Robust Morphological Tagging with Word Representations

Thomas Müller and **Hinrich Schütze**
Center for Information and Language Processing
University of Munich, Germany
muellets@cis.lmu.de

Abstract

We present a comparative investigation of word representations for part-of-speech (POS) and morphological tagging, focusing on scenarios with considerable differences between training and test data where a *robust* approach is necessary. Instead of adapting the model towards a specific domain we aim to build a robust model across domains. We developed a test suite for robust tagging consisting of six languages and different domains. We find that representations similar to Brown clusters perform best for POS tagging and that word representations based on linguistic morphological analyzers perform best for morphological tagging.

1 Introduction

Most natural language processing (NLP) tasks can be better solved if a preprocessor tags each word in the natural language input with a label like "noun, singular" or "verb, past tense" that gives some indication of the syntactic role that the word plays in its context. The most common form of such preprocessing is POS tagging. However, for morphologically rich languages, a large subset of the languages of the world, POS tagging in its original form – where labels are syntactic categories with little or no morphological information – does not make much sense. The reason is that POS and morphological properties are mutually dependent, so solving only one task or solving the tasks sequentially is inadequate. The most important dependence of this type is that POS can be read off morphology in many cases; e.g., the morphological suffix "-iste" is a reliable indicator of the informal second person singular preterite indicative form of a verb in Spanish. In what follows, we use the term "morphological tagging" to refer to "morphological and POS tagging" since morphological tags generally include POS information.

The importance of morphological tagging as part of the computational linguistics processing pipeline motivated us to conduct the research reported in this paper. The specific setting that we address is increasingly recognized as the setting in which most practical NLP takes place: We look at scenarios with considerable differences between the training data and the application data, i.e., between the data that the tagger is trained on and the data that it is applied to. This type of scenario is frequent because of the great diversity and variability of natural language and because of the high cost of annotation – which makes it impossible to create large training sets for each new domain. For this reason, we address morphological tagging in a setting in which training and application data differ.

The most common approach to this setting is domain adaptation. Domain adaptation has been demonstrated to have good performance in scenarios with differently distributed training/test data. However, it has two disadvantages. First, it requires the availability of data from the target domain. Second, we need to do some extra work in domain adaptation – consisting of taking target domain data and using it to adapt our NLP system to the target domain – and we end up with a number of different versions of our NLP system. The extra work required and the pro-

Human Language Technologies: The 2015 Annual Conference of the North American Chapter of the ACL, pages 526–536,
Denver, Colorado, May 31 – June 5, 2015.

liferation of different versions increase the possibility of errors and increase the complexity of deploying NLP technology. Similar to other recent work (Zhang and Wang, 2009), we therefore take an approach that is different from domain adaptation. We build a system that is *robust across domains without any modification*. As a result, no extra work is required when the system is applied to a new domain: there is only one system and we can use it for all domains.

The key to making NLP components robust across domains is *the use of powerful domain-independent representations for words*. One of the main contributions of this paper is that we compare the performance of the most important representations that can be used for this purpose. We find that two of these are best suited for robust tagging. MarLiN (Martin et al., 1998) clusters – a derivative of Brown clusters – perform best for POS tagging. MarLiN clusters are also an order of magnitude more efficient to induce than the original Brown clusters. We provide an open source implementation of MarLiN clustering as part of this publication (Section 8). We compare the word representations to Morphological Analyzers (MAs), which are finite-state transducers that find the stems of a form and use them to derive all its possible morphological readings. MAs produce the best results in our experiments on morphological tagging. Our initial expectation was that domain differences and lack of coverage would put manually created MAs at a disadvantage when compared to learning algorithms that are run on very large text corpora. However, our results clearly show that MA-based representations are the best representations to use for robust morphological tagging.

The motivation for our work is that both morphological tagging and the "robust" application setting are important areas of research in NLP. To support this research, we created an extensive evaluation set for six languages. This involved identifying morphologically rich languages in which usable data sets with different distributional properties were available, designing mappings between different tag sets, organizing a manual annotation effort for one of the six languages and preparing large "general" (not domain-specific) data sets for unsupervised learning of word representations. The preparation and publication (Section 8) of this test suite is in itself a significant contribution.

The remainder of this paper is structured as follows. Section 2 discusses related work. Section 3 presents the representations we tested. Section 4 describes the data sets and the annotation and conversion efforts required to create the in-domain (ID) and out-of-domain (OOD) data sets. In Section 5, we describe the experiments and discuss our findings. In Section 6, we provide an analysis of our results. Section 7 summarizes our findings and contributions.

2 Related Work

Morphological tagging (Oflazer and Kuruöz, 1994; Hajič and Hladká, 1998) is the task of assigning a morphological reading to a token in context. The morphological reading consists of features such as case, gender, person and tense and is represented as a single tag. This allows for the application of standard sequence labeling algorithms such as Conditional Random Fields (CRFs) (Lafferty et al., 2001), but also puts an upper bound on the accuracy as only readings occurring in the training set can be produced. It is still the standard approach to morphological disambiguation as the number of readings that cannot be produced is usually small.

The related work can be divided in systems that try to exploit certain properties of a language (Habash and Rambow, 2005; Yuret and Türe, 2006) and language-independent systems (Hajič, 2000; Smith et al., 2005). In this paper, we adopt a language-independent approach.

Semi-supervised learning attempts to increase the accuracy of a machine learning system by using additional unlabeled data. Word representations, especially Brown clusters, have been extensively used for named entity recognition (NER) (Miller et al., 2004), parsing (Koo et al., 2008) and POS tagging (Collobert and Weston, 2008; Huang et al., 2009). In these papers, word representations were shown to yield consistent improvements and to often outperform traditional semi-supervised methods such as self-training. Prior work on semi-supervised training for morphological tagging includes Spoustová et al. (2009) and Chrupala (2011). In contrast to this earlier work on morphological tagging, we study a number of morphologically more complex and diverse languages. We also compare learned represen-

tations to representations obtained from MAs.

Domain adaptation (DA) attempts to adapt a model trained on a source domain to a target domain. DA can be broadly divided into supervised and unsupervised approaches depending on whether labeled target domain data is available or not. Among unsupervised approaches to DA, representation learning (Ando and Zhang, 2005; Blitzer et al., 2006) uses the unlabeled target domain data to induce a structure that is suitable for transferring information from the labeled source domain to the target domain. Similar to representation learning for DA, we attempt to include word representations into the model. However, we induce the representation from a *general domain* in an attempt to obtain a model that has robust high accuracy across domains, for the source domain as well as for the target domains, for which neither labeled nor unlabeled training data is available.

3 Representations

We survey the following distributional representations: (i) count vectors reduced by a Singular Value Decomposition (SVD), (ii) word clusters induced using the likelihood of a class-based language model, (iii) distributed embeddings trained using a neural network and (iv) accumulated tag counts, a task-specific representation obtained from an automatically tagged corpus.

Singular value decomposition of word-feature cooccurrence matrices (Schütze, 1995) has been found to be a fast and efficient way to obtain distributed embeddings. The approach selects a subset of the vocabulary as so-called feature words, usually by including words up to a certain frequency rank. Every word form can then be represented by the accumulated counts of feature words occurring to its left and right. Then an SVD is applied to the cooccurrence matrix as a form of dimension reduction and to reduce sparsity.

We also experimented with unreduced count vectors, but they did not give better results than SVD reduced count vectors. SVD-based representations have been used in English POS induction (Lamar et al., 2010) as well as as features in English POS tagging and syntactic chunking (Huang et al., 2009); they have a similar level of accuracy as unsupervised Hidden Markov Models (HMMs) in these studies.

Language model-based (LM-based) word clusters were introduced by Brown et al. (1992) and later found to be helpful in a range of NLP tasks. The basic idea is to find the optimal clustering with respect to the likelihood of a class-based language model:

$$g = \arg\max_{g} \prod_{i=1}^{|D|} p(g(x_i)|g(x_{i-1})) \cdot p(x_i|g(x_i))$$

where $g(x)$ is the cluster assignment function that maps a word form x to a cluster and $|D|$ denotes the length of the training set. Brown et al. (1992) propose a greedy bottom-up algorithm for the optimization that merges the pair of clusters that yields the smallest loss in likelihood; as well as a more efficient approximation of that algorithm that limits the number of clusters under consideration and still works well in practice. It is used by most work in the literature (Liang, 2005; Turian et al., 2010; Koo et al., 2008).

We, however, found the algorithm proposed by Martin et al. (1998) to be faster and to give slightly better results. The algorithm is similar to K-means in that it starts with an initial clustering and greedily improves the objective function by moving single words to their optimal cluster. In contrast to K-means, it updates the objective function immediately. The algorithm has also been shown to work well in unsupervised POS induction (Clark, 2003; Blunsom and Cohn, 2011). Our implementation of this algorithm is called MarLiN and has been made available as open-source software (Section 8). Miller et al. (2004) use tags of different granularity induced from unlabeled text to improve the performance of an averaged perceptron tagger (Collins, 2002) on an English NER task.

The Brown algorithm induces a tree where leaves represent a single word form and the root node the entire vocabulary. Intermediate nodes represent clusters of different sizes and can be addressed by a binary string specifying the path from the root node to the cluster. Brown clusters are also used by Koo et al. (2008) to improve dependency parsing for English and Czech. Chrupala (2011) compare Brown clusters to a Latent Dirichlet Allocation

(LDA) model on Spanish and French morphological tagging and find them to yield similar performance.[1]

Neural networks have been used by Collobert and Weston (2008) to train embeddings for POS tagging as well as other NLP tasks. These embeddings – henceforth *CW embeddings* – are trained by building a neural network that given contexts of a word as input is trained to discriminate between the correct center word and a random word. The proposed training algorithm is reported to need several days or even weeks, but has been reimplemented by Al-Rfou et al. (2013), who induced embeddings for the Wikipedias of more than 100 languages. Turian et al. (2010) find that the performance of Brown clusters is competitive with more training intensive embeddings like CW. In our experiments, we find that MarLiN clusters slightly outperform CW. We do not evaluate bag-of-words models such as WORD2VEC (Mikolov et al., 2013), because the ordering of words is essential for finding morphological properties.

Accumulated tag counts (ACT) are a form of task-specific sparse representation. The unlabeled corpus is first annotated by a tagger; for each occurring word form, the number of times a specific tag was assigned can then be used as a representation. Goldberg and Elhadad (2013) and (Szántó and Farkas, 2014) show that using such information in the word-preterminal emission probabilities of PCFGs can improve parsing accuracy. Specifically, Szántó and Farkas (2014) show that this approach performs as well as an MA in some cases. We find MAs to be more effective than the accumulated count embeddings; this is not a contradiction as we try to improve the performance of the tagger itself.

4 Data Preparation

Our test suite consists of data sets for six different languages: Czech (cs), English (en), German (de), Hungarian (hu), Spanish (es) and Latin (la). Czech, German, Hungarian and Latin are morphologically rich. We chose these languages because they represent different families: Germanic (English, German), Romance (Latin, Spanish), Slavic (Czech) and Finno-Ugric (Hungarian) and different degrees of morphological complexity and syncretism. For example, English and Spanish rarely mark case while the other languages do; and as an agglutinative language, Hungarian features a low number of possible readings for a word form while languages like German can have more than 40 different readings for a word form. An additional criterion was to have a sufficient amount of labeled OOD data. The data sets also feature an interesting selection of domain differences. For example, for Latin we have texts from different epochs while the English data contains canonical and non-canonical text.

Labeled Data. This section describes the annotation and conversion we performed to create consistent ID and OOD data sets.[2] No conversion was required for Hungarian, English and Latin as the data is already annotated in a consistent way.

For Hungarian we use the (multi-domain) Szeged Dependency Treebank (Vincze et al., 2010). We use the part that was used in the SPMRL 2013 shared task (Seddah et al., 2013) as ID data (news-wire) and an excerpt from the novel *1984* and a *Windows 2000* manual as OOD data.

For Latin we use the PROIEL treebank (Haug and Jøhndal, 2008). It consists of data from the *Vulgate* (bible text, $\approx$ 380 AD), *Commentarii de Bello Gallico* ($\approx$ 50 BC), Letters from Cicero to his friend Atticus ($\approx$ 50 BC) and *The Pilgrimage of Aetheria* ($\approx$ 380 AD). We use the biggest text source *(Vulgate)* as ID data and the remainder as OOD data.

For English we use the SANCL shared task data (Petrov and McDonald, 2012), which consists of Ontonotes 4.0 as ID data and five OOD domains from the Google Web treebank: Yahoo! Answers, weblogs, news groups, business reviews and emails. For Czech we use the part of the Prague Dependency Treebank (PDT) (Böhmová et al., 2003) that was used in the CoNLL 2009 shared tasks (Hajič et al., 2009) as ID data. We use the Czech part of the Multext East (MTE) corpus (Erjavec, 2010) as OOD data. MTE consists of translations of the

[1] The authors claim that LDA Gibbs sampling is faster than the induction of Brown clusters because it only depends linearly on the number of clusters. We, however, could not train their models on our bigger data sets as the sampling depends linearly on the number of tokens.

[2] Table 5 of the appendix provides a structured overview over the domains and resources used for each language. The appendix can be found at `http://cistern.cis.lmu.de/marmot/naacl2015/appendix.pdf`.

novel *1984* that have been annotated morphologically. PDT and MTE have been annotated using two different guidelines that without further annotation effort could only be merged by reducing them to a common subset. Specifically, we removed features such as sub POS tags as well as markers for (in)animacy. The PDT features a number of tags that are ambiguous and could not always be resolved. The gender feature Q for example can mean feminine or neuter. If we could not disambiguate such a tag, we removed it; this results in morphological tags that are not present in the MTE corpus and a relatively high number of unseen tags. Instead of describing the conversion process in greater detail we refer to our conversion scripts (Section 8).

For Spanish we use the part of the AnCora corpus (Taulé et al., 2008) of CoNLL 2009 and the IULA treebank (Marimon et al., 2012), which consists of five domains: law, economics, medicine, computer science and environment. We use the AnCora corpus as ID data set and IULA as OOD data set. The two treebanks have been annotated using the same annotation scheme, but slightly different guidelines. Similar to Czech we merged the data sets by deleting features that could not be merged or were not present in one of the treebanks. Again we refer to the conversion script for further details (Section 8).

For German we use the Tiger treebank (Brants et al., 2002) in the same split as Müller et al. (2013) as ID data and the Smultron corpus (Volk et al., 2010) as OOD data. Smultron consists of four parts: a description of Alpine hiking routes, a DVD manual, an excerpt of Sophie's World and economics texts. It has been annotated with POS and syntax, but not with morphological features. We annotated Smultron following the Tiger guidelines. The annotation process was similar to Marimon et al. (2012) in that the data sets were automatically tagged with the MORPH tagger MarMoT (Müller et al., 2013) and then manually corrected by two annotators. This tagger is a strong baseline as we could include features based on gold lemma, POS and syntax (Seeker and Kuhn, 2013). The agreement of the annotators was .9628 and the κ agreement .64.[3] As most of the differences between the annotators were cases where only one of the annotators had corrected an obvious error that the other had overlooked, the differences were resolved by the annotators themselves.

We used the provided segmentation if available and otherwise split ID data 8/1/1 into training, development and test sets and OOD data 1/1 into development and test sets if not mentioned otherwise. We thus have a classical setup of in-domain news paper text vs. prose, medical, law, economic or technical texts for Czech, German, Spanish and Hungarian. For English we have canonical vs. non-canonical data and for Latin data of different epochs (ca. 400 AD vs 50 BC). Additionally, for German one of the test domains is written in Swiss German.

Looking at some statistics of the labeled data sets,[4] we find that: Hungarian and Latin are the languages with the highest OOV rates (27% and 37%, which for reasons of consistency we will henceforth write as follows: .27 and .37); Hungarian has a very productive agglutinative morphology while the high number of Latin OOVs can be explained by the small training set (<60,000); Czech features the highest unknown tag rate (.05) as well as the highest unseen word-tag rate (.16). This can be explained by the limits of the conversion procedure we discussed above, e.g., ambiguous features like Q.

Unlabeled Data. As unlabeled data we use Wikipedia dumps from 2014 for all languages except for Latin for which we use the Patrologia Latina, a collection of clerical texts from ca. 100 AD to 1200 AD from Corpus Corporum (Roelli, 2014). We do not use the Latin version of Wikipedia because it is written by enthusiasts, not by native speakers, and contains many errors.

We preprocessed the Wikipedia dumps with WIKIPEDIAEXTRACTOR (Attardi and Fuschetto, 2013) and NLTK'S (Bird et al., 2009) implementation of PUNKT (Kiss and Strunk, 2006) to detect sentence boundaries. Tokenization was performed using MAGYARLANC (Hungarian, Zsibrita et al. (2013)), STANFORD TOKENIZER (English, Manning et al. (2014)), FREELING (Spanish, Padró and Stanilovsky (2012)) and CZECHTOK[5] (Czech). For

[3] For calculating κ, we assume that random agreement occurs when both annotators agree with the reading proposed by the tagger. We then estimate the probability of random agreement by multiplying the individual estimated probabilities of changing the proposed tagging. This yields a random agreement probability of .8965.

[4] Complete tables are in the appendix: Tables 1 and 2.

[5] http://sourceforge.net/projects/

Latin, we removed punctuation because PROIEL does not contain punctuation. We also split off the clitics *ne*, *que* and *ve* if the resulting token was accepted by LATMOR (Springmann et al. (2014)). Following common practice, we normalized the text by replacing digits with 0s.[6]

In our experiments, we extract representations for the 250,000 most frequent word types. This vocabulary size is comparable to other work; e.g., Turian et al. (2010) use 269,000 types. This threshold yields low fractions of uncovered tokens[7] for English and Latin (.009 and .02). For the other languages, this fraction rises to .04. We also extract the morphological readings of the words in this vocabulary using MAGYARLANC (Hungarian, Zsibrita et al. (2013)), FREELING (English and Spanish, Padró and Stanilovsky (2012)), SMOR (German, Schmid et al. (2004)), an MA from Charles University (Czech, Hajič (2001)) and LATMOR (Latin, Springmann et al. (2014)). Throughout this paper we extract one feature for each cluster id or MA reading of the current word form. For example, SMOR produces two readings for the German word form *erhielt* 'received': <1><SG><PAST><IND> and <2><SG><PAST><IND>, we thus fire two features representing the respective tags whenever *erhielt* is seen in the data. We also experimented with cluster indexes of neighboring uni/bigrams, but obtained no consistent improvement. For the dense embeddings we analogously extract the vector of the current word form.

5 Experiments

For all our experiments we use MarMoT (Müller et al., 2013) a joint POS and morphological tagger.[8] The CRF tagger employs a pruning strategy on forward-backward lattices to efficiently handle big tag sets and higher orders. Its feature set is similar to Ratnaparkhi (1996) and Toutanova et al. (2003) and includes prefixes, suffixes, immediate lexical context and shape features based on capitalization, special characters and digits. MarMoT was shown to be a competitive POS and morphological tagger across six languages (Müller et al., 2013). In order to make sure that it is also robust in an OOD setup we compare it to the two popular taggers SVMTool (Giménez and Marquez, 2004) and Morfette (Chrupała et al., 2008). The results are summarized in Table 1.

MarMoT uses stochastic gradient descent and produces different results in each training run. We therefore always report the average of five runs. The OOD numbers are macro-averages over the different OOD data sets of a language. [9] The tables in this paper are based on the development sets; the only exception to this is Table 5, which is based on the test set. MarMoT outperforms SVMTool and Morfette on every language and setup (ID / OOD) except for the Spanish OOD data set. For Czech, German and Latin the improvements over the best baseline are >1. Different orders of MarMoT behave as expected: higher-order models (order>1) outperform first-order models. The only exception to this is Latin. This suggests a drastic difference of the tag transition probabilities between the Latin ID and OOD data sets. Given the results in Table 1 and for simplicity we use an second-order MarMoT model in all subsequent experiments.

LM-based clustering. We first compare different implementations of LM-based clustering. The implementation of Brown clustering by Liang (2005) is most commonly used. Its hierarchical binary structure can be used to extract clusterings of varying granularity by selecting different prefixes of the path from the root to a specific word form. Following other work (Ratinov and Roth, 2009; Turian et al., 2010), we induce 1000 clusters and select path lengths 4, 6, 10 and 20. We call this representation Brown_path. We compare Brown_path to mkcls[10] (Och, 1999) and MarLiN. These implementations just induce flat clusterings of a certain size; we thus run them for cluster sizes 100, 200, 500 and 1000 to also obtain cluster ids of different sizes. The cluster sizes roughly resemble the granularity obtained in $\text{Brown}_{\text{path}}$. We call the corresponding mod-

czechtok/

[6] For statistics of the unlabeled data sets cf. Table 3 of the appendix.

[7] Cf. Table 4 in the appendix.

[8] http://cistern.cis.lmu.de/marmot/

[9] Throughout this paper we use the approximate randomization test (Yeh, 2000) to establish significance. To this end, we compare the output of the medians of the five independent models. We regard p-values $<.05$ as significant.

[10] mkcls implements a similar training algorithm as MarLiN, but uses simulated annealing, not greedy maximization.

		MarMoT (1)		MarMoT (2)		MarMoT (3)		Morfette		SVMTool	
		ID	OOD	ID	OOD	ID	OOD	ID	OOD	ID	OOD
morph	cs	93.27	77.83	**93.89**	78.52	93.86	**78.55**	91.48	76.56	91.06	75.41
	de	88.90	82.74	90.26	84.19	**90.54***	**84.30**	85.89	80.28	85.98	78.08
	es	98.21	93.24	**98.22**	93.62	98.16	93.42	97.95	**93.97***	97.96	91.36
	hu	**96.11**	89.78	96.07	**89.83**	95.92	89.70	95.47	89.18	94.72	88.44
	la	86.09	**67.90***	86.44	67.47	**86.47**	67.40	83.68	65.06	84.09	65.65

Table 1: Baseline experiments comparing MarMoT models of different orders with Morfette and SVMTool. Numbers denote average accuracies on ID and OOD development sets on the full morphological tagging task. A result significantly better than the other four ID (resp. OOD) results in its row is marked with $*$.

		$\text{Brown}_{\text{flat}}$		$\text{Brown}_{\text{path}}$		MarLiN		mkcls	
		ID	OOD	ID	OOD	ID	OOD	ID	OOD
pos	cs	99.19	97.25	99.18	97.21	99.19	**97.26**	**99.21**	**97.26**
	de	98.08	93.42	98.07	93.47	98.10	93.44	**98.11**	**93.64***
	en	96.99	91.67	97.02	91.71	97.01	91.71	**97.03**	**91.86***
	es	98.84	97.91	98.84	**97.97**	**98.87**	**97.97**	98.84	97.90
	hu	97.95	93.40	97.89	93.39	97.98	93.36	**97.99**	**93.42**
	la	96.78	86.49	96.62	86.60	96.91	**87.24**	**96.95**	87.19
morph	cs	94.20	78.95	94.23	79.01	**94.35**	**79.14**	94.32	79.11
	de	90.71	85.39	90.75	85.44	**90.78**	**85.58**	90.68	85.47
	es	98.47	95.08	98.47	95.12	**98.48**	**95.15**	**98.48**	95.13
	hu	96.60	90.57	96.52	90.54	96.60	90.64	**96.61**	**90.66**
	la	87.53	71.69	87.44	71.60	**87.87**	**72.08**	87.67	71.88

Table 2: Tagging results for LM-based models

		Baseline		MarLiN		CW	
		ID	OOD	ID	OOD	ID	OOD
pos	cs	99.00	96.80	**99.16***	**97.06**	99.12	97.00
	de	97.87	92.21	**98.03**	**93.35***	**98.03**	93.02
	en	96.92	91.12	**97.05**	91.72	97.00	**91.86***
	es	98.62	96.70	98.79	**97.82***	**98.80**	97.31
	hu	97.49	92.79	**97.94**	93.30	97.88	**93.40**
	la	95.80	81.92	**96.35***	**85.52***	95.88	84.50
morph	cs	93.89	78.52	**94.23***	**78.91**	94.10	78.80
	de	90.26	84.19	90.54	85.08	**90.59**	**85.21**
	es	98.22	93.62	**98.44**	**94.97***	**98.44**	94.32
	hu	96.07	89.83	96.47	90.60	**96.48**	**90.95***
	la	86.44	67.47	**86.95**	**70.30***	86.76	69.32

Table 3: Tagging results for CW

els $\text{Brown}_{\text{flat}}$, mkcls and MarLiN. The runtime of the Brown algorithm depends quadratically on the number of clusters while mkcls and MarLiN have linear complexity. This is reflected in the training times: For German the Brown algorithm takes $\approx 5000\,\text{min}$, mkcls $\approx 2000\,\text{min}$ and MarLiN $\approx 500\,\text{min}$.

Table 2 shows that the absolute differences between systems are small, but overall MarLiN and mkcls are better.[11] We conclude that systems based on the algorithm of Martin et al. (1998) are slightly more accurate for tagging and are several times faster than the more frequently used version of Brown et al. (1992). We thus use MarLiN for the remainder of this paper.

Neural Network Representations. We compare MarLiN with the implementation of CW by Al-Rfou et al. (2013). They extracted 64-dimensional representations for only the most frequent 100,000 word forms. To make the comparison fair, we use the intersection of our and their representation vocabularies.[12] The results in Table 3 show that MarLiN is best in 15 out of 22 cases and significantly better in eight. CW is best in 9 out of 22 cases and significantly better in two. We conclude that LM-based representations are more suited for tagging as they can be induced faster, are smaller and give better results.

SVD and ACT Representations. For the SVD-based representation we use feature ranks out of $\{500, 1000\}$ and dimensions out of $\{50, 100, 200, 500\}$. We found that l1-normalizing the vectors before and after the SVD improved results slightly. For the accumulated tag counts (ACT) we annotate the data with our baseline model and extract word-tag probabilities. The probabilities are then used as sparse real-valued features. Table 4 shows that all representations outperform the baseline. Improvements are biggest for Latin. Overall, SVD outperforms ACT and is outperformed by MarLiN and MA. MarLiN gives the best representations for POS tagging while MA outperforms MarLiN in MORPH tagging. Table 5 shows that the findings for the baseline, MarLiN and MA also hold for the test set.

[11] $\text{Brown}_{\text{path}}$ reaches the same performance as MarLiN in one case: pos/es/OOD.

[12] We also use representations from Wikipedia (instead of Corpus Corporum) for Latin to increase the similarity of the training data.

		Baseline		ACT		MarLiN		MA		SVD	
		ID	OOD	ID	OOD	ID	OOD	ID	OOD	ID	OOD
pos	cs	99.00	96.80	99.11	97.03	**99.19**	**97.26**	99.18	97.25	99.11	97.09
	de	97.87	92.21	98.00	92.92	**98.10**	**93.44***	98.00	92.87	98.09	92.88
	en	96.92	91.12	96.97	91.47	**97.01**	91.71	96.99	91.57	97.00	**91.75**
	es	98.62	96.70	98.79	97.09	**98.87**	**97.97**	**98.87**	97.89	98.80	97.16
	hu	97.49	92.79	97.84	93.15	97.98	93.36	**98.12***	**93.77***	97.86	93.30
	la	95.80	81.92	96.17	83.40	**96.91**	**87.24***	96.81	86.31	96.36	85.01
morph	cs	93.89	78.52	94.16	78.75	94.35	79.14	**94.48***	**79.41***	94.14	78.94
	de	90.26	84.19	90.56	84.78	**90.78**	85.58	90.75	**85.75**	90.69	85.15
	es	98.22	93.62	98.38	93.92	98.48	95.15	**98.56***	**95.43***	98.40	94.18
	hu	96.07	89.83	96.25	90.07	96.60	90.64	**96.83***	**91.14***	96.46	90.50
	la	86.44	67.47	86.96	68.61	87.87	72.08	**88.40***	**73.23***	87.45	70.81

Table 4: Tagging results for the baseline and four different representations

		Baseline		MarLiN		MA	
		ID	OOD	ID	OOD	ID	OOD
pos	cs	98.88	96.43	**99.11***	96.94	99.06	**96.95**
	de	97.32	91.10	**97.73***	**92.00***	97.60	91.49
	en	97.36	89.81	**97.58***	**90.65***	97.47	90.51
	es	98.66	97.94	**98.94***	98.33	98.87	**98.38**
	hu	96.84	92.11	97.08	92.95	**97.46***	**93.25***
	la	93.02	81.35	**95.20**	**87.58***	95.11	86.45
morph	cs	93.93	77.50	94.33	78.12	**94.50***	**78.37***
	de	88.41	82.78	89.18	83.91	**89.32***	**84.09**
	es	98.30	95.65	98.53	95.92	**98.54**	**96.33***
	hu	94.82	88.82	95.46	89.98	**95.85***	**90.46***
	la	82.09	65.59	84.67	71.25	**85.91***	**72.42***

Table 5: Test set results for: baseline, MarLiN, MA

			$f = 0$	$0 < f < 10$	$f >= 10$
morph	cs	MarLiN	0.29	0.22	0.11
		MA	0.37	0.35	0.16
	de	MarLiN	1.02	0.17	0.19
		MA	0.85	0.29	0.42
	es	MarLiN	1.36	0.15	0.02
		MA	1.50	0.27	0.04
	hu	MarLiN	0.62	0.18	0.00
		MA	1.07	0.20	0.03
	la	MarLiN	3.76	0.80	0.06
		MA	4.98	0.69	0.09

Table 6: Improvement compared to the baseline for different frequency ranges of words on OOD

morph	cs	MarLiN	gen 0.70	cas 0.41	pos 0.35
		MA	gen 0.85	cas 0.51	pos 0.31
	de	MarLiN	gen 1.23	pos 1.14	num 0.62
		MA	gen 1.37	pos 0.63	num 0.59
	es	MarLiN	sub 1.49	gen 1.21	pos 1.07
		MA	sub 1.34	gen 1.24	pos 1.10
	hu	MarLiN	cas 0.71	sub 0.66	pos 0.52
		MA	cas 0.88	sub 0.84	pos 0.76
	la	MarLiN	pos 5.19	cas 3.46	gen 3.25
		MA	pos 4.68	gen 3.85	cas 3.01

Table 7: Improvement compared to the baseline for different features

6 Analysis

We now analyze why MarLiN and MA perform better than the baseline. First we compare the improvements in absolute error rate over the baseline by grouping word forms by their training set frequency f. The number are shown in Table 6. We find that most of the improvement comes from OOV words. Rare words (frequency <10) show a smaller, but still important contribution while the contribution of frequent words can be almost neglected for four languages. The exception is German where frequent words contribute more to the error reduction than rare words. This could be caused by syncretisms such as in plural noun phrases where the gender is not marked in determiner and adjective and can only be derived from the head noun; e.g., the adjectives in *schwere Schulfächer* 'difficult school subjects' and *verdächtige Personen* 'suspect persons' are unmarked for gender and the correct genders (neuter vs. feminine) cannot be inferred from distributional information or suffixes for the nouns (although gender is easy to infer distributionally for singular forms of nouns).

Looking at the morphological features with the highest improvement in absolute error rate (Table 7) we find, that the features with the highest improvement are POS, SUB-POS (a finer division of POS, e.g., nouns are split into proper / common nouns), gender, case and number. For all languages POS and – if part of the annotation – SUB-POS are among the

three features with the highest improvements. Gender is also always among the three features with the highest improvements for the four languages that have gender (es, de, la, cs). We just discussed an example for German where gender could not be derived from context or inflectional suffixes. Other languages also have word forms that do not mark gender, e.g., Spanish masculine *ave* 'bird' vs. feminine *llave* 'key'. The gender can, however, easily be derived if the word representation encodes whether a word form has been seen with a specific determiner or adjective on its right or left.

Lastly, we use Jaccard similarity[13] to compare the sets of gold and predicted morphological features. Jaccard can be interpreted as a soft variant of accuracy: If the two tags are identical it yields 1 and otherwise it corresponds to the number of correctly predicted features divided by the size of the union of gold and predicted features.

morph		cs	de	es	hu	la
	accuracy	79.41	85.72	95.43	91.14	73.23
	Jaccard	89.89	90.71	96.77	93.52	83.68

This table demonstrates that the evaluation measure we have used throughout this paper – a tag counts as completely wrong if a single feature was misidentified even though all others are correct – is conservative. On a feature-by-feature basis accuracy would be much higher. The difference is largest for Czech and Latin.

7 Conclusion

We have presented a test suite for morphological tagging consisting of in-domain (ID) and out-of-domain (OOD) data sets for six languages: Czech, English, German, Hungarian, Latin and Spanish. We converted some of the data sets to obtain a reasonably consistent annotation and manually annotated the German part of the Smultron treebank. We surveyed four different word representations: SVD-reduced count vectors, LM-based clusters, accumulated tag counts and CW embeddings. We found that the LM-based clusters outperformed the other representations for POS and MORPH tagging, ID and OOD data sets and all languages. We also showed that our implementation of MarLiN (Martin et al., 1998) is an order-of-magnitude more efficient and performs slightly better than the implementation by Liang (2005). We also compared the learned representations to manually created Morphological Analyzers (MAs). We found that MarLiN outperforms MAs in POS tagging, but that it is substantially worse in morphological tagging. In our analysis of the results, we showed that both MarLiN and MAs decrease the error most for out-of-vocabulary words and for the features POS and gender.

[13] $\text{Jaccard}(U, V) = |U \cap V|/|U \cup V|$

8 Resources

As part of this publication we also release the following resources at `http://cistern.cis.lmu.de/marmot/`: (i) our implementation of MarLiN as open-source (ii) the morphological layer of the German part of the SMULTRON corpus. For easier reproducibility, we also made (iii) the preprocessed Wikipedia dumps and the induced representation dictionaries available. (iv) Morphological dictionaries were released to the extent this was compatible with the usage agreement. (v) We also published the conversion code for unifying the Spanish and Czech annotations.

Acknowledgments

We would like to thank the anonymous reviewers for their comments. The first author is a recipient of the Google Europe Fellowship in Natural Language Processing, and this research is supported by this Google Fellowship. The annotation of the SMULTRON data was supported by Deutsche Forschungsgemeinschaft (grant DFG 2246/2, Wordgraph).

References

Rami Al-Rfou, Bryan Perozzi, and Steven Skiena. 2013. Polyglot: Distributed word representations for multilingual NLP. In *Proceedings of CoNLL*.

Rie Kubota Ando and Tong Zhang. 2005. A high-performance semi-supervised learning method for text chunking. In *Proceedings of ACL*.

Giuseppe Attardi and Antonia Fuschetto. 2013. Wikipedia Extractor. `http://medialab.di.unipi.it/wiki/Wikipedia_Extractor`.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python*. " O'Reilly Media, Inc.".

John Blitzer, Ryan McDonald, and Fernando Pereira. 2006. Domain adaptation with structural correspondence learning. In *Proceedings of EMNLP*.

Phil Blunsom and Trevor Cohn. 2011. A hierarchical Pitman-Yor process HMM for unsupervised part of speech induction. In *Proceedings of ACL-HLT*.

Alena Böhmová, Jan Hajič, Eva Hajičová, and Barbora Hladká. 2003. The Prague dependency treebank. In *Proceedings of Treebanks*. Springer.

Sabine Brants, Stefanie Dipper, Silvia Hansen, Wolfgang Lezius, and George Smith. 2002. The TIGER treebank. In *Proceedings of the workshop on treebanks and linguistic theories*.

Peter F. Brown, Peter V. Desouza, Robert L. Mercer, Vincent J. Della Pietra, and Jenifer C. Lai. 1992. Class-based n-gram models of natural language. *Computational linguistics*.

Grzegorz Chrupała, Georgiana Dinu, and Josef Van Genabith. 2008. Learning morphology with Morfette. In *Proceedings of LREC*.

Grzegorz Chrupala. 2011. Efficient induction of probabilistic word classes with LDA. In *Proceedings of IJCNLP*.

Alexander Clark. 2003. Combining distributional and morphological information for part of speech induction. In *Proceedings of EACL*.

Michael Collins. 2002. Discriminative training methods for hidden Markov models: Theory and experiments with perceptron algorithms. In *Proceedings of EMNLP*.

Ronan Collobert and Jason Weston. 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of ICML*.

Tomaž Erjavec. 2010. MULTEXT-East version 4: Multilingual morphosyntactic specifications, lexicons and corpora. In *Proceedings of LREC*.

Jesús Giménez and Lluis Marquez. 2004. SVMTool: A general POS tagger generator based on support vector machines. In *Proceedings of LREC*.

Yoav Goldberg and Michael Elhadad. 2013. Word segmentation, unknown-word resolution, and morphological agreement in a Hebrew parsing system. *Computational Linguistics*.

Nizar Habash and Owen Rambow. 2005. Arabic tokenization, part-of-speech tagging and morphological disambiguation in one fell swoop. In *Proceedings of ACL*.

Jan Hajič and Barbora Hladká. 1998. Tagging inflective languages: Prediction of morphological categories for a rich, structured tagset. In *Proceedings of Coling*.

Jan Hajič, Massimiliano Ciaramita, Richard Johansson, Daisuke Kawahara, Maria Antònia Martí, Lluís Màrquez, Adam Meyers, Joakim Nivre, Sebastian Padó, Jan Štěpánek, et al. 2009. The CoNLL-2009 shared task: Syntactic and semantic dependencies in multiple languages. In *Proceedings of CoNLL*.

Jan Hajič. 2000. Morphological tagging: Data vs. dictionaries. In *Proceedings of NAACL*.

Jan Hajič. 2001. Czech Free Morphology.

Dag TT Haug and Marius Jøhndal. 2008. Creating a parallel treebank of the old Indo-European bible translations. In *Proceedings of LaTeCH*.

Zhongqiang Huang, Vladimir Eidelman, and Mary Harper. 2009. Improving a simple bigram hmm part-of-speech tagger by latent annotation and self-training. In *Proceedings of NAACL*.

Tibor Kiss and Jan Strunk. 2006. Unsupervised multilingual sentence boundary detection. *Computational Linguistics*.

Terry Koo, Xavier Carreras, and Michael Collins. 2008. Simple semi-supervised dependency parsing. In *Proceedings of ACL*.

John Lafferty, Andrew McCallum, and Fernando Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of ICML*.

Michael Lamar, Yariv Maron, Mark Johnson, and Elie Bienenstock. 2010. SVD and clustering for unsupervised POS tagging. In *Proceedings of ACL*.

Percy Liang. 2005. *Semi-supervised learning for natural language*. Ph.D. thesis, Massachusetts Institute of Technology.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of ACL: System Demonstrations*.

Montserrat Marimon, Beatriz Fisas, Núria Bel, Marta Villegas, Jorge Vivaldi, Sergi Torner, Mercè Lorente, Silvia Vázquez, and Marta Villegas. 2012. The IULA treebank. In *Proceedings of LREC*.

Sven Martin, Jorg Liermann, and Hermann Ney. 1998. Algorithms for bigram and trigram word clustering. *Speech communication*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In *Proceedings of ICLR: Workshop*.

Scott Miller, Jethran Guinness, and Alex Zamanian. 2004. Name tagging with word clusters and discriminative training. In *Proceedings of NAACL-HLT*.

Thomas Müller, Helmut Schmid, and Hinrich Schütze. 2013. Efficient higher-order CRFs for morphological tagging. In *Proceedings of EMNLP*.

Franz J. Och. 1999. An efficient method for determining bilingual word classes. In *Proceedings of EACL*.

Kemal Oflazer and Ìlker Kuruöz. 1994. Tagging and morphological disambiguation of turkish text. In *Proceedings of the Applied natural language processing*.

Lluís Padró and Evgeny Stanilovsky. 2012. Freeling 3.0: Towards wider multilinguality. In *Proceedings of LREC*.

Slav Petrov and Ryan McDonald. 2012. Overview of the 2012 shared task on parsing the web. In *Proceedings of SANCL*.

Lev Ratinov and Dan Roth. 2009. Design challenges and misconceptions in named entity recognition. In *Proceedings of CoNLL*.

Adwait Ratnaparkhi. 1996. A maximum entropy model for part-of-speech tagging. In *Proceedings of EMNLP*.

Phillip Roelli. 2014. Corpus Corporum. `http://www.mlat.uzh.ch/MLS/`.

Helmut Schmid, Arne Fitschen, and Ulrich Heid. 2004. SMOR: A German computational morphology covering derivation, composition and inflection. In *Proceedings of LREC*.

Hinrich Schütze. 1995. Distributional part-of-speech tagging. In *Proceedings of EACL*.

Djamé Seddah, Reut Tsarfaty, Sandra Kübler, Marie Candito, Jinho Choi, Richárd Farkas, Jennifer Foster, Iakes Goenaga, Koldo Gojenola, Yoav Goldberg, et al. 2013. Overview of the SPMRL 2013 shared task: Cross-framework evaluation of parsing morphologically rich languages. In *SPMRL*. Association for Computational Linguistics.

Wolfgang Seeker and Jonas Kuhn. 2013. The effects of syntactic features in automatic prediction of morphology. In *Proceedings of EMNLP*.

Noah A. Smith, David A. Smith, and Roy W. Tromble. 2005. Context-based morphological disambiguation with random fields. In *Proceedings of EMNLP*.

Drahomíra "johanka" Spoustová, Jan Hajič, Jan Raab, and Miroslav Spousta. 2009. Semi-supervised training for the averaged perceptron pos tagger. In *Proceedings of EACL*.

Uwe Springmann, Dietmar Najock, Hermann Morgenroth, Helmut Schmid, Annette Gotscharek, and Florian Fink. 2014. OCR of historical printings of Latin texts: problems, prospects, progress. In *Proceedings of DATeCH*.

Zsolt Szántó and Richárd Farkas. 2014. Special techniques for constituent parsing of morphologically rich languages. In *Proceedings of EACL*.

Mariona Taulé, Maria Antònia Martí, and Marta Recasens. 2008. Ancora: Multilevel annotated corpora for Catalan and Spanish. In *Proceedings of LREC*.

Kristina Toutanova, Dan Klein, Christopher D. Manning, and Yoram Singer. 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of NAACL*.

Joseph Turian, Lev Ratinov, and Yoshua Bengio. 2010. Word representations: A simple and general method for semi-supervised learning. In *Proceedings of ACL*.

Veronika Vincze, Dóra Szauter, Attila Almási, György Móra, Zoltán Alexin, and János Csirik. 2010. Hungarian dependency treebank. In *Proceedings of LREC*.

Martin Volk, Anne Göhring, Torsten Marek, and Yvonne Samuelsson. 2010. SMULTRON (version 3.0) — The Stockholm MULtilingual parallel TReebank.

Alexander Yeh. 2000. More accurate tests for the statistical significance of result differences. In *Proceedings of COLING*.

Deniz Yuret and Ferhan Türe. 2006. Learning morphological disambiguation rules for Turkish. In *Proceedings of NAACL*.

Yi Zhang and Rui Wang. 2009. Cross-domain dependency parsing using a deep linguistic grammar. In *Proceedings of ACL-AFNLP*.

János Zsibrita, Veronika Vincze, and Richárd Farkas. 2013. magyarlanc: A toolkit for morphological and dependency parsing of Hungarian. In *Proceedings of RANLP*.