Unsupervised Disambiguation of Syncretism in Inflected Lexicons

Ryan Cotterell and **Christo Kirov** and **Sabrina J. Mielke** and **Jason Eisner**
Department of Computer Science, Johns Hopkins University
{ryan.cotterell@,ckirov1@,sjmielke@,jason@cs}.jhu.edu

Abstract

Lexical ambiguity makes it difficult to compute various useful statistics of a corpus. A given word form might represent any of several morphological feature bundles. One can, however, use unsupervised learning (as in EM) to fit a model that probabilistically disambiguates word forms. We present such an approach, which employs a neural network to smoothly model a prior distribution over feature bundles (even rare ones). Although this basic model does not consider a token's context, that very property allows it to operate on a simple list of unigram type counts, partitioning each count among different analyses of that unigram. We discuss evaluation metrics for this novel task and report results on 5 languages.

1 Introduction

Inflected lexicons—lists of morphologically inflected forms—are commonplace in NLP. Such lexicons currently exist for over 100 languages in a standardized annotation scheme (Kirov et al., 2018), making them one of the most multi-lingual annotated resources in existence. These lexicons are typically annotated at the type level, i.e., each word type is listed with its *possible* morphological analyses, divorced from sentential context.

One might imagine that most word types are unambiguous. However, many inflectional systems are replete with a form of ambiguity termed syncretism—a systematic merger of morphological slots. In English, some verbs have five distinct inflected forms, but regular verbs (the vast majority) merge two of these and so distinguish only four. The verb *sing* has the past tense form sang but the participial form sung; the verb *talk*, on the other hand, employs talked for both functions. The form talked is, thus, said to be syncretic. Our task is to partition the count of talked in a corpus between the past-tense and participial readings, respectively.

	SG	PL	SG	PL
NOM	**Wort**	**Wörter**	Herr	**Herren**
GEN	Wortes	**Wörter**	**Herrn**	**Herren**
ACC	**Wort**	**Wörter**	**Herrn**	**Herren**
DAT	Worte	Wörtern	**Herrn**	**Herren**

Table 1: Full paradigms for the German nouns *Wort* ("word") and *Herr* ("gentleman") with abbreviated and tabularized UniMorph annotation. The syncretic forms are bolded and colored by ambiguity class. Note that, while in the plural the nominative and accusative are always syncretic across all paradigms, the same is not true in the singular.

In this paper, we model a generative probability distribution over *annotated* word forms, and fit the model parameters using the token counts of *unannotated* word forms. The resulting distribution predicts how to partition each form's token count among its possible annotations. While our method actually deals with all ambiguous forms in the lexicon, it is particularly useful for syncretic forms because syncretism is often systematic and pervasive.

In English, our unsupervised procedure learns from the counts of irregular pairs like sang–sung that a verb's past tense tends to be more frequent than its past participle. These learned parameters are then used to disambiguate talked. The method can also learn from regular paradigms. For example, it learns from the counts of pairs like runs–run that singular third-person forms are common. It then uses these learned parameters to guess that tokens of run are often singular or third-person (though never both at once, because the lexicon does not list that as a possible analysis of run).

Proceedings of NAACL-HLT 2018, pages 548–553
New Orleans, Louisiana, June 1 - 6, 2018.

2 Formalizing Inflectional Morphology

We adopt the framework of word-based morphology (Aronoff, 1976; Spencer, 1991). In the present paper, we consider only inflectional morphology. An **inflected lexicon** is a set of word types. Each **word type** is a 4-tuple of a part-of-speech tag, a lexeme, an inflectional slot, and a surface form.

A **lexeme** is a discrete object (represented by an arbitrary integer or string, which we typeset in *cursive*) that indexes the word's core meaning and part of speech. A **part-of-speech (POS) tag** is a coarse syntactic category such as VERB. Each POS tag allows some set of lexemes, and also allows some set of inflectional **slots** such as "1st-person present singular." Each allowed ⟨tag, lexeme, slot⟩ triple is realized—in only one way—as an inflected **surface form**, a string over a fixed phonological or orthographic alphabet Σ. In this work, we take Σ to be an orthographic alphabet.

A **paradigm** $\pi(t, \ell)$ is the mapping from tag t's slots to the surface forms that "fill" those slots for lexeme ℓ. For example, in the English paradigm $\pi(\text{VERB}, \textit{talk})$, the past-tense slot is said to be filled by talked, meaning that the lexicon contains the tuple $\langle \text{VERB}, \textit{talk}, \text{PAST}, \text{talked} \rangle$.[1]

We will specifically work with the UniMorph annotation scheme (Sylak-Glassman, 2016). Here each slot specifies a morpho-syntactic bundle of inflectional features (also called a morphological tag in the literature), such as tense, mood, person, number, and gender. For example, the German surface form Wörtern is listed in the lexicon with tag NOUN, lemma *Wort*, and a slot specifying the feature bundle $\left[\text{NUM=PL}, \text{CASE=DAT}\right]$. An example of UniMorph annotation is found in Table 1.

2.1 What is Syncretism?

We say that a surface form f is **syncretic** if two slots $s_1 \neq s_2$ exist such that some paradigm $\pi(t, \ell)$ maps both s_1 and s_2 to f. In other words, a single form fills multiple slots in a paradigm: syncretism may be thought of as intra-paradigmatic ambiguity. This definition does depend on the exact annotation scheme in use, as some schemes collapse syncretic slots. For example, in German nouns, *no* lexeme distinguishes the nominative, accusative and genitive plurals. Thus, a human-created lexicon might employ a single slot $\left[\text{NUM=PL}, \text{CASE=NOM/ACC/GEN}\right]$ and say that Wörter fills just this slot rather than three separate slots. For a discussion, see Baerman et al. (2005).

[1] Lexicographers will often refer to a paradigm by its **lemma**, which is the surface form that fills a certain designated slot such as the infinitive. We instead use lexemes because lemmas may be ambiguous: bank is the lemma for at least two nominal and two verbal paradigms.

2.2 Inter-Paradigmatic Ambiguity

A different kind of ambiguity occurs when a surface form belongs to more than one paradigm. A form f is inter-paradigmatically ambiguous if $\langle t_1, \ell_1, s_1, f \rangle$ and $\langle t_2, \ell_2, s_2, f \rangle$ are both in the lexicon for lexemes $\langle t_1, \ell_1 \rangle \neq \langle t_2, \ell_2 \rangle$.

For example, talks belongs to the English paradigms $\pi(\text{VERB}, \textit{talk})$ and $\pi(\text{NOUN}, \textit{talk})$. The model we present in §3 will resolve both syncretism and inter-paradigmatic ambiguity. However, our exposition focuses on the former, as it is cross-linguistically more common.

2.3 Disambiguating Surface Form Counts

The previous sections §2.1 and §2.2 discussed two types of ambiguity found in inflected lexicons. The goal of this paper is the *disambiguation* of raw surface form counts, taken from an unannotated text corpus. In other words, given such counts, we seek to impute the fractional counts for individual lexical entries (4-tuples), which are unannotated in raw text. Let us assume that the word talked is observed $c\,(\text{talked})$ times in a raw English text corpus. We do not know which instances of talked are participles and which are past tense forms. However, given a probability distribution $p_{\boldsymbol{\theta}}(t, \ell, s \mid f)$, we may disambiguate these counts in expectation, i.e., we attribute a count of

$$c\,(\text{talked}) \cdot p_{\boldsymbol{\theta}}(\text{VERB}, \textit{talk}, \text{PAST\_PART} \mid \text{talked})$$

to the past participle of the VERB *talk*. Our aim is the construction and unsupervised estimation of the distribution $p_{\boldsymbol{\theta}}(t, \ell, s \mid f)$.

While the task at hand is novel, what applications does it have? We are especially interested in *sampling* tuples $\langle t, \ell, s, f \rangle$ from an inflected lexicon. Sampling is a necessity for creating train-test splits for evaluating morphological inflectors, which has recently become a standard task in the literature (Durrett and DeNero, 2013; Hulden et al., 2014; Nicolai et al., 2015; Faruqui et al., 2016), and has seen two shared tasks (Cotterell et al., 2016, 2017). Creating train-test splits for training inflectors involves sampling *without replacement* so that all test types are unseen. Ideally, we would like more frequent word types in the training portion and less

frequent ones in the test portion. This is a realistic evaluation: a training lexicon for a new language would tend to contain frequent types, so the system should be tested on its ability to extrapolate to rarer types that could not be looked up in that lexicon, as discussed by Cotterell et al. (2015). To make the split, we sample N word types without replacement, which is equivalent to collecting the first N distinct forms from an annotated corpus generated from the same unigram distribution.

The fractional counts that our method estimates may also be useful for corpus linguistics—for example, tracking the frequency of specific lexemes over time, or comparing the rate of participles in the work of two different authors.

Finally, the fractional counts can aid the training of NLP methods that operate on a raw corpus, such as distributional embedding of surface form types into a vector space. Such methods sometimes consider the morphological properties (tags, lexemes, and slots) of nearby context words. When the morphological properties of a context word f are ambiguous, instead of tagging (which may not be feasible) one could *fractionally* count the occcurrences of the possible analyses according to $p_{\boldsymbol{\theta}}(t, \ell, s \mid f)$, or else characterize f's morphology with a single *soft* indicator vector whose elements are the probabilities of the properties according to $p_{\boldsymbol{\theta}}(t, \ell, s \mid f)$.

3 A Neural Latent Variable Model

In general, we will only observe unannotated word forms f. We model these as draws from a distribution over form types $p_{\boldsymbol{\theta}}(f)$, which marginalizes out the unobserved structure of the lexicon—which tag, lexeme and slot generated each form. Training the parameters of this latent-variable model will recover the posterior distribution over analyses of a form, $p_{\boldsymbol{\theta}}(t, \ell, s \mid f)$, which allows us to disambiguate counts at the type level.

The latent-variable model is a Bayesian network,

$$p_{\boldsymbol{\theta}}(f) = \sum_{\langle t,\ell,s\rangle \in \mathcal{T}\times\mathcal{L}\times\mathcal{S}} p_{\boldsymbol{\theta}}(t)\, p_{\boldsymbol{\theta}}(\ell \mid t)\, p_{\boldsymbol{\theta}}(s \mid t)\, \delta(f \mid t, \ell, s) \tag{1}$$

where $\mathcal{T}, \mathcal{L}, \mathcal{S}$ range over the possible tags, lexemes, and slots of the language, and $\delta(f \mid t, \ell, s)$ returns 1 or 0 according to whether the lexicon lists f as the (unique) realization of $\langle t, \ell, s\rangle$. We fix $p_\theta(s \mid t) = 0$ if the lexicon lists no tuples of the form $\langle t, \cdot, s, \cdot\rangle$, and otherwise model

$$p_{\boldsymbol{\theta}}(s \mid t) \propto \exp\left(\mathbf{u}^\top \tanh\left(\mathbf{W} \cdot \mathbf{v}_{t,s}\right)\right) > 0 \tag{2}$$

where $\mathbf{v}_{t,s}$ is a multi-hot vector whose "1" components indicate the morphological features possessed by $\langle t, s\rangle$: namely attribute-value pairs such as POS=VERB and NUM=PL. Here $\mathbf{u} \in \mathbb{R}^d$ and $\mathbf{W}$ is a conformable matrix of weights. This formula specifies a neural network with d hidden units, which can learn to favor or disfavor specific soft conjunctions of morphological features. Finally, we define $p_{\boldsymbol{\theta}}(t) \propto \exp \omega_t$ for $t \in \mathcal{T}$, and $p_{\boldsymbol{\theta}}(\ell \mid t) \propto \exp \omega_{t,\ell}$ or 0 if the lexicon lists no tuples of the form $\langle t, \ell, \cdot, \cdot\rangle$. The model's parameter vector $\boldsymbol{\theta}$ specifies $\mathbf{u}$, $\mathbf{W}$, and the ω values.

3.1 Inference and Learning

We maximize the regularized log-likelihood

$$\sum_{f \in \mathcal{F}} c(f) \log p_{\boldsymbol{\theta}}(f) + \frac{\lambda}{2} ||\boldsymbol{\theta}||_2^2 \tag{3}$$

where $\mathcal{F}$ is the set of surface form types and $p_\theta(f)$ is defined by (1). It is straightforward to use a gradient-based optimizer, and we do. However, (3) could also be maximized by an intuitive EM algorithm: at each iteration, the E-step uses the current model parameters to partition each count $c(f)$ among possible analyses, as in (2.3), and then the M step improves the parameters by following the gradient of *supervised* regularized log-likelihood as if it had observed those fractional counts.

On each iteration, either algorithm loops through all listed (t, s) pairs, all listed (t, ℓ) pairs, and all observed forms f, taking time at most proportional to the size of the lexicon. In practice, training completes within a few minutes on a modern laptop.

3.2 Baseline Models

To the best of our knowledge, this disambiguation task is novel. Thus, we resort to comparing three variants of our model in lieu of a previously published baseline. We evaluate three simplifications of the slot model, to investigate whether the complexity of equation (2) is justified.

UNIF: $p(s \mid t)$ is uniform over the slots s that are listed with t. This involves no learning.

FREE: $p(s \mid t) \propto \exp \omega_{t,s}$: a model with a single parameter $\omega_{t,s} \in \mathbb{R}$ per slot. This can capture any distribution, but it has less inductive bias:

slots that share morphological features do not share parameters.

LINEAR: $p(s \mid t) \propto \exp(\mathbf{u}^\top \mathbf{v}_{t,s})$: a linear model with no conjunctions between morphological features. This chooses the features orthogonally, in the sense that (e.g.) if verbal paradigms have a complete 3-dimensional grid of slots indexed by their PERSON, NUM, and TENSE attributes, then sampling from $p(s \mid \text{VERB})$ is equivalent to independently sampling these three coordinates. Moreover, $p(\text{NUM=PL} \mid \text{NOUN}) = p(\text{NUM=PL} \mid \text{VERB})$.

4 Experiments

4.1 Computing Evaluation Metrics

We first evaluate **perplexity**. Since our model is a tractable generative model, we may easily evaluate its perplexity on held-out tokens. For each language, we randomly partition the observed surface tokens into 80% training, 10% development, and 10% test. We then estimate the parameters of our model by maximizing (3) on the counts from the training portion, selecting hyperparameters such that the estimated parameters[2] minimize perplexity on the development portion. We then report perplexity on the test portion.

Using the same hyperparameters, we now train our latent-variable model $p_{\boldsymbol{\theta}}$ *without* supervision on 100% of the observed surface forms f. We now measure how poorly, for the average surface form type f, we recovered the maximum-likelihood distribution $\hat{p}(t, \ell, s \mid f)$ that would be estimated *with* supervision in terms of **KL-divergence**:

$$\sum_f \hat{p}(f)\ \mathrm{KL}(\hat{p}(\cdot \mid f) \mid\mid p_{\boldsymbol{\theta}}(\cdot \mid f)) \qquad (4)$$
$$= \frac{1}{N} \sum_{i=1}^{N} \log_2 \frac{\hat{p}(t_i, \ell_i, s_i \mid f_i)}{p_{\boldsymbol{\theta}}(t_i, \ell_i, s_i \mid f_i)}$$

We can see that this formula reduces to a simple average over disambiguated tokens i.

4.2 Training Details and Hyperparameters

We optimized on training data using batch gradient descent with a fixed learning rate. We used perplexity on development data to jointly choose the learning rate, the initial random $\boldsymbol{\theta}$ (from among several random restarts), the regularization coefficient $\lambda \in \{10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}\}$ and the neural network architecture. The NEURAL architecture shown in eq. (2) has 1 hidden layer, but we actually generalized this to consider networks with $k \in \{1, 2, 3, 4\}$ hidden layers of $d = 100$ units each. In some cases, the model selected on development data had k as high as 3. Note that the LINEAR model corresponds to $k = 0$.

4.3 Datasets

Each language constitutes a separate experiment. In each case we obtain our lexicon from the UniMorph project and our surface form counts from Wikipedia. To approximate supervised counts to estimate $\hat{p}$ in the KL evaluation, we analyzed the surface form tokens in Wikipedia (in context) using the tool in Straka et al. (2016), as trained on the disambiguated Universal Dependencies (UD) corpora (Nivre et al., 2016). We wrote a script[3] to convert the resulting analyses from UD format into $\langle t, \ell, s, f \rangle$ tuples in UniMorph format for five languages—Czech (cs), German (de), Finnish (fi), Hebrew (he), Swedish (sv)—each of which displays both kinds of ambiguity in its UniMorph lexicon. Lexicons with these approximate supervised counts are provided as supplementary material.

4.4 Results

Our results are graphed in Fig. 1, exact numbers are found in Table 2. We find that the NEURAL model slightly outperforms the other baselines on languages except for German. The LINEAR model is quite competitive as well.

	NEURAL NET		FREE		LINEAR		UNIFORM	
lang	perp	KL	perp	KL	perp	KL	perp	KL
cs	**621**	**0.56**	643	0.58	637	0.67	896	1.19
de	776	2.39	**775**	**2.25**	776	2.33	813	3.03
fi	**300**	**0.99**	319	1.18	304	1.03	889	2.61
he	**96**	**0.27**	130	0.69	97	0.29	675	3.69
sv	**547**	**0.06**	565	0.14	568	0.08	1025	1.5

Table 2: Results for the best performing neural network (hyperparameters selected on dev) and the three baselines under both performance metrics. Best are bolded.

UNIF would have a KL divergence of 0 bits if all forms were either unambiguous or uniformly ambiguous. Its higher value means the unsupervised task is nontrivial. Our other models substantially

[2] Our vocabulary and parameter set are determined from the *lexicon*. Thus we create a regularized parameter ω_ℓ, yielding a smoothed estimate $p(\ell)$, even if the training count $c(\ell) = 0$.

[3] The script discarded up to 31% of the tokens because the UD analysis could not be successfully converted into an UniMorph analysis that was present in the lexicon.

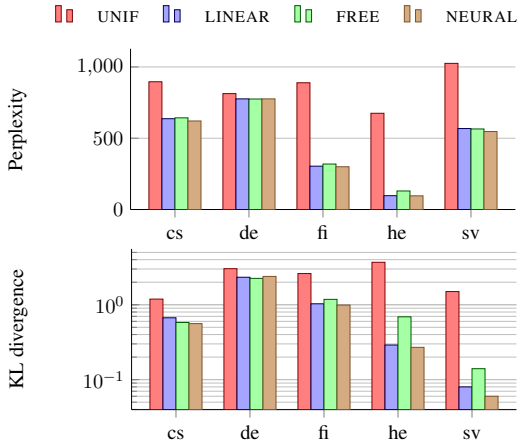

Figure 1: Unsupervised and supervised test results under each model, averaged over 50 training-dev-test splits.

outperform UNIF. NEURAL matches the supervised distributions reasonably closely, achieving an average KL of < 1 bit on all languages but German.

5 Related Work

By far the closest work to ours is the seminal paper of Baayen and Sproat (1996), who asked the following question: "Given a form that is previously unseen in a sufficiently large training corpus, and that is morphologically n-ways ambiguous [...] what is the best estimator for the lexical prior probabilities for the various functions of the form?" While we address the same task, i.e., estimation of a lexical prior, Baayen and Sproat (1996) assume supervision in the form of an disambiguated corpus. We are the first to treat the specific task in an unsupervised fashion. We discuss other work below.

Supervised Morphological Tagging. Morphological tagging is a common task in NLP; the state of the art is currently held by neural models (Heigold et al., 2017). This task is distinct from the problem at hand. Even if a tagger obtains the possible analyses from a lexicon, it is still trained in a supervised manner to choose among analyses.

Unsupervised POS Tagging. Another vein of work that is similar to ours is that of unsupervised part-of-speech (POS) tagging. Here, the goal is map sequences of forms into coarse-grained syntactic categories. Christodoulopoulos et al. (2010) provide a useful overview of previous work. This task differs from ours on two counts. First, we are interested in finer-grained morphological distinctions: the universal POS tagset (Petrov et al., 2012) makes 12 distinctions, whereas UniMorph has languages expressing hundreds of distinctions. Second, POS tagging deals with the induction of syntactic categories from sentential context.

We note that purely unsupervised morphological tagging, has yet to be attempted to the best of our knowledge.

6 Conclusion

We have presented a novel generative latent-variable model for resolving ambiguity in unigram counts, notably due to syncretism. Given a lexicon, an unsupervised model partitions the corpus count for each ambiguous form among its analyses listed in a lexicon. We empirically evaluated our method on 5 languages under two evaluation metrics. The code is availabile at `https://sjmielke.com/papers/syncretism`, along with type-disambiguated unigram counts for all lexicons provided by the UniMorph project (100+ languages).

References

Mark Aronoff. 1976. *Word Formation in Generative Grammar*. Number 1 in Linguistic Inquiry Monographs. MIT Press, Cambridge, MA.

Harald Baayen and Richard Sproat. 1996. Estimating lexical priors for low-frequency morphologically ambiguous forms. *Computational Linguistics*, 22(2):155–166.

Matthew Baerman, Dunstan Brown, and Greville G. Corbett. 2005. *The Syntax-Morphology Interface: A study of Syncretism*, volume 109. Cambridge University Press.

Christos Christodoulopoulos, Sharon Goldwater, and Mark Steedman. 2010. Two decades of unsupervised POS induction: How far have we come? In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 575–584, Cambridge, MA. Association for Computational Linguistics.

Ryan Cotterell, Christo Kirov, John Sylak-Glassman, Géraldine Walther, Ekaterina Vylomova, Patrick Xia, Manaal Faruqui, Sandra Kübler, David Yarowsky, Jason Eisner, and Mans Hulden. 2017. The CoNLL-SIGMORPHON 2017 shared task: Universal morphological reinflection in 52 languages. In *Proceedings of the CoNLL-SIGMORPHON 2017 Shared Task: Universal Morphological Reinflection*, Vancouver, Canada. Association for Computational Linguistics.

Ryan Cotterell, Christo Kirov, John Sylak-Glassman, David Yarowsky, Jason Eisner, and Mans Hulden. 2016. The SIGMORPHON 2016 shared task—morphological reinflection. In *Proceedings of the*

14th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology, pages 10–22, Berlin, Germany. Association for Computational Linguistics.

Ryan Cotterell, Nanyun Peng, and Jason Eisner. 2015. Modeling word forms using latent underlying morphs and phonology. *Transactions of the Association for Computational Linguistics*, 3:433–447.

Greg Durrett and John DeNero. 2013. Supervised learning of complete morphological paradigms. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*, pages 1185–1195, Atlanta, Georgia. Association for Computational Linguistics.

Manaal Faruqui, Yulia Tsvetkov, Graham Neubig, and Chris Dyer. 2016. Morphological inflection generation using character sequence to sequence learning. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*, pages 634–643, San Diego, California. Association for Computational Linguistics.

Georg Heigold, Guenter Neumann, and Josef van Genabith. 2017. An extensive empirical evaluation of character-based morphological tagging for 14 languages. In *Proceedings of the Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 505–513, Valencia, Spain. Association for Computational Linguistics.

Mans Hulden, Markus Forsberg, and Malin Ahlberg. 2014. Semi-supervised learning of morphological paradigms and lexicons. In *Proceedings of the Conference of the European Chapter of the Association for Computational Linguistics*, pages 569–578, Gothenburg, Sweden. Association for Computational Linguistics.

Christo Kirov, Ryan Cotterell, John Sylak-Glassman, Géraldine Walther, Ekaterina Vylomova, Patrick Xia, Manaal Faruqui, Sabrina J. Mielke, Arya McCarthy, Sandra Kübler, David Yarowsky, Jason Eisner, and Mans Hulden. 2018. Unimorph 2.0: Universal morphology. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC)*. European Language Resources Association (ELRA).

Garrett Nicolai, Colin Cherry, and Grzegorz Kondrak. 2015. Inflection generation as discriminative string transduction. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*, pages 922–931, Denver, Colorado. Association for Computational Linguistics.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajic, Christopher D. Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, Reut Tsarfaty, and Daniel Zeman. 2016. Universal dependencies v1: A multilingual treebank collection. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC)*, Paris, France. European Language Resources Association (ELRA).

Slav Petrov, Dipanjan Das, and Ryan McDonald. 2012. A universal part-of-speech tagset. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC)*, Istanbul, Turkey. European Language Resources Association (ELRA).

Andrew Spencer. 1991. *Morphological Theory: An Introduction to Word Structure in Generative Grammar*. Wiley-Blackwell.

Milan Straka, Jan Hajič, and Jana Straková. 2016. UDPipe: Trainable pipeline for processing CoNLL-U files performing tokenization, morphological analysis, POS tagging and parsing. In *LREC*.

John Sylak-Glassman. 2016. The composition and use of the universal morphological feature schema (Unimorph schema). Technical report, Johns Hopkins University.