The Effects of Syntactic Features in Automatic Prediction of Morphology

Wolfgang Seeker and **Jonas Kuhn**
Institute for Natural Language Processing
University of Stuttgart
{seeker,jonas}@ims.uni-stuttgart.de

Abstract

Morphology and syntax interact considerably in many languages and language processing should pay attention to these interdependencies. We analyze the effect of syntactic features when used in automatic morphology prediction on four typologically different languages. We show that predicting morphology for languages with highly ambiguous word forms profits from taking the syntactic context of words into account and results in state-of-the-art models.

1 Introduction

In this paper, we investigate the interplay between syntax and morphology with respect to the task of assigning morphological descriptions (or tags) to each token of a sentence. Specifically, we examine the effect of syntactic information when it is integrated into the feature model of a morphological tagger. We test the effect of syntactic features on four languages – Czech, German, Hungarian, and Spanish – and find that syntactic features improve our tagger considerably for Czech and German, but not for Hungarian and Spanish. Our analysis of constructions that show morpho-syntactic agreement suggests that syntactic features are important if the language shows frequent word form syncretisms[1] that can be disambiguated by the syntactic context.

The meaning of a sentence is structurally encoded by morphological and syntactic means.[2] Different languages, however, use them to a different extent. Languages like English encode grammatical information (like the subject vs object status of an argument) via word order, whereas languages like Czech or Hungarian use different word forms. Automatic analysis of languages with rich morphology needs to pay attention to the interaction between morphology and syntax in order to arrive at suitable computational models. Linguistic theory (e. g., Bresnan (2001), Melčuk (2009)) suggests many interactions between morphology and syntax. For example, languages with a case system use different forms of the same word to mark different syntactic (or semantic) relations (Blake, 2001). In many languages, two words that participate in a syntactic relation show covariance in some or all of their morphological features (so-called agreement, Corbett (2006)).[3]

Automatic annotation of morphology assigns morphological descriptions (e. g., *nominative-singular-masculine*) to word forms. It is usually modeled as a sequence model, often in combination with part-of-speech tagging and lemmatization (Collins, 2002; Hajič, 2004; Smith et al., 2005; Chrupała et al., 2008, and others). Sequence models achieve high accuracy and coverage but since they only use linear context they only approximate some of the underlying hierarchical relationships. As an example for these hierarchical relationships,

---

[1]Syncretism describes the situation where a word form is ambiguous between several different morphological descriptions within its inflection paradigm.

[2]And also by prosodic means, which we will not discuss since text-based tools rarely have access to this information.

[3]For example, in English, the subject of a sentence and the finite verb agree with respect to their number and person feature.
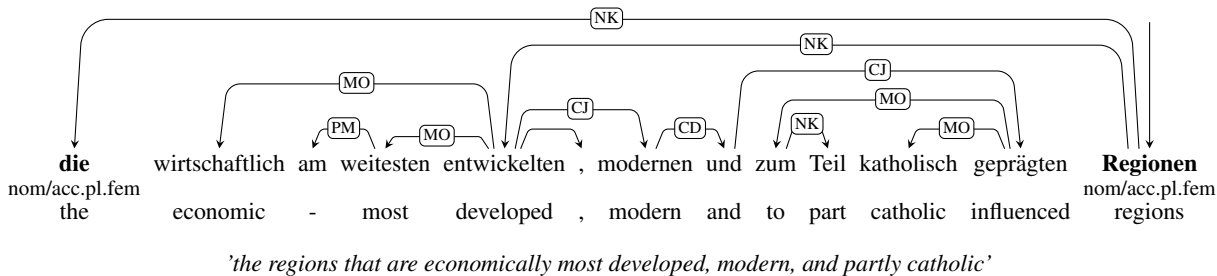
Figure 1: Example of a German noun phrase. First and last word agree in number, gender, and case value.

Figure 1 shows a German noun phrase taken from the German TiGer corpus (Brants et al., 2002). The two bold-faced words are the determiner and the head noun of the phrase, and they agree in their gender, number, and case values. The word *Regionen* (regions) is four-way ambiguous for its case value, which is reduced to a two-way ambiguity between *nominative* and *accusative* by the determiner. Further disambiguation would require information about the syntactic role of the noun phrase in a sentence. There are 11 tokens between these two words, which would require a context window of at least 13 to capture the agreement relation within a sequence model. Syntactically, however, as indicated by the dependency tree, the determiner and the head are linked directly. The interdependency between morphology and syntax in the example thus manifests itself in the morphological disambiguation of a highly syncretic word form because of its government or agreement relation to its respective syntactic head/dependents.

Of course, the sequence model is most of the time a reasonable approximation, because the majority of noun phrases in the TiGer corpus are not as long as the example in Figure 1.[4] Furthermore, not all languages show this kind of relationship between morphological forms and syntactic relation as demonstrated for German. But taking advantage of the morphosyntactic dependencies in a language can give us better models that may even be capable of handling the more difficult or rare cases. We therefore advocate that models for predicting morphology should be designed with the typological characteristics of a language and its morphosyntactic properties in mind, and should, where appropriate, integrate

---

[4] We find 57,551 noun phrases with less than three tokens between determiner and noun and 4,670 with three or more.

syntactic information in order to better model the morphosyntactic interdependencies of the language.

In the remainder of the paper, we show empirically that taking syntactic information into account produces state-of-the-art models for languages with a high interdependency between morphology and syntax. We use a simple setup, where we combine a morphological tagger and a dependency parser in a bootstrapping architecture in order to analyze the effect of syntactic information on the performance of the morphological tagger (Section 2). Using syntactic features in morphology prediction requires a syntactically annotated corpus for training a statistical parser, which may not be available for languages with few resources. We show in Section 3 that only very little syntactically annotated data is required to achieve the improvements. We furthermore expect that the improved morphological information also improves parsing performance and present a preliminary experiment in Section 4.

2 Experiments

In this section, we present a series of experiments that investigate the effect of syntactic information on the prediction of morphological features. We start by describing our data sets and the system that we used for the experiments.

2.1 Languages and Data Sets

We test our hypotheses on four different languages: Czech, German, Hungarian, and Spanish.

Spanish, a Romance language, and German, a Germanic language, constitute inflecting languages that show verbal and nominal morphology, but not as sophisticated as Czech and Hungarian. As we will see in the experiments, it is relatively easy to

334

predict the morphological information annotated in the Spanish data set.

Czech and Hungarian represent languages with very rich morphological systems both in verbal and nominal morphological paradigms. They differ significantly in the way in which morphological information is encoded in word forms. Czech, a Slavic language, is an inflecting language, where one suffix may signal several different morphological categories simultaneously (e. g., number, gender, case). In contrast, Hungarian, a Finno-Ugric language, is of the agglutinating type, where each morphological category is marked by its own morpheme.

Both German and Czech show various form syncretisms in their inflection paradigms. Form syncretisms emerge when the same word form is ambiguous between several different morphological descriptions, and they are a major challenge to automatic morphological analysis. Spanish shows syncretism in the verbal inflection paradigms. In Hungarian, form syncretisms are much less frequent. The case paradigm of Hungarian only shows one form syncretism between dative and genitive case (out of about 18 case suffixes).

All languages show agreement between subject and verb, and within the noun phrase. The word order in Czech and Hungarian is very variable whereas it is more restrictive in Spanish and German.

As our data, we use the CoNLL 2009 Shared Task data sets (Hajič et al., 2009) for Czech and Spanish. For German, we use the dependency conversion of the TiGer treebank by Seeker and Kuhn (2012), splitting it into 40k/5k/5k sentences for training/development/test. For Hungarian, we use the Szeged Dependency Treebank (Vincze et al., 2010), with the split of Farkas et al. (2012).

2.2 System Description

To test our hypotheses, we implemented a tagger that assigns full morphological descriptions to each token in a sentence. The system was inspired by the morphological tagger included in mate-tools.[5] Like the tagger provided with mate-tools, it is a classifier that tags each token using the surrounding tokens in

---

[5] A collection of language independent, data-driven analysis tools for lemmatization, pos-tagging, morphological analysis, and dependency parsing: http://code.google.com/p/mate-tools

its feature model. Models are trained using passive-aggressive online training (Crammer et al., 2003). The system makes two passes over each sentence: The first pass provides predicted tags that are used as features during the second pass. We also adopted the idea of a tag filter, which deterministically assigns tags for words that always occur with the same tag in the training data.

For all matters of syntactic annotation in this paper, we use the graph-based dependency parser by Bohnet (2010), also included in mate-tools. All data sets are annotated with gold syntactic information, which is used to train the parsing models.

For our experiments, we use a bootstrapping approach: the parser uses the output of the morphology in its feature set, and the morphological tagger we want to analyze uses the output of the parser as syntactic features. Since it is best to keep the training setting as similar as possible to the test setting, we use 10-fold jackknifing to annotate our training data with predicted morphology or syntax respectively.

Jackknifing differs from cross-validation only in its purpose. Cross-validation is used for evaluating data, jackknifing is used to annotate data. The data set is split into n parts, and n-1 parts are used to train a model for annotating the nth part. This is then rotated n times such that each part is annotated by the automatic tool without training it on its own test data. Jackknifing is important for creating a realistic training scenario that provides automatic preprocessing. For annotating development and test sets, models are trained on the jackknifed training set.

2.3 The Effects of Syntactic Features

In the first experiment, we use the system described in Section 2.2 to predict morphological information on all four languages. We start with describing the general setup and the feature set, and continue with a discussion of the results.

The experimental setup is as follows: the German and Spanish data sets are annotated with lemma and part-of-speech information using 10-fold jackknifing. The annotation is done with mate-tools' lemmatizer and pos-tagger. For Czech and Hungarian, we keep the annotation provided with the data sets.

Note that our experimental setup does not include lemmas or part-of-speech tags as part of the prediction of the morphology but annotates them in a pre-

processing step. It is not necessary to separate part-of-speech and lemma from the prediction of morphology and, in fact, many systems perform these steps simultaneously (e. g. Spoustová et al. (2009)). Doing morphology prediction as a separate step allows us to use lemma and part-of-speech information in the feature set.[6]

static features		
form	form1b	form2b
form3b	form1a	lemma2a
pos1b	pos2b	pos1a
form+pos	pos+s1	pos+s2
pos+s3	pos+s4	lemma+p2
lemma+p3	pos+number	form+form1b
pos+pos1a	pos+pos1b+pos2b	s1+s1_1b
s1+s1_1a	s2+s2_1a	last-verb-lemma
last-verb-pos	next-verb-lemma	next-verb-pos
dynamic features		
tag1b+tag2b	tag2b+tag3b	tag1a
tag1a+tag1b	tag1a+tag2a	tag2a+tag3a
pos1b+case1b	last-verb-tag	next-verb-tag
pos1b+case1b+pos2b+case2b		
Hungarian only features		
pos+uppercase		
Czech only features		
pos+p2		
Spanish only features		
s5	p1	p4
p5	s2_1a	s3_1a
s4_1a		

Table 1: Baseline feature set. *form* means word form, *lemma* is lemma, *pos* is part-of-speech, *s1/p1* stand for suffix and prefix of length 1 (characters), *tag* is the morphological tag predicted by the system, *1b/1a* means 1 token before/after the current token, and + marks feature conjunctions. *number* marks if the form contains a digit.

After preprocessing the data, our baseline system is trained using the feature set shown in Table 1. The baseline system does not make use of any syntactic information but predicts morphological information based solely on tokens and their linear context. The features are divided into static features, which can be computed on the input, and dynamic features, which are computed also on previous output of the system (cf. two passes in Section 2.2).

---

[6]Lemma and part-of-speech prediction may also profit from syntactic information, see e.g. Prins (2004) or Bohnet and Nivre (2012).

The feature sets in Table 1 were developed specifically for our experiments and are the result of an automatic forward/backward feature selection process. The purpose of the feature selection was to arrive at a baseline system that performs well without any syntactic information. With such an optimized baseline system, we can measure the contribution of syntactic features more reliably.

The *last-verb/next-verb* and *pos+case* features are variants of the features proposed in Votrubec (2006). They extract information about the first verb within the last 10/the next 30 tokens in the sentence. The *case* feature extracts the case value from previously assigned morphological tags. Note that the *verb* features are approximating syntactic information by making the assumption that the closest verbs are likely to be syntactic heads for many words.

static features				
h_lemma	h_s2	h_s3	pos+h_pos	s1+h_s1
h_dir	h_dir+h_pos			
ld_s1	ld_s2	ld_p1	ld_p4	
dynamic features				
h_tag	ld_tag			

Table 2: Syntactic features. *h_* and *ld_* mark features from the head and the left-most daughter, *dir* is a binary feature marking the direction of the head with respect to the current token.

After training the baseline models, we use them to annotate the whole data set with morphological information (using 10-fold jackknifing for the training portions). We then use 10-fold jackknifing again to annotate the data sets with the dependency parser.

At this point, all our data sets are annotated with predicted morphology from our baseline system and with syntactic information from the parser, which uses the morphological information from our baseline system in its feature set. We can now retrain our morphological tagger using features that are derived from the dependency trees provided by the parser. Note that this is not a stacking architecture, since the second system does not use the predicted morphology output from the baseline system. The loop simply ensures that we get the best possible syntactic features.

We extract two kinds of syntactic features: features of the syntactic head of the current token, and

	dev set		test set	
	all	oov	all	oov
Czech				
morfette	90.37	68.66	90.01	67.25
our baseline	92.51	73.12	92.29	72.58
pred syntax	*93.18	74.04	*92.82	73.11
gold syntax	*93.64	75.20	*93.30	74.96
German				
morfette	86.78	66.37	84.58	61.05
our baseline	90.92	72.52	89.11	69.67
pred syntax	*92.07	75.06	*90.10	71.18
gold syntax	*92.70	*76.29	*90.87	*73.20
Hungarian				
morfette	*96.19	*85.82	95.99	*85.43
our baseline	96.08	84.49	95.94	83.76
pred syntax	96.18	84.70	96.11	83.85
gold syntax	*96.46	85.30	*96.35	84.50
Spanish				
morfette	97.83	89.67	97.76	91.00
our baseline	97.83	89.05	97.59	90.88
pred syntax	97.84	89.08	97.67	90.91
gold syntax	98.11	90.34	97.88	91.61

Table 3: The effect of syntactic features when predicting morphological information. * mark statistically significantly better models compared to our baseline (sentence-based t-test with $\alpha = 0.05$).

	dev set		test set	
	all	oov	all	oov
Czech				
featurama	94.75	84.12	94.78	84.23
our baseline	93.80	80.47	93.57	80.53
pred syntax	*94.40	81.51	*94.24	81.61
gold syntax	*94.80	82.45	*94.64	82.80
German				
RFTagger	90.63	72.11	89.04	70.80
our baseline	92.59	80.73	91.48	78.83
pred syntax	*93.70	82.71	*92.51	80.20
gold syntax	*94.28	*84.12	*93.32	*82.35
Hungarian				
our baseline	97.27	92.61	97.03	91.28
pred syntax	97.38	92.39	97.19	91.50
gold syntax	*97.63	92.79	*97.45	91.92
Spanish				
our baseline	98.23	92.46	98.02	93.15
pred syntax	98.24	92.30	98.07	93.03
gold syntax	98.40	92.82	*98.22	93.64

Table 4: The effect of syntactic features when predicting morphology **using lexicons**. * mark statistically significantly better models compared to our baseline (sentence-based t-test with $\alpha = 0.05$).

features of the left-most daughter of the current token. We also experimented with other types, e. g. the right-most daughter, but these features did not improve the model. This is likely due to the way these languages encode morphological information and may be different for other languages. From the head and the left-most daughter, we construct features about form, lemma, affixes, and tags. Table 2 lists the syntactic features that we use in the model.

With the syntactic features available due to the parsing step, we train new models with the full system. For each language, we run four experiments. The first two are baseline experiments, where we use the off-the-shelf morphological tagger morfette (Chrupała et al., 2008) and our own baseline system, both of which do not use any syntactic features. In the third experiment, we evaluate our full system using the syntactic features provided by the dependency parser. As an oracle experiment, we also report results on the full system when using the gold standard syntax from the treebank. Table 3 presents all results in terms of accuracy on all tokens (all)

and out-of-vocabulary tokens only (oov). Out-of-vocabulary tokens do not occur in the training data.

We find trends along several axes: Generally, the syntactic features work well for Czech and German, whereas for Hungarian and Spanish, they do not yield any significant improvement. The improvements for German and Czech are between 0.5 (Czech) and 1.0 (German) percentage points absolute in token accuracy, and between 0.2 (Czech test set) and 2.5 (German dev set) percentage points absolute in accuracy of unknown words. There are no obvious differences between the development and the test set in any of the languages.

Compared to the morfette baseline, we find our systems to be either superior or equal to morfette in terms of token accuracy. Regarding accuracy on unknown words, morfette outperforms our systems for Hungarian, but is outperformed on Czech and German. For Spanish, all systems yield similar results.

Looking at the oracle experiment, we see that for all languages, the system can learn something from syntax. For Czech and German, this is clearly the

case, for Hungarian and Spanish, the differences are small but visible. There are pronounced differences between the predicted and the gold syntax experiments in Czech and German. Clearly, the parser makes mistakes that propagate through to the prediction of the morphology.

2.4 Syntax vs Lexicon

The current state-of-the-art in predicting morphological features makes use of morphological lexicons (e.g. Hajič (2000), Hakkani-Tür et al. (2002), Hajič (2004)). Lexicons define the possible morphological descriptions of a word and a statistical model selects the most probable one among them. In the following experiment, we test whether the contribution of syntactic features is similar or different to the contribution of morphological lexicons.

Lexicons encode important knowledge that is difficult to pick up in a purely statistical system, e. g. the gender of nouns, which often cannot be deduced from the word form (Corbett, 1991).[7]

We extend our system from the previous experiment to include information from a morphological dictionaries. For Czech, we use the morphological analyzer distributed with the Prague Dependency Treebank 2 (Hajič et al., 2006). For German, we use DMor (Schiller, 1994). For Hungarian, we use (Trón et al., 2006), and for Spanish, we use the morphological analyzer included in Freeling (Carreras et al., 2004). The output of the analyzers is given to the system as features that simply record the presence of a particular morphological analysis for the current word. The system can thus use the output of any tool regardless of its annotation scheme, especially if the annotation scheme of the treebank is different from the one of the morphological analyzer.

Table 4 presents the results of experiments where we add the output of the morphological analyzers to our system. Again, we run experiments with and without syntactic features. For Czech, we also show results from featurama[8] with the feature set developed by Votrubec (2006). For German, we show results for RFTagger (Schmid and Laws, 2008).

As expected, the information from the morphological lexicon improves the overall performance

---

[7]Lexicons are also often used to speed up processing considerably by restricting the search space of the statistical model.

[8]http://sourceforge.net/projects/featurama/

considerably compared to the results in Table 3, especially on unknown tokens. This shows that even with the considerable amounts of training data available nowadays, rule-based morphological analyzers are important resources for morphological description (cf. Hajič (2000)). The contribution of syntactic features in German and Czech is almost the same as in the previous experiment, indicating that the syntactic features contribute information that is orthogonal to that of the morphological lexicon. The lexicon provides lexical knowledge about a word form, while the syntactic features provide the syntactic context that is needed in German and Czech to decide on the right morphological tag.

2.5 Language Differences

From the previous experiments, we conclude that syntactic features help in the prediction of morphology for Czech and German, but not for Hungarian and Spanish. To further investigate the difference between Czech and German on the one hand, and Hungarian and Spanish on the other, we take a closer look at the output of the tagger.

We find an interesting difference between the two pairs of languages, namely the performance with respect to agreement. Agreement is a phenomenon where morphology and syntax strongly interact. Morphological features co-vary between two items in the sentence, but the relation between these items can occur at various linguistic levels (Corbett, 2006). If the syntactic information helps with predicting morphological information, we expect this to be particularly helpful with getting agreement right. All languages show agreement to some extent. Specifically, all languages show agreement in number (and person) between the subject and the verb of a clause. Czech, German, and Spanish show agreement in number, gender, and case (not Spanish) within a noun phrase. Hungarian shows case agreement within the noun phrase only rarely, e.g. for attributively used demonstrative pronouns.

In order to test the effect on agreement, we measure the accuracy on tokens that are in an agreement relation with their syntactic head. We counted subject verb agreement as well as agreement with respect to number, gender, and case (where applicable) between a noun and its dependent adjective and determiner. Table 5 displays the counts from the devel-

opment sets of each language. We compare the baseline system that does not use any syntactic information with the output of the morphological tagger that uses the gold syntax. We use the gold syntax rather than the predicted one in order to eliminate any influence from parsing errors. As can be seen from the results, the level of agreement relations in Czech and German improves when using syntactic information, whereas in Spanish and Hungarian, only very tiny changes occur.

agreement	baseline	gold syntax
Czech		
sbj-verb	3199/4044 = 79.10	3264/4044 = 80.71
NP case	8719/9132 = 95.48	8821/9132 = 96.59
NP num	8933/9132 = 97.82	9016/9132 = 98.73
NP gen	8493/9132 = 93.00	8768/9132 = 96.01
German		
sbj-verb	4412/4696 = 93.95	4562/4696 = 97.15
NP case	13340/13951 = 95.62	13510/13951 = 96.84
NP num	13631/13951 = 97.71	13788/13951 = 98.83
NP gen	13253/13951 = 95.00	13528/13951 = 96.97
Hungarian		
sbj-verb	8653/10219 = 84.68	8655/10219 = 84.70
NP case	402/891 = 45.12	412/891 = 46.24
Spanish		
sbj-verb	1930/2004 = 96.31	1932/2004 = 96.41
NP num	8810/8849 = 99.56	8816/8849 = 99.63
NP gen	8810/8849 = 99.56	8821/8849 = 99.68

Table 5: Agreement counts in morphological annotation compared between the baseline system and the oracle system using gold syntax.

For Czech and German, these results sugguest that syntactic information helps with agreement. We believe that the reasons why it does not help for Hungarian and Spanish are the following: for Spanish, we see that also the baseline model achieves very high accuracies (cf. Table 3) and also high rates of correct agreement. It seems that for Spanish, syntactic context is simply not necessary to make the correct prediction. For Hungarian, the reason lies within the inflectional paradigms of the language, which do not show any form syncretism, meaning that word forms in Hungarian are usually not ambiguous within one morphological category (e.g. case). Making a morphological tag prediction, however, is difficult only if the word form itself is ambiguous between several morphological tags. In this case, using the agreement relation between the word and its syntactic head can help the system making

the proper prediction. This is the situation that we find in Czech and German, where form syncretism is pervasive in the inflectional paradigms.

2.6 Syntactic Features in Czech

In Section 2.4 we compared the performance of our system on Czech to another system, featurama (see Table 4). Featurama outperforms our baseline system by a percentage point in token accuracy (and even more for unknown tokens). Syntactic information closes that gap to a large extent but only using gold syntax gets our system on a par with featurama.

The question then arises whether the syntactic features actually contribute something new to the task, or whether the same effect could also be achieved with linear context features alone as in featurama. In order to test this we run an additional experiment, where we add some of the syntax features to the feature set of featurama. Specifically, we add the static features from Table 2 that do not use lemma or part-of-speech information. Due to the way featurama works, we cannot use features from the morphological tags (the dynamic features).

The results in Table 6 show that also featurama profits from syntactic features, which corroborates the findings from the previous experiments. We also note again that better syntax would improve results even more.

	dev set		test set	
	all	oov	all	oov
featurama	94.75	84.12	94.78	84.23
pred syntax	95.18	84.65	95.09	84.52
gold syntax	*95.39	84.62	*95.34	85.03

Table 6: Syntactic features for featurama (Czech). * mark statistically significantly better models compared to featurama (sentence-based t-test with $\alpha = 0.05$).

3 How Much Syntax is Needed?

Syntactic features require syntactically annotated corpora. Without a treebank to train the parser, the morphology cannot profit from syntactic features.[9] This may be problematic for languages for which there is no treebank, because creating a treebank is expensive. Fortunately, it turns out that very small amounts of syntactically annotated data are enough

---

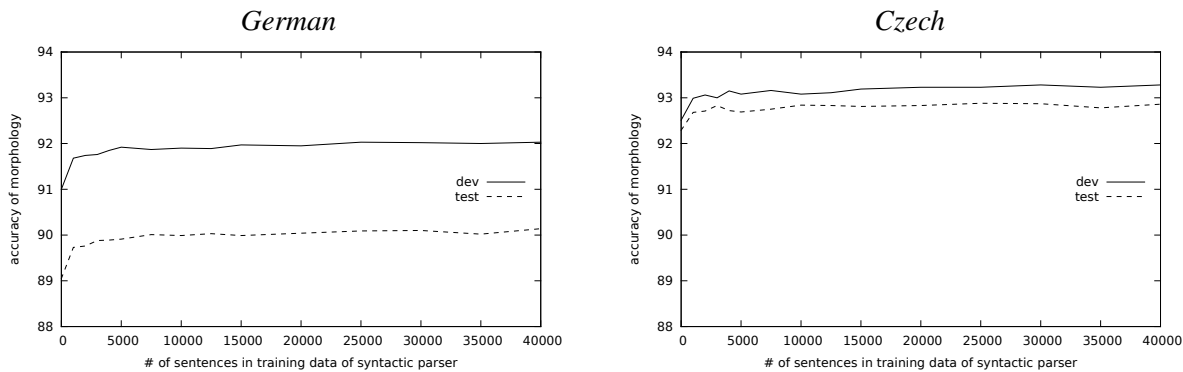[9]Which is of course only a problem for statistical parsers.

Figure 2: Dependency between amount of training data for syntactic parser and quality of morphological prediction.

to provide a parsing quality that is sufficient for the morphological tagger.

In order to test what amount of training data is needed, we train several parsing models on increasing amounts of syntactically annotated data. For example, the first experiment uses the first 1,000 sentences of the treebank. We perform 5-fold jackknifing with the parser on these sentences to annotate them with syntax. Then we train one parsing model on these 1,000 sentences and use it to annotate the rest of the training data as well as the development and the test set. This gives us the full data set annotated with syntax that was learned from the first 1,000 sentences of the treebank. The morphological tagger is then trained on the full training set and applied to development and test set.

Figure 2 shows the dependency between the amount of training data given to the parser and the quality of the morphological tagger using syntactic features provided by this parser. The left-most point corresponds to a model that does not use syntactic information. For both languages, German and Czech, we find that already 1,000 sentences are enough training data for the parser to provide useful syntactic information to the morphological tagger. After 5,000 sentences, both curves flatten out and stay on the same level. We conclude that using syntactic features for morphological prediction is viable even if there is only small amounts of syntactic data available to train the parser.

As a related experiment, we also test if we can get the same effect with a very simple and thus much faster parser. We use the brute-force algorithm described in Covington (2001), which selects for each

token in the sentence another token as the head. It does not have any tree requirements, so it is not even guaranteed to yield a cycle-free tree structure. In Table 7, we compare the simple parser with the mate-parser, both trained on the first 5,000 sentences of the treebank. Evaluation is done in terms of labeled (LAS) and unlabeled attachment score (UAS).[10]

	dev set		test set	
	LAS	UAS	LAS	UAS
Czech				
simple parser (5k)	71.57	78.96	69.09	77.23
full parser (5k)	76.77	84.38	74.70	83.00
German				
simple parser (5k)	83.06	85.23	78.56	81.18
full parser (5k)	87.56	90.08	83.69	86.58

Table 7: Simple parser vs full parser – syntactic quality. Trained on first 5,000 sentences of the training set.

As expected, the simple parser performs much worse in terms of syntactic quality. Table 8 shows the performance of the morphological tagger when using the output of both parsers as syntactic features. For Czech, both parsers seem to supply similar information to the morphological tagger, while for German, using the full parser is clearly better. In both cases, the morphological tagger outperforms the models that do not use syntactic information (cf. Table 3). The performance on unknown words is however much worse for both languages. We conclude that even with a simple parser and little training data, the morphology can make use of syntactic information to some extent.

---

[10]LAS: $\frac{\text{correct edges with correct labels}}{\text{all edges}}$, UAS: $\frac{\text{correct edges}}{\text{all edges}}$

	dev set		test set	
	all	oov	all	oov
Czech				
no syntax	92.51	73.12	92.29	72.58
simple syntax	92.96	73.45	92.53	72.66
full syntax	93.08	73.64	92.69	73.39
German				
no syntax	90.92	72.52	89.11	69.67
simple syntax	91.52	73.34	89.66	70.52
full syntax	91.92	83.46	89.91	80.50

Table 8: Simple parser vs full parser – morphological quality. The parsing models were trained on the first 5,000 sentences of the training data, the morphological tagger was trained on the full training set.

4 Does Better Morphology lead to Better Parses?

In the previous sections, we show that syntactic information improves a model for predicting morphology for Czech and German, where syntax and morphology interact considerably. A natural question then is whether the improvement also occurs in the other direction, namely whether the improved morphology also leads to better parsing models.

In the previous experiments, we run a 10-fold jackknifing process to annotate the training data with morphological information using no syntactic features and afterwards use jackknifing with the parser to annotate syntax. The syntax is subsequently used as features for our predicted-syntax experiments. We can apply the same process once more with the morphology prediction in order to annotate the training data with morphological information that is predicted using the syntactic features. A parser trained on this data will then use the improved morphology as features. If the improved morphology has an impact on the parser, the quality of the second parsing model should then be superior to the first parsing model, which uses the morphology predicted without syntactic information. Note that for the following experiments, neither morphology model uses the morphological lexicon.

Table 9 presents the evaluation of the two parsing models (one using morphology without syntactic features, the other one using the improved morphology). The results show no improvement in parsing performance when using the improved morphology. Looking closer at the output, we find differences be-

	dev set		test set	
	LAS	UAS	LAS	UAS
Czech				
baseline morph	81.73	88.45	81.02	87.77
morph w/ syntax	81.63	88.37	80.83	87.61
German				
baseline morph	91.16	92.97	88.06	90.24
morph w/ syntax	91.20	92.97	88.15	90.34

Table 9: Impact of the improved morphology on the quality of the dependency parser for Czech and German.

tween the two parsing models with respect to grammatical functions that are morphologically marked. For example, in German, performance on subjects and accusative objects improves while performance for dative objects and genitives decreases. This suggests different strengths in the two parsing models. However, the question how to make use of the improved morphology in parsing clearly needs more research in the future. A promising avenue may be the approach by Hohensee and Bender (2012).

5 Related Work

Morphological taggers have been developed for many languages. The most common approach is the combination of a morphological lexicon with a statistical disambiguation model (Hakkani-Tür et al., 2002; Hajič, 2004; Smith et al., 2005; Spoustová et al., 2009; Zsibrita et al., 2013).

Our work has been inspired by Versley et al. (2010), who annotate a treebank with morphological information after the syntax had been annotated already. The system used a finite-state morphology to propose a set of candidate tags for each word, which is then further restricted using hand-crafted rules over the already available syntax tree.

Lee et al. (2011) pursue the idea of jointly predicting syntax and morphology, out of the motivation that joint models should model the problem more faithfully. They demonstrate that both sides can use information from each other. However, their model is computationally quite demanding and its overall performance falls far behind the standard pipeline approach where both tasks are done in sequence.

The problem of modeling the interaction between morphology and syntax has recently attracted some attention in the SPMRL workshops (Tsarfaty et al.,

341

2010). Modeling morphosyntactic relations explicitly has been shown to improve statistical parsing models (Tsarfaty and Sima'an, 2010; Goldberg and Elhadad, 2010; Seeker and Kuhn, 2013), but the co-dependency between morphology and syntax makes it a difficult problem, and linguistic intuition is often contradicted by the empirical findings. For example, Marton et al. (2013) show that case information is the most helpful morphological feature for parsing Arabic, but only if it is given as gold information, whereas using case information from an automatic system may even harm the performance.

Morphologically rich languages pose different challenges for automatic systems. In this paper, we work with European languages, where the problem of predicting morphology can be reduced to a tagging problem. In languages like Arabic, Hebrew, or Turkish, widespread ambiguity in segmentation of single words into meaningful morphemes adds an additional complexity. Given a good segmentation tool that takes care of this, our approach is applicable to these languages as well. For Hebrew, this problem has also been addressed by jointly modeling segmentation, morphological prediction, and syntax (Cohen and Smith, 2007; Goldberg and Tsarfaty, 2008; Goldberg and Elhadad, 2013).

6   Conclusion

In this paper, we have demonstrated that using syntactic information for predicting morphological information is helpful if the language shows form syncretism in combination with morphosyntactic phenomena like agreement. A model that uses syntactic information is superior to a sequence model because it leverages the syntactic dependencies that may hold between morphologically dependent words as suggested by linguistic theory. We also showed that only small amounts of training data for a statistical parser would be needed to improve the morphological tagger. Making use of the improved morphology in the dependency parser is not straight-forward and requires more investigation in the future.

Modeling the interaction between morphology and syntax is important for building successful parsing pipelines for languages with free word order and rich morphology. Moreover, our experiments show that paying attention to the individual properties of a language can help us explain and predict the behavior of automatic tools. Thus, the term "morphologically rich language" should be viewed as a broad term that covers many different languages, whose differences among each other may be as important as the difference with languages with a less rich morphology.

References

Barry J. Blake. 2001. *Case*. Cambridge University Press, Cambridge, New York, 2nd edition.

Bernd Bohnet and Joakim Nivre. 2012. A Transition-Based System for Joint Part-of-Speech Tagging and Labeled Non-Projective Dependency Parsing. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 1455–1465, Jeju, South Korea. Association for Computational Linguistics.

Bernd Bohnet. 2010. Very high accuracy and fast dependency parsing is not a contradiction. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 89–97, Beijing, China. International Committee on Computational Linguistics.

Sabine Brants, Stefanie Dipper, Silvia Hansen-Shirra, Wolfgang Lezius, and George Smith. 2002. The TIGER treebank. In *Proceedings of the 1st Workshop on Treebanks and Linguistic Theories*, pages 24–41, Sozopol, Bulgaria.

Joan Bresnan. 2001. *Lexical-Functional Syntax*. Blackwell Publishers.

Xavier Carreras, Isaac Chao, Llus Padr, and Muntsa Padr. 2004. Freeling: An open-source suite of language analyzers. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC'04)*, pages 239–242. European Language Resources Association (ELRA).

Grzegorz Chrupała, Georgiana Dinu, and Josef van Genabith. 2008. Learning morphology with morfette. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, pages 2362–2367, Marrakech, Morocco. European Language Resources Association (ELRA). http://www.lrec-conf.org/proceedings/lrec2008/.

Shay B. Cohen and Noah A. Smith. 2007. Joint morphological and syntactic disambiguation. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 208–217, Prague, Czech Republic. Association for Computational Linguistics.

Michael Collins. 2002. Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing*, pages 1–8. Association for Computational Linguistics, July.

Greville G. Corbett. 1991. *Gender*. Cambridge Textbooks in Linguistics. Cambridge University Press.

Greville G. Corbett. 2006. *Agreement*. Cambridge Textbooks in Linguistics. Cambridge University Press.

Michael A. Covington. 2001. A fundamental algorithm for dependency parsing (with corrections). In *Proceedings of the 39th Annual ACM Southeast Conference*, Athens, Gorgia. ACM.

Koby Crammer, Ofer Dekel, Shai Shalev-Shwartz, and Yoram Singer. 2003. Online passive-aggressive algorithms. In *Proceedings of the 16th Annual Conference on Neural Information Processing Systems*, volume 7, pages 1217–1224, Cambridge, Massachusetts, USA. MIT Press.

Richárd Farkas, Veronika Vincze, and Helmut Schmid. 2012. Dependency parsing of hungarian: Baseline results and challenges. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 55–65, Avignon, France. Association for Computational Linguistics.

Yoav Goldberg and Michael Elhadad. 2010. Easy first dependency parsing of modern Hebrew. In *Proceedings of the NAACL HLT 2010 First Workshop on Statistical Parsing of Morphologically-Rich Languages*, pages 103–107, Los Angeles, California, USA. Association for Computational Linguistics.

Yoav Goldberg and Michael Elhadad. 2013. Word segmentation, unknown-word resolution, and morphological agreement in a hebrew parsing system. *Computational Linguistics*, 39(1):121–160.

Yoav Goldberg and Reut Tsarfaty. 2008. A single generative model for joint morphological segmentation and syntactic parsing. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics*, pages 371–379, Columbus, Ohio. Association for Computational Linguistics.

Jan Hajič. 2000. Morphological Tagging: Data vs. Dictionaries. In *Proceedings of the 6th ANLP Conference / 1st NAACL Meeting*, pages 94—101, Seattle, Washington. Association for Computational Linguistics.

Jan Hajič. 2004. *Disambiguation of Rich Inflection (Computational Morphology of Czech)*. Nakladatelství Karolinum, Prague, Czech Republic.

Jan Hajič, Jarmila Panevová, Eva Hajičová, Petr Sgall, Petr Pajas, Jan Štěpánek, Jií Havelka, and Marie Mikulová. 2006. Prague Dependency Treebank 2.0.

Jan Hajič, Massimiliano Ciaramita, Richard Johansson, Daisuke Kawahara, Maria Antònia Martí, Lluís Màrquez, Adam Meyers, Joakim Nivre, Sebastian Padó, Jan Stepának, Pavel Stranák, Mihai Surdeanu, Nianwen Xue, and Yi Zhang. 2009. The CoNLL-2009 shared task: Syntactic and Semantic dependencies in multiple languages. In *Proceedings of the 13th Conference on Computational Natural Language Learning: Shared Task*, pages 1–18, Boulder, Colorado, USA. Association for Computational Linguistics.

Dilek Z. Hakkani-Tür, Kemal Oflazer, and Gökhan Tür. 2002. Statistical morphological disambiguation for agglutinative languages. *Computers and the Humanities*, 36(4):381–410.

Matt Hohensee and Emily M. Bender. 2012. Getting more from morphology in multilingual dependency parsing. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 315–326, Montréal, Canada. Association for Computational Linguistics.

John Lee, Jason Naradowsky, and David A. Smith. 2011. A discriminative model for joint morphological disambiguation and dependency parsing. In *Proceedings of the 49th annual meeting of the Association for Computational Linguistics*, pages 885–894, Portland, USA. Association for Computational Linguistics.

Yuval Marton, Nizar Habash, and Owen Rambow. 2013. Dependency parsing of modern standard arabic with lexical and inflectional features. *Computational Linguistics*, 39(1):161–194.

Igor Melčuk. 2009. *Dependency in linguistic description*.

Robbert Prins. 2004. Beyond N in N-gram tagging. In Leonoor Van Der Beek, Dmitriy Genzel, and Daniel Midgley, editors, *Proceedings of the ACL 2004 Student Research Workshop*, pages 61–66, Barcelona, Spain. Association for Computational Linguistics.

Anne Schiller. 1994. Dmor - user's guide. Technical report, University of Stuttgart.

Helmut Schmid and Florian Laws. 2008. Estimation of conditional probabilities with decision trees and an application to fine-grained POS tagging. In *Proceedings of the 22nd International Conference on Computational Linguistics*, pages 777–784, Morristown, NJ, USA. Association for Computational Linguistics.

Wolfgang Seeker and Jonas Kuhn. 2012. Making Ellipses Explicit in Dependency Conversion for a German Treebank. In *Proceedings of the 8th International Conference on Language Resources and Evaluation*, pages 3132–3139, Istanbul, Turkey. European Language Resources Association (ELRA).

Wolfgang Seeker and Jonas Kuhn. 2013. Morphological and syntactic case in statistical dependency parsing. *Computational Linguistics*, 39(1):23–55.

Noah A. Smith, David A. Smith, and Roy W. Tromble. 2005. Context-based morphological disambiguation with random fields. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 475–482, Vancouver, British Columbia, Canada, October. Association for Computational Linguistics.

Drahomíra "Johanka" Spoustová, Jan Hajič, Jan Raab, and Miroslav Spousta. 2009. Semi-supervised training for the averaged perceptron POS tagger. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pages 763–771, Athens, Greece. Association for Computational Linguistics.

Viktor Trón, Péter Halácsy, Péter Rebrus, András Rung, Péter Vajda, and Eszter Simon. 2006. Morphdb.hu: Hungarian lexical database and morphological grammar. In *Proceedings of the 5th International Conference on Language Resources and Evaluation*, pages 1670–1673, Genoa, Italy.

Reut Tsarfaty and Khalil Sima'an. 2010. Modeling morphosyntactic agreement in constituency-based parsing of Modern Hebrew. In *Proceedings of the NAACL HLT 2010 First Workshop on Statistical Parsing of Morphologically-Rich Languages*, pages 40–48, Los Angeles, California, USA. Association for Computational Linguistics.

Reut Tsarfaty, Djamé Seddah, Yoav Goldberg, Sandra Kübler, Marie Candito, Jennifer Foster, Yannick Versley, Ines Rehbein, and Lamia Tounsi. 2010. Statistical parsing of morphologically rich languages (SPMRL): what, how and whither. In *Proceedings of the NAACL HLT 2010 First Workshop on Statistical Parsing of Morphologically-Rich Languages*, pages 1–12, Los Angeles, California, USA. Association for Computational Linguistics.

Yannick Versley, Kathrin Beck, Erhard Hinrichs, and Heike Telljohann. 2010. A syntax-first approach to high-quality morphological analysis and lemma disambiguation for the tba-d/z treebank. In *9th Conference on Treebanks and Linguistic Theories (TLT9)*, pages 233–244.

Veronika Vincze, Dóra Szauter, Attila Almási, György Móra, Zoltán Alexin, and János Csirik. 2010. Hungarian Dependency Treebank. In *Proceedings of the 7th Conference on International Language Resources and Evaluation*, pages 1855–1862, Valletta, Malta. European Language Resources Association (ELRA).

Jan Votrubec. 2006. Morphological tagging based on averaged perceptron. In *WDS'06 Proceedings of Contributed Papers*, pages 191–195, Praha, Czechia. Matfyzpress, Charles University.

János Zsibrita, Veronika Vincze, and Richárd Farkas. 2013. magyarlanc 2.0: szintaktikai elemzés és felgyorsított szófaji egyértelmsítés. In Attila Tanács and Veronika Vincze, editors, *IX. Magyar Számítógépes Nyelvészeti Konferencia*, pages 368–374, Szeged, Hungary.