The evaluation of the lemmatization also takes advantage of the

TABLE 5 Size of the IJS-ELAN Corpus

	Slovene	English
Translation segments	31,900	31,900
Punctuation tokens	90,279	83,761
Word tokens	501,437	590,575
Word types	50,331	24,377
Lexical words	43,278	20,592

IJS-ELAN corpus, by extracting unknown words from it, lemmatizing them, and assessing the results.

TAGGING FOR MORPHOSYNTAX

Syntactic word-class tagging (van Halteren 1999), often referred to as part-of-speech tagging, has been an extremely active research topic in the last decade. Most taggers take a training set, where previously each token (word) had been correctly annotated with its part-of-speech, and learn a model of the language. This model enables them to more or less accurately predict the part-of-speech for words in new texts.

Some taggers learn the complete necessary model from the training set, while others must—or can—make use of background knowledge, in particular a morphological lexicon. The lexicon contains the possible morphological interpretations of the word-forms, i.e., their ambiguity classes. The task of the tagger is to assign the correct interpretation to the word-form, taking context into account. So, for example, the English word *looks* will have the ambiguity class consisting of the tags for third person singular present tense verb, and plural common noun. In the sentence "Every woman can make the most of her looks with our fully illustrated guide to personalized makeup" a tagger should assign to *looks* the nominal tag, while in "She looks great," the tag should be a verbal one.

In order to be able to annotate new texts we therefore need a training corpus, a trainable tagger, and preferably a wide-coverage lexicon. In this section, we explain the tagger we chose for our experiments, its training on the MULTEXT-East corpus, and how we performed the two-step annotation of the IJS-ELAN corpus from which we then generated our testing set.

Choosing a Tagger

For our experiments, we needed an accurate, fast, flexible, and robust tagger than would accommodate the large Slovene morphosyntactic tagset. Crucially, it also had to be able to tag unknown words, i.e., word-forms not encountered in the training set or background lexicon, since these, after all, are the word-forms we are learning to lemmatize.

In an evaluation exercise (Džeroski et al. 2000), we tested several different taggers on the Slovene *1984* corpus. They were: the Rule Based Tagger (RBT) (Brill 1995), the Maximum Entropy Tagger (MET) (Ratnaparkhi 1996), the Memory-Based Tagger (MBT) (Daelemans et al. 1996), and the Trigram Tagger TnT (Brants 2000). The comparative evaluation of RBT, MET, MBT, and TnT was performed by taking the body of *1984* as the training set, and its appendix ("The Principles of Newspeak") as the validation set. The evaluation took into account all tokens, words, and punctuation. While Džeroski et al. (2000) considered several different tagsets, we give here the results only for the maximal tagset, where tags are full MSDs.

These results indicate that accuracy is relatively even over all four taggers, at least for known words: The best result was obtained by MBT (93.6%), followed by RBT (92.9%), TnT (92.2%), and MET (91.6%). The differences in tagging accuracies over unknown words are more marked: Here TnT leads (67.5%), followed by MET (55.9%), RBT (45.4%), and MBT (44.5%). Apart from accuracy, the question of training and tagging speed is also quite important, especially when experimenting with different training regimes and when tagging large amounts of text. Here RBT was by far the slowest (three days for training), followed by MET, with MBT and TnT being very fast (both less than one minute).

Given the above assessment, we chose for our experiments the TnT tagger: It exhibits good accuracy on known words, excellent accuracy on unknown words, and is robust and efficient. In addition, it is easy to install and run, and incorporates several methods of smoothing and handling unknown words. Our choice has since been confirmed by other experiments (Zavrel et al. 2000; Megyesi 2001).

At this point, it is also worth explaining the strategy TnT uses to tag unknown words, as this capability is, of course, crucial for the complete lemmatization system to work. In the training phase, TnT collects all the hapax words (words that appear only once in the training corpus) and uses them to approximate the behavior of unknown words. TnT builds a suffix tree from the hapax words and associates each suffix with its ambiguity class, i.e., with the set of the tags of the words ending with the suffix. Then, on encountering an unknown word in a text to tag, TnT matches the longest stored suffix to the word, and in this manner predicts its ambiguity class. It then proceeds to disambiguate the tag of the word in the usual manner.

As can be seen, the unknown word guessing module of TnT already performs a kind of simple form-driven morphological analysis, by associating ambiguity classes with certain word endings. However, lemmas cannot be assigned to word-forms in this module, and disambiguation and further morphological analysis is required. Needless to say, errors introduced in the tagging of unknown words will lead to incorrect functioning of the second stage of the lemmatization, but, as will be seen later on, the lemmatization is, to a certain extent, tolerant to faults in the MSD assignments made by the tagger.

Learning the Initial Tagging Model

The Slovene *1984* was first converted to TnT training format, where each line contains just the (lowercased) token and its correct tag. For word tags, we used their MSDs, while punctuation marks were tagged as themselves. This gave us a tagset of 1,024, comprising the sentence boundary, 13 punctuation tags, and the 1,010 MSDs.

Training the TnT tagger produces a table of MSD n-grams (n=1,2,3) and a lexicon of word-forms together with their frequency annotated ambiguity classes. The n-gram file for our training set contained 1,024 uni-, 12,293 bi-, and 40,802 trigrams, while the lexicon contains 16,415 entries. Example stretches from the n-gram and lexicon file are given in Figure 6.

The excerpt from the n-gram file states that the tag `Vcps – sma` appeared 544 times in the training corpus. It was followed by the tag `Vcip3s – –n` 82 times. The triplet `Vcps – sma`, `Vcip3s – –n`, `Afpmsnn` appeared 17 times.

The excerpt from the lexicon file states that the word-form `juhe` appeared in the corpus twice and was tagged `Ncfsg` in both cases. The word-form `julija` appeared 59 times and was tagged 58 times as `Npfsn` (lemma `julija` = *Julia*) and once as `Ncmsa – –n` (lemma `julij` = *July*). The ambiguity class of the word-form `julija` is thus the tagset {`Npfsn`, `Ncmsa – –n`}. Incidentally, this ambiguity class also reveals the weakness of our training set: since it consists of only one text, it does not offer a good sample of general language. So, while *July* will be typically much more common that *Julia*, the situation is reversed in *1984*.

We then tested the performance of the TnT tagger on the IJS-ELAN corpus. On a small test set (cca 1,000 words), it was found that the overall accuracy was only approximately 70%. We therefore tried to improve this rather low accuracy.

Refining the Tagging

The low accuracy of the tagging can be traced both to the inadequate lexicon of the tagger, as more than a quarter of all word tokens in IJS-ELAN were

```
...
Vcps-sma        544
 Vcip3s--n       82
        Afpmsnn 17
        Aopmsn   2
        Ncmsn   12
        Npmsn    1
        Css      2
        Afpnpa   1
        Q        3
...
```

(a)

FIGURE 6. Excerpts from the a) n-gram and b) lexicon files generated by the TnT tagger.

```
...
juhe      2    Ncfsg       2
julij     1    Npmsn       1
julija   59    Npfsn      58   Ncmsa--n  1
julije    4    Npfsg       4
juliji   10    Npfsd      10
julijin   4    Aspmsa--n  2   Aspmsn    2
...
```

(b)

FIGURE 6. (Continued).

unknown, as well as to trigrams applied to very different text types than the novel used for training. To offset these shortcomings, we employed two methods, one primarily meant to augment the n-grams, and the other the lexicon.

It is well known that "seeding" the training set with validated samples from the texts to be annotated can significantly improve results. We selected a sample comprising 1% of the corpus segments (approx. 5,000 words) evenly distributed across the whole of the corpus. The sample was then manually validated and corrected, also with the help of Perl scripts, which pointed out certain typical mistakes, e.g., the failure of case, number, and gender agreement between adjectives and nouns.

The tagger n-grams were then relearned using the concatenation of the Slovene *1984* with the validated ELAN sample. The resulting model contains 1,083 uni-, 13,468 bi-, and 46,183 trigrams. Note that, in comparison to the initial model, this one contains 59 new MSDs; these arise from inflected forms (mainly numerals, adjectives, and pronouns), which had not previously occurred in the training data. It is not really surprising that new MSDs are encountered with the enlargement of the training set: As can be seen in Table 1, the full (lexical) set of Slovene MSDs numbers over 2,000, but only half of them (Table 4) appear in the *1984* corpus.

Second, it has been shown (Hajič 2000) that a good lexicon is much more important for quality tagging of inflective languages than the higher-level models, e.g., bi- and trigrams. A word that is included in a TnT lexicon gains the information on its ambiguity class (i.e., the set of context-independent possible tags) as well as the lexical probabilities of these tags.

The Slovene part of the ELAN corpus was therefore first lexically annotated, courtesy of the company Amebis, d.o.o. (which produces the MS Word spelling checker for Slovene). The large lexicon used covers most of the words in the corpus; only 3% of the tokens remain unknown. This lexical annotation includes not only the MSDs, but also, paired with the MSDs, the possible lemmas of the word-form.

We first tried using a lexicon derived from these annotations directly as a backup lexicon with TnT. While the results were significantly better than with the first attempt, a number of obvious errors remained and additional new errors were introduced at times. Most of the mistakes had to do with tagging a word for dual number. While the morphological forms for this number are perfectly legitimate (and productive) for Slovene inflecting words, they occur only rarely in actual texts. The reason so many words were tagged for dual stems from the fact that the tagger is often forced to fall back on unigram probabilities, but the backup lexicon contains only the ambiguity class, with the probabilities of the competing tags being evenly distributed. And as the tags in an ambiguity class are alphabetically sorted, and dual (d) occurs before plural or singular, TnT assigned to the word-form the first tag available.

To remedy the situation, a heuristic was used to estimate the lexical frequencies of unseen words as follows. First, both disambiguated and lexical annotations from the training corpus were used to create an example lexicon. As the MSDs for the word-forms came from the lexicon, the ambiguity classes were complete, in the sense that they contained all possible MSDs; and because the MSDs were also collected from the disambiguated annotations, the ambiguity classes contained empirical frequencies for the MSDs. To normalize lexical frequencies, we then added together all the frequencies for identical ambiguity classes. Finally, for words unseen in the training corpus, we simply substituted their lexical frequencies with the summed frequencies for the ambiguity class is question.

To take an actual example from the corpus, the word-forms *kasnejšimi*, *manjšimi*, *revnejšimi* (*later*, *smaller*, *poorer*) all appear in the training corpus, and have the lexical ambiguity class `Afcfpi Afcmpi Afcnpi`, i.e., adjective qualificative comparative feminine/ masculine/neuter plural instrumental. In the corpus, *kasnejšimi* and *revnejšimi* each appear once and are tagged as `Afcfpi`, while *manjšimi* appears twice and is once also tagged as `Afcfpi`, and once as `Afcmpi`. The example frequencies for this ambiguity class are therefore 1,2,0, and all the (15) unseen word-forms from the lexicon with this ambiguity class are assigned the corresponding weights.

Using this lexicon and the seeded model, we then re-tagged the Slovene part of the IJS-ELAN corpus. We manually validated a small sample of the tagged corpus, consisting of around 5,000 tokens. Table 6 gives the accuracy separately over adjectives, nouns, and main verbs over all of these open-class inflecting words, as well as over all words, and, finally, all the tokens (words plus punctuation symbols) in the sample.

As has been mentioned, the lexical annotations included lemmas along with the MSDs. Once the MSD disambiguation had been performed, it was therefore trivial to annotate the words with their lemmas. But while all the words, known as well as unknown, have been annotated with an MSD, there remains approximately 3% of the corpus word tokens which are not included in the backup lexicon and hence do not have lemma annotations.

TABLE 6 IJS-ELAN Tagging Accuracy

	All	Errors	Accuracy
Nouns	1276	133	89.6%
Adjectives	499	77	84.6%
Main Verbs	505	17	96.6%
Open	2280	227	90.0%
Words	3820	318	91.7%
Tokens	4454	318	92.9%

EVALUATING THE SYSTEM

The entire experimental procedure of this study, including the learning and evaluation of the analyser, tagger, and lemmatizer, is outlined in Figure 7.

This section describes the evaluation of the proposed lemmatization system, first on a large number words unknown to the analyzer, and then on words unknown to both the tagger and the morphological analyzer.

Evaluating the Analyzer

We first evaluated the performance of the morphological analyzer on unknown words. To this end, we extracted from the tagged and lemmatized IJS-ELAN corpus all the open class inflecting word tokens which are

1. From the MULTEXT-East Lexicon (MEL) (Table 2) for each MSD in the open word classes:
   Learn rules for morphological analysis using CLOG (Table 3).
2. From the MULTEXT-East *1984* tagged corpus (MEC) (Table 4):
   Learn a tagger T0 using TnT (Figure 6).
3. From IJS-ELAN untagged corpus (IEC) (Table 5), take a small subset S0 (of cca 1000 words):
   Evaluate performance of T0 on this sample (~70%–quite low).
4. From IEC, take a subset S1 (of cca 5,000 words) and manually tag and validate it:
   Learn a tagger T1 from MEC $\cup$ S1 using TnT.
5. Use a large backup lexicon (AML) that provides the ambiguity classes:
   Lematize IEC using this lexicon and estimate the frequencies of MSDs within ambiguity classes using the tagged corpus MEC $\cup$ S1.
6. From IEC, take a subset S2 of (cca 5,000 words), tag it with T1 + AML yielding IEC-T, and manually validate it:
   This gives an estimate of tagging accuracy (Table 6).
7. Take the tagged and lemmatized IEC-T, extract all open class inflecting word tokens which posses a lemma (were in the AML lexicon), yielding the set AK; those that do not possess a lemma, go to LU.
8. Test the analyzer on AK (Table 7).
9. Test the lemmatizer (consisting of the tagger + analyzer) on LU (Table 8).

FIGURE 7. An outline of the experimental procedure.

annotated with their lemma, i.e., were included in the large backup lexicon used for tagging, but are not part of the MULTEXT-East lexicon, which had been used for training the CLOG analyzer. The extracted lexicon is in format identical to the MULTEXT-East lexicon: Each entry contains the word-form from the corpus, and the (automatically assigned) lemma and MSD. It should be noted that even though the accuracy of MSD tagging for the corpus is around 90% (cf. Table 6), the assigned lemmas are—given that they came paired with MSDs from a lexicon—by definition correct for the MSD assigned by the tagger. This evaluation therefore uses real corpus data, but is not penalized by tagging errors.

Each test lexical entry, i.e., triplet word-form, lemma, and MSD, is distinct, even though it might occur in the corpus more than once. We thus do not favor high-frequency words in the corpus. The validation lexicon has over 10,000 entries, and Table 7 summarizes the accuracy over the three categories and overall. We give the total number of test lexicon entries, the number of erroneously lemmatized ones, the accuracy percentage, and, in the last column, the baseline accuracy, i.e., what the accuracy would be if we simply assigned the word-form as the lemma.

An analysis of the mistakes reveals various causes of errors. Especially with adjectives, by far the most prevalent is caused by foreign words which do not inflect according to the usual rules of Slovene; more than two thirds of the errors are due to two uninflecting adjectives *neto* (*net*) and *bruto* (*gross*). With nouns, the most common type of error is pluralia tantum nouns, which have the lemma in nominative plural, but the system assigns them the (nonexistent) singular form. Also with nouns, a frequent mistake is to either posit a schwa elision where there isn't one, or not to posit it when it exists. Finally, verb mistakes have mostly to do with assigning to the verb the wrong paradigmatic class. Given that dealing with these errors would involve access to information which is not present in the orthographic form, it is difficult to see any simple means of improving our present accuracy.

Evaluating the Lemmatizer

In order to test the system as a whole, we concentrated on the words of the corpus which had not been assigned a lemma during the tagging process,

TABLE 7 Accuracy of Morphological Analyzer

PoS	Entries	Error	Accuracy	Baseline
Noun	4,834	85	98.2%	31.9%
Adjective	4,764	50	98.9%	8.9%
Main Verb	588	10	98.3%	11.9%
All	10,186	145	98.6%	20.0%

i.e., were not included in the backup lexicon. This task thus combined both tagging of unknown words as well as their subsequent lemmatization. The resulting lemmatizations were then hand validated and assessed.

As with the first evaluation, we first extracted the test lexicon, choosing those word-forms that had been tagged as a noun, adjective, or main verb and do not have an assigned lemma. Note that validating over such a lexicon can measure accuracy, but not recall; if an unknown noun is erroneously tagged as, say, a numeral, this fact goes unnoticed.

We further reduced the test lexicon by excluding entries whose word-forms contain non-alphanumeric characters (except hyphen), are less than four characters long, and appear only once in the corpus. These conditions are meant to ensure we do not try to lemmatize Web addresses, formulas, typing mistakes, and similar "words." Finally, we ran the list through an English spell checker and by means of this eliminated all the English words; the IJS-ELAN corpus contains two computer-related texts, so there was a fair number of these. The remaining test lexicon has 763 entries, each consisting of a word-form and an automatically assigned MSD; the list has 695 distinct word-forms.

On this list, we then ran our morphological analyzer and manually checked the proposed lemmatizations. It should be noted that this is a difficult process even for a human, and often involves checking with a dictionary and looking at the wider context of the word-form. The results of the automatic lemmatization are given in Table 8.

Unsurprisingly, the analysis of errors shows that it is the tagger that is almost invariably responsible for the error. First, it should be noted that the tagger can make an error in the inflectional features of the word-form, and the analyzer will still, in many cases, be able to determine the correct lemma, as many inflectionally distinct forms have identical surface representations. This is the reason why the lemmatizer performance on unknown words is higher even than the performance of the tagger on known words, estimated at 90%, cf. Table 6.

However, the error tolerance does not hold for lexeme-inherent features, such as part-of-speech or noun gender; the (meaningless) word-form *grauba* with the stem *graub-* will have the lemma *graub* if it a masculine noun, *grauba*

TABLE 8 Accuracy of Lemmatization on Unknown Words

PoS	Entries	Error	Accuracy
Noun	405	36	91.1%
Adjective	308	16	94.8%
Main verb	50	9	82.0%
All	763	61	92.0%

if it is a feminine noun, and *graubati* if it is a verb. In Table 9, we give a breakdown for the reasons of lemmatization mistakes. As can be seen, the most common tagger mistake is assigning a wrong gender to a noun. For unknown words, this is a difficult task even for humans, as the context is often not informative enough (e.g., by providing an adjective, which will agree with the noun in its gender) to enable determining the gender. Similarly difficult is determining the type of an inflected adjective. Namely, qualificative adjectives have identical forms to ordinal ones, except in the lemma form; and it is the meaning, rather than the context of the adjective, that determines its type. The third most common type of mistake is misassigning the part-of-speech to a word, in most cases by verbing a noun. Finally, we have eight cases of foreign words (such as *kiwi*) which have a lemma that is unexpected for Slovene inflections.

As a final point, we should note that while most of the above errors result from the analyzer producing the wrong lemma, there are ten cases where the analysis simply failed, i.e., the word-form, given the posited MSD, could not be resolved into a lemma. Invariably, these failures pointed to tagging mistakes, meaning that, at least in some cases, the analyzer can act as validation aid for tagging.

CONCLUSIONS

We have addressed the problem of lemmatization of unknown words, in particular nouns, adjectives, and verbs, in Slovene texts. This is a core normalization step for many language processing tasks expected to deal with unrestricted texts. We approached this problem by combining a morphological analyzer and a morphosyntactic tagger. The language models for the components were inductively learned from the slovene parts of the MULTEXT-East 15,000 lemma morphological lexicon and 100,000 word tagged corpus, respectively. To improve accuracy, the tagging model was supplemented by a large background lexicon.

We tested the learned analyzer and tagger separately and in combination on an open domain test set, namely the Slovene part of the IJS-ELAN

TABLE 9 Causes of Lemmatization Errors

Reason	Number
Noun gender	30
Adjective type	13
Part-of-speech	8
Foreign word	8
Other	2
All	61

500,000 word corpus. The tagger achieves an accuracy of 92.9% over all tokens in the test set, and 90.0% over nouns, adjectives, and verbs only. The analyzer was tested on a large sample of unknown words from the corpus, and achieved an accuracy of 98.6%. Finally, we tested the system on words that were unknown both to the analyzer and tagger; the resulting accuracy of the lemmatization was 92.0%

The combination of the morphological analyzer and the tagger is performed in a novel way. Typically, the results of morphological analysis would be given as input to a tagger. Here, we give the results of tagging to the morphological analyzer: An unknown word-form appearing in a text is passed on to the analyzer together with its morphosyntactic tag produced by the tagger. Of course, this approach relies on the tagger itself incorporating an unknown word-guessing module, although this module need only predict tags, not the lemmas themselves (as is the case in TnT).

Except for our own work (Džeroski and Erjavec 2000), there are, to our knowledge, no published results for lemmatization of unknown words in Slovene, or even other slavic languages, so it is difficult to give a comparable evaluation of the results. In comparison with our previous work, we have significantly extended the coverage of the system and more than halved the error rate, due primarily to increasing the size of training set for the analyzer and to building a better tagging model.

There are various options on how to improve our current accuracy. The most drastic improvement would be achieved by bettering the performance of the tagger by enlarging the rather small training corpus. However, accurate annotation of corpora is a very time-consuming task. Another obvious step would be to take advantage of the fact that new words often appear more than once in a text. So, instead of lemmatizing each word occurrence on its own, the evidence could be gathered for all the similar word-forms and the results combined to reduce the error rate.

Another option would be to combine the morphological analyzer with the tagger in a more standard fashion and which, to an extent, is already incorporated in the TnT tagger. Here we use morphological analyzer first to help the tagger postulate the ambiguity classes for unknown words. While this proposal might sound circular, we can view as the lemmatizer and the tagger each imposing certain constratints on the context dependent triplet of word-form, lemma, and MSD. It is up to further research to discover in which way such constraints are best combined. It would also be interesting to compare our approach to morphological analysis, where rules are learned separately for each morphosyntactic description (MSD) to an approach where rules are learned for all MSDs of a word class together.

Nevertheless, even with the current level of accuracy of 92%, the lemmatization approach proposed is already quite useful; we have, so far, applied it as an aid to the creation and updating of lexica from language corpora.

REFERENCES

Bel, N., N. Calzolari, and M. Monachini (eds.). 1995. Common Specifications and Notation for Lexicon Encoding and Preliminary Proposal for the Tagsets. MULTEXT Deliverable D1.6.1B, ILC, Pisa.

Brants, T. 2000. TnT-A Statistical Part-of-Speech Tagger. In *Proceedings of the Sixth Applied Natural Language Processing Conference* (*ANLP-2000*), pages 224–231, Seattle, WA. http://www.coli.uni-sb.de/˜thorsten/tnt/.

Brill, E. 1995. Transformation-based error-driven learning and natural language processing: A case study in part-of-speech tagging. *Computational Linguistics* 21(4):543–565.

Calzolari, N., and J. McNaught (eds.). 1996. Synopsis and Comparison of Morphosyntactic Phenomena Encoded in Lexicons and Corpora: A Common Proposal and Applications to European Languages. EAGLES Document EAG—CLWG—MORPHSYN/R, ILC, Pisa. http://www.ilc.pi.cnr.it/EAGLES/home.html.

Chanod, J.-P., and P. Tapanainen. 1995. Creating a Tagset, Lexicon and Guesser for a French Tagger. In *Proceedings of the ACL SIGDAT workshop From Text to Tags: Issues in Multilingual Language Analysis*, pages 58–64, Dublin, Ireland.

Daelemans, W., J. Zavrel, P. Berck, and S. Gillis. 1996. MBT: A memory-based part of speech tagger-generator. In *Proceedings of the Fourth Workshop on Very Large Corpora*, Copenhagen, eds. E. Ejerhed and I. Dagan, pages 14–27.

Dimitrova, L., T. Erjavec, N. Ide, H.-J. Kaalep, V. Petkevič, and D. Tufis. 1998. Multext-east: Parallel and comparable corpora and lexicons for six central and eastern european languages. In *Proceedings of the COLING-ACL '98*, pages 315–319, Montréal, Quebec, Canada. http://nl.ijs.si/ME/.

Džeroski, S., and T. Erjavec, 2000. Learning to lemmatise Slovene words. In eds. J. Cussens and S. Džeroski, *Learning Language in Logic*, Number 1925 in Lecture notes in artificial intelligence, Berlin: Springer-Verlag.

Džeroski, S., T. Erjavec, and J. Zavrel. 2000. Morphosyntactic tagging of Slovene: Evaluating PoS taggers and tagsets. In *Proceedings of the Second International Conference on Language Resources and Evaluation (LREC'00)*, pages 1099–1104, Paris, France.

Erjavec, T. 1998. The multext-east Slovene lexicon. In *Proceedings of the 7th Slovene Electrotechnical Conference* (ERK '98) pages 189–192, Portorož, Slovenia. http://nl.ijs.si/et/Bib/ERK98/.

Erjavec, T. 1999. The ELAN Slovene-English aligned corpus. In *Proceedings of the Machine Translation Summit VII*, pages 349–357, Singapore. http://nl.ijs.si/elan/.

Erjavec, T. (ed.). 2001a. Specifications and Notation for MULTEXT-East Lexicon Encoding. MULTEXT-East Report, Concede Edition D1.1F/Concede, Institute Jožef Stefan, Ljubljana. http://nl.ijs.si/ME/V2/msd/.

Erjavec, T. 2001b. Harmonised morphosyntactic tagging for seven languages and Orwell's 1984. In *Proceedings of the 6th Natural Language Processing Pacific Rim Symposium* (*NLPRS'01*), pages 487–492, Tokyo, Japan. http://nl.ijs.si/ME/V2/.

Erjavec, T. 2002. The IJS-ELAN Slovene-English parallel corpus. *International Journal of Corpus Linguistics 7*(1):1–20.

Erjavec, T., A. Lawson, and L. Romary. 1998. East meets West: Producing multilingual resources in a European context. In *Proceedings of the First International Conference on Language Resources and Evaluation* (LREC'98), pages 233–240, Granada.

Hajič, J. 2000. Morphological tagging: Data vs. dictionaries. In *Proceedings of the ANLP/NAACL 2000*, pages 94–101, Seattle, WA.

Manandhar, S., S. Džeroski, and T. Erjavec. 1998. Learning multilingual morphology with CLOG. In *Proceedings of Inductive Logic Programming: 8th International Workshop* (*ILP-98*), Number 1446 in Lecture Notes in Artificial Intelligence, ed. D. Page, pages 135–144. Berlin, Springer-Verlag.

Megyesi, B. 2001. Comparing data-driven learning algorithms for PoS tagging of Swedish. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing* (*EMNLP 2001*), pages 151–158, Pittsburgh, PA.

Mikheev, A. 1997. Automatic rule induction for unknown-word guessing. *Computational Linguistics 23*(3): 405–424.

Mooney, R. J., and M. E. Califf. 1995. Induction of first-order decision lists: Results on learning the past tense of English verbs. *Journal of Artificial Intelligence Research 3*(1):1–24.

Ratnaparkhi, A. 1996. A Maximum entropy part of speech tagger. In *Proceedings of the ACL-SIGDAT Conference on Empirical Methods in Natural Language Processing*, pages 491–497, Philadelphia, PA.

Sperberg-McQueen, C. M., and L. Burnard (eds.). 1999. *Guidelines for Electronic Text Encoding and Interchange, Revised Reprint*. The TEI Consortium. http://www.teic.org/.

Tufiş, D. 1999. Tiered tagging and combined language model classifiers. In *Text, Speech and Dialogue*, Number 1692 in Lecture Notes in Artificial Intelligence, eds. F. Jelinek and E. Noth, pages 28–33. Berlin: Springer-Verlag.

van Halteren, H. (ed.). 1999. *Syntactic Wordclass Tagging*. Dordrecht: Kluwer Academic Publishers.

Varadi, T., and C. Oravecs. 1999. Morpho-syntactic ambiguity and tagset design for Hungarian. In *Proceedings of the EACL-99 Workshop on Linguistically Interpreted Corpora (LINC-99)*, pages 8–12, Berger.

Zavrel, J., F. van Eynde, and W. Daelemans. 2000. Part of speech tagging and lemmatisation for the spoken Dutch corpus. In *Proceedings of the second International Conference on Language Resources and Evaluation* (*LREC'00*), pages 1427–1433, Athens.