Contextualization of Morphological Inflection

Ekaterina Vylomova[ə] **Ryan Cotterell**[ʔ,fi] **Timothy Baldwin**[ə] **Trevor Cohn**[ə] and **Jason Eisner**[ʔ]

[ə]School of Computing and Information Systems, University of Melbourne, Melbourne, Australia
[fi]The Computer Laboratory, University of Cambridge, Cambridge, UK
[ʔ]Department of Computer Science, Johns Hopkins University, Baltimore, USA
{vylomovae,tbaldwin,tcohn}@unimelb.edu.au
rdc42@cam.ac.uk jason@cs.jhu.edu

Abstract

Critical to natural language generation is the production of correctly inflected text. In this paper, we isolate the task of predicting a fully inflected sentence from its partially lemmatized version. Unlike traditional morphological inflection or surface realization, our task input does not provide "gold" tags that specify what morphological features to realize on each lemmatized word; rather, such features must be inferred from sentential context. We develop a neural hybrid graphical model that explicitly reconstructs morphological features before predicting the inflected forms, and compare this to a system that directly predicts the inflected forms without relying on any morphological annotation. We experiment on several typologically diverse languages from the Universal Dependencies treebanks, showing the utility of incorporating linguistically-motivated latent variables into NLP models.

1 Introduction

NLP systems are often required to generate grammatical text, e.g., in machine translation, summarization, dialogue, and grammar correction. One component of grammaticality is the use of contextually appropriate closed-class morphemes. In this work, we study **contextual inflection**, which has been recently introduced in the CoNLL-SIGMORPHON 2018 shared task (Cotterell et al., 2018) to directly investigate context-dependent morphology in NLP. There, a system must inflect partially lemmatized tokens in sentential context. For example, in English, the system must reconstruct the correct word sequence *two cats are sitting* from partially lemmatized sequence *two _cat_ are sitting*. Among other things, this requires: (1) identifying *cat* as a noun in this context, (2) recognizing that *cat* should be inflected as plural to agree with the nearby verb and numeral, and (3) realizing this inflection as the suffix *s*. Most past work in supervised computational morphology—including the previous CoNLL-SIGMORPHON shared tasks on morphological reinflection (Cotterell et al., 2017)—has focused mainly on step (3) above.

As the task has been introduced into the literature only recently, we provide some background. Contextual inflection amounts to a highly constrained version of language modeling. Language modeling predicts all words of a sentence from scratch, so the usual training and evaluation metric—perplexity—is dominated by the language model's ability to predict *content*, which is where most of the uncertainty lies. Our task focuses on just the ability to reconstruct certain missing parts of the sentence—inflectional morphemes and their orthographic realization. This refocuses the modeling effort from semantic coherence to morphosyntactic coherence, an aspect of language that may take a back seat in current language models (see Linzen et al., 2016; Belinkov et al., 2017). Contextual inflection does not perfectly separate grammaticality modeling from content modeling: as illustrated in Tab. 1, mapping *two cats _be_ sitting* to the fully-inflected *two cats were sitting* does not require full knowledge of English grammar—the system does not have to predict the required word order nor the required auxiliary verb *be*, as these are supplied in the input. Conversely, this example does still require predicting some content—the semantic choice of past tense is *not* given by the input and must be guessed by the system.[1]

The primary contribution of this paper is a novel structured neural model for contextual inflection. The model first predicts the sequence of morphological tags from the partially lemmatized sequence and, then, it uses the predicted tag and lemma to inflect the word. We use this model

[1]This morphological feature is *inherent* in the sense of Booij (1996).

Proceedings of NAACL-HLT 2019, pages 2018–2024
Minneapolis, Minnesota, June 2 - June 7, 2019.

Context:	*two*	*cats*	--	*sitting*
Lemmata:	two	cat	be	sit
Tags:	POS=NUM	POS=NOUN Num=Plur	--	POS=VERB Tense=Pres VerbForm=Part
Target:	*two*	*cats*	*were*	*sitting*

Table 1: Example data entry: the target word *be* should be properly inflected into *were* to fit the sentential context.

to evince a simple point: models are better off jointly predicting morphological tags from context than directly learning to inflect lemmata from sentential context. Indeed, none of the participants in the 2018 shared task jointly predicted tags with the inflected forms. Comparing our new model to several competing systems, we show our model has the best performance on the majority of languages. We take this as evidence that predicting morphological tags jointly with inflecting is a better method for this task. Furthermore, we provide an analysis discussing the role of morphological complexity in model performance.

2 Joint Tagging and Inflection

Given a language, let $\mathcal{M}$ be a set of morphological tags in accordance with the Universal Dependencies annotation (Nivre et al., 2016). Each $m \in \mathcal{M}$ has the form $m = \langle t, \sigma \rangle$, where t is a part of speech, and the slot σ is a set of attribute–value pairs that represent morphosyntactic information, such as number, case, tense, gender, person, and others. We take $t \in \mathcal{T}$, the set of universal parts of speech described by Petrov et al. (2012). A sentence consists of a finite word sequence $\mathbf{w}$ (we use boldface for sequence variables). For every word w_i in the sequence, there is a corresponding analysis in terms of a morphological tag $m_i \in \mathcal{M}$ and a lemma ℓ_i. In general, w_i is determined by the pair $\langle \ell_i, m_i \rangle$.[2] Using this notation, Cotterell et al. (2018)'s shared task is to predict a sentence $\mathbf{w}$ from its partially lemmatized form $\boldsymbol{\ell}$, inferring $\mathbf{m}$ as an intermediate latent variable. Our dataset (§3) has all three sequences for each sentence.

2.1 A Structured Neural Model

Consider an extreme case when *all* words are lemmatized.[3] We introduce a structured neural model

[2] Although w_i can sometimes be computed by concatenating ℓ_i with m_i-specific affixes, it can also be irregular.

[3] In case of partially lemmatized sequence we still train the model to predict the tags over the entire sequence, but evaluate it only for lemmatized slots.

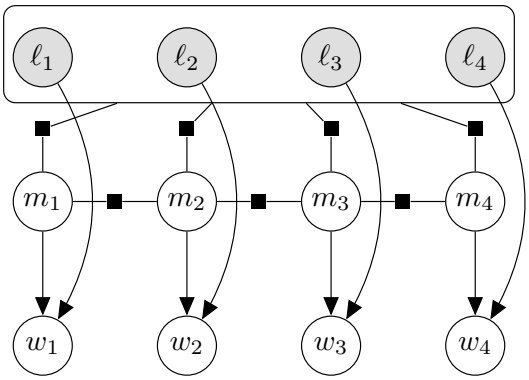

Figure 1: Our structured neural model shown as a hybrid (directed–undirected) graphical model. We omitted several arcs for convenience; namely, every morphological tag m_i depends on the entire sequence $\boldsymbol{\ell}$.

for contextual inflection, as follows:

$$p(\mathbf{w}, \mathbf{m} \mid \boldsymbol{\ell}) = \left(\prod_{i=1}^{n} p(w_i \mid \ell_i, m_i) \right) p(\mathbf{m} \mid \boldsymbol{\ell}) \tag{1}$$

In other words, the distribution is over interleaved sequences of one-to-one aligned inflected words and morphological tags, conditioned on a lemmatized sequence, all of length n. This distribution is drawn as a hybrid (directed–undirected) graphical model (Koller and Friedman, 2009) in Fig. 1. We define the two conditional distributions in the model in §2.2 and §2.3, respectively.

2.2 A Neural Conditional Random Field

The distribution $p(\mathbf{m} \mid \boldsymbol{\ell})$ is defined to be a conditional random field (CRF; Lafferty et al., 2001). In this work, our CRF is a conditional distribution over morphological taggings of an input sequence. We define this conditional distribution as

$$p(\mathbf{m} \mid \boldsymbol{\ell}) = \frac{1}{Z(\boldsymbol{\ell})} \prod_{i=1}^{n} \psi\left(m_i, m_{i-1}, \boldsymbol{\ell}\right) \tag{2}$$

where $\psi(\cdot, \cdot, \cdot) \geq 0$ is an arbitrary potential, $Z(\boldsymbol{\ell})$ normalizes the distribution, and m_0 is a distinguished start-of-sequence symbol.

In this work, we opt for a recurrent neural potential—specifically, we adopt a parameterization similar to the one given by Lample et al. (2016). Our potential ψ is computed as follows. First, the sequence $\boldsymbol{\ell}$ is encoded into a sequence of word vectors using the strategy described by Ling et al. (2015): word vectors are passed to a bidirectional LSTM (Graves et al., 2005), where the corresponding hidden states are concatenated at each time step. We simply refer to the hidden state $\mathbf{h}_i \in \mathbb{R}^d$ as the result of said concatenation at the i-th step. Using $\mathbf{h}_i$, we can define the potential function as $\psi(m_i, m_{i-1}) = \exp\left(A_{m_i,m_{i-1}} + \mathbf{o}_{m_i}^\top \mathbf{h}_i\right)$, where $A_{m_i,m_{i-1}}$ is a transition weight matrix and $\mathbf{o}_{m_i} \in \mathbb{R}^d$ is a morphological tag embedding; both are learned.

2.3 The Morphological Inflector

The conditional distribution $p(w_i \mid \ell_i, m_i)$ is parameterized by a neural encoder–decoder model with hard attention from Aharoni and Goldberg (2017). The model was one of the top performers in the 2016 SIGMORPHON shared task (Cotterell et al., 2016); it achieved particularly high accuracy in the low-resource setting. Hard attention is motivated by the observation that alignment between the input and output sequences is often monotonic in inflection tasks. In the model, the input lemma is treated as a sequence of characters, and encoded using a bidirectional LSTM (Graves and Schmidhuber, 2005), to produce vectors $\mathbf{x}_j$ for each character position j. Next the word $w_i = \mathbf{c} = c_1 \cdots c_{|w_i|}$ is generated in a decoder character-by-character:

$$p(c_j \mid \mathbf{c}_{<j}, l_i, m_i) = \text{softmax}\left(\mathbf{W} \cdot \phi(\mathbf{z}_1, \ldots, \mathbf{z}_j) + \mathbf{b}\right) \quad (3)$$

where $\mathbf{z}_j$ is the concatenation of the current attended input $\mathbf{x}_j$ alongside morphological features, m_i, and an embedding of the previously generated symbol c_{j-1}; and finally ϕ is an LSTM over the sequence of $\mathbf{z}_j$ vectors. The decoder additionally predicts a type of operation.[4] The distribution in Eq. (3), strung together with the other conditionals, yields a joint distribution over the entire character sequence:

$$p(\mathbf{c} \mid \ell_i, m_i) = \prod_{j=1}^{|w_i|} p(c_j \mid \mathbf{c}_{<j}, \ell_i, m_i) \quad (4)$$

For instance, to map the lemma *talk* to its past form *talked*, we feed in `POS=V;Tense=PAST <w> t a l k </w>` and train the network to output `<w> t a l k e d </w>`, where we have augmented the orthographic character alphabet Σ with the feature–attribute pairs that constitute the morphological tag m_i.

2.4 Parameter Estimation and Decoding

We optimize the log-likelihood of the training data with respect to all model parameters. As Eq. (1) is differentiable, this is achieved with standard gradient-based methods. For decoding we use a greedy strategy where we first decode the CRF, that is, we solve the problem $\mathbf{m}^\star = \text{argmax}_{\mathbf{m}} \log p(\mathbf{m} \mid \boldsymbol{\ell})$, using the Viterbi (1967) algorithm. We then use this decoded $\mathbf{m}^\star$ to generate forms from the inflector. Note that finding the one-best string under our neural inflector is intractable, and for this reason we use greedy search.

3 Experiments

Dataset. We use the Universal Dependencies v1.2 dataset (Nivre et al., 2016) for our experiments. We include all the languages with information on their lemmata and fine-grained grammar tag annotation that also have `fasttext` embeddings (Bojanowski et al., 2017), which are used for word embedding initialization.[5]

Evaluation. We evaluate our model's ability to predict: (i) the correct morphological tags from the lemma context, and (ii) the correct inflected forms. As our evaluation metric, we report 1-best accuracy for both tags and word form prediction.

Configuration. We use a word and character embedding dimensionality of 300 and 100, respectively. The hidden state dimensionality is set to 200. All models are trained with Adam (Kingma and Ba, 2014), with a learning rate of 0.001 for 20 epochs.

Baselines. We use two baseline systems: (1) the CoNLL–SIGMORPHON 2018 subtask 2 neural encoder–decoder with an attention mechanism ("SM"; Cotterell et al. (2018)), where the encoder represents a target form context as a concatenation of its lemma, its left and right word forms, their

[4] The model can be viewed as a transition system trained over aligned character-level strings to learn sequences of operations (`write` or `step`).

[5] We also choose mainly non-Wikipedia datasets to reduce any possible intersection with the data used for the *FastText* model training

Language	tag	form				
	JOINT	GOLD	JOINT	DIRECT	SM	CPH
Bulgarian	81.6	91.9	**78.8**	71.5	77.1	76.9
English	89.6	95.6	**90.4**	86.8	86.5	86.7
Basque	66.6	82.2	61.1	59.7	**61.2**	60.2
Finnish	66.0	86.5	**59.3**	51.2	56.6	56.4
Gaelic	68.3	84.5	**69.5**	64.5	68.9	66.9
Hindi	85.3	88.3	81.4	85.4	86.8	**87.5**
Italian	92.3	85.1	80.4	85.2	88.7	**90.5**
Latin	82.6	89.7	**75.7**	71.4	74.2	74.9
Polish	71.9	96.1	**74.8**	61.8	72.4	70.2
Swedish	81.9	96.0	**82.5**	75.4	78.4	80.9

Table 2: Accuracy of the models for various prediction settings. **tag** refers to tag prediction accuracy, and **form** to form prediction accuracy. Our model is JOINT; GOLD denotes form prediction conditioned on gold target morphological tags; the other columns are baseline methods.

lemmata and tag representations, and then the decoder generates the target inflected form character-by-character; and (2) a monolingual version of the best performing system of the shared task ("CPH"; Kementchedjhieva et al. (2018)) that augments the above encoder–decoder with full (sentence-level) left and right contexts (comprising of forms, their lemmata and morphological tags) as well as predicts morphological tags for a target form as an auxiliary task.[6] In both cases, the hyperparameters are set as described in Cotterell et al. (2018). We additionally evaluate the SIGMORPHON baseline system on prediction of the target form without any information on morphological tags ("DIRECT").

4 Results and Discussion

Tab. 2 presents the accuracy of our best model across all languages.[7] Below we highlight two main lessons from our error analysis that apply to a wider range of generation tasks, e.g., machine translation and dialog systems.

Directly Predicting Morphology. Tab. 2 indicates that all systems that make use of morphological tags outperform the DIRECT baseline on most languages. The comparison of our hybrid model with latent morphological tags to the direct form generation baseline in SM suggests that we should be including linguistically-motivated latent variables into models of natural language generation. We observe in Tab. 2 that predicting the tag together with the form (joint) often improves performance. The most interesting comparison here is with the multi-task CPH method, which includes morphology into the model without joint modeling; our model achieves higher results on 7/10 languages.

Morphological Complexity Matters. We observed that for languages with rich case systems, e.g., the Slavic languages (which exhibit a lot of fusion), the agglutinative Finno-Ugric languages, and Basque, performance is much worse. These languages present a broader decision space and often require inferring which morphological categories need to be in agreement in order to make an accurate prediction. This suggests that generation in languages with more morphological complexity will be a harder problem for neural models to solve. Indeed, this problem is under-explored, as the field of NLP tends to fixate on generating English text, e.g., in machine translation or dialogue system research.

Error Analysis. We focused error analysis on prediction of agreement categories. Our analysis of adjective–noun agreement category prediction suggests that our model is able to infer adjective number, gender, and case from its head noun. Verb gender, which appears only in the past tense of many Slavic languages, seems to be harder to predict. Given that the linear distance between the subject and the verb may be longer, we suspect the network struggles to learn longer-distance dependencies, consistent with the findings of Linzen et al. (2016). Overall, automatic inference of agreement categories is an interesting problem that deserves more attention, and we leave it for future work.

We also observe that most uncertainty comes from morphological categories such as noun number, noun definiteness (which is expressed morphologically in Bulgarian), and verb tense, all of which are inherent (Booij, 1996)[8] and typically cannot be predicted from sentential context if they do not participate in agreement.[9] On the other hand, aspect, although being closely related to tense, is well-predicted since it is mainly expressed as a separate lexeme. But, in general, it is still problematic to make a prediction in languages where aspect is morphologically marked or highly mixed with

[6] It has been shown to improve the model's performance.

[7] The accuracy numbers are on average higher than the ones achieved in terms of the CoNLL–SIGMORPHON 2018 subtask 2 since we did not filter out tokens that are typically not inflected (such as articles or prepositions).

[8] Such categories exist in most languages that exhibit some degree of morphological complexity.

[9] Unless there is no strong signal within a sentence such as *yesterday*, *tomorrow*, or *ago* as in the case of tense.

tense as in Basque.

We additionally compared 1-best and 10-best predictions for tags. Most mispredictions existing in 1-best lists are due to inherent categories mentioned above (that allow multiple plausible options that can fit the sentential context). Indeed, the problem is usually solved by allowing system to output 10-best lists. There, precision@10 is on average 8 points higher than precision@1.

Finally, our analysis of case category prediction on nouns shows that more common cases such as the nominative, accusative, and genitive are predicted better, especially in languages with fixed word order. On the other hand, cases that appear less frequently and on shifting positions (such as the instrumental), as well as those not associated with specific prepositions, are less well predicted. In addition, we evaluated the model's performance when *all* forms are replaced by their corresponding lemmata (as in *two cat be sit*). For freer word order languages such as Polish or Latin, we observe a substantial drop in performance because most information on inter-word relations and their roles (expressed by means of case system) is lost.

5 Related Work

The primary evaluation for most contemporary language and translation modeling research is perplexity, BLEU (Papineni et al., 2002), or METEOR (Banerjee and Lavie, 2005). Undoubtedly, such metrics are necessary for extrinsic evaluation and comparison. However, relatively few studies have focused on intrinsic evaluation of the model's mastery of grammaticality. Recently, Linzen et al. (2016) investigated the ability of an LSTM language model to capture sentential structure, by evaluating subject–verb agreement with respect to number, and showed that under strong supervision, the LSTM is able to approximate dependencies.

Taking it from the other perspective, a truer measure of grammatical competence would be a task of mapping a meaning representation to text, where the meaning representation specifies all necessary semantic content—content lemmata, dependency relations, and "inherent" closed-class morphemes (semantic features such as noun number, noun definiteness, and verb tense)—and the system is to realize this content according to the morphosyntactic conventions of a language, which means choosing word order, agreement morphemes, function words, and the surface forms of all words. Such tasks have been investigated to some extent—generating text from tectogrammatical trees (Hajic et al., 2002; Ptáček and Žabokrtský, 2006) or from an AMR graph (Song et al., 2017). Belz et al. (2011) organized a related surface realization shared task on mapping unordered and uninflected dependency trees to properly ordered inflected sentences. The generated sentences were afterwards assessed by human annotators, making the task less scalable and more time consuming. Although our task is not perfectly matched to grammaticality modeling, the upside is that it is a "lightweight" task that works directly on text. No meaning representation is required. Thus, training and test data in any language can be prepared simply by lemmatizing a naturally occurring corpus.

Finally, as a morphological inflection task, the form generation task is closely related to previous SIGMORPHON shared tasks (Cotterell et al., 2016, 2017). There, most neural models achieve high accuracy on many languages at type-level prediction of the form from its lemma and slot. The current task is more challenging in that the model has to perform token-level form generation and inherently infer the slot from the contextual environment. Our findings are in line with those from the CoNLL-SIGMORPHON 2018 shared task (Cotterell et al., 2018) and provide extra evidence of the utility of morphosyntactic features.

6 Conclusion

This work proposed a method for contextual inflection using a hybrid architecture. Evaluation over several diverse languages showed consistent improvements over state of the art. Our analysis demonstrated that the contextual inflection can be a highly challenging task, and the inclusion of morphological features prediction is an important element in such a system. We also highlighted two types of morphological categories, contextual and inherent, in which the former relies on agreement and the latter comes from a speaker's intention.

Acknowledgements

We thank all anonymous reviewers for their comments. The first author would like to acknowledge the Google PhD fellowship. The second author would like to acknowledge a Facebook Fellowship.

References

Roee Aharoni and Yoav Goldberg. 2017. Morphological inflection generation with hard monotonic attention. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 2004–2015.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72.

Yonatan Belinkov, Nadir Durrani, Fahim Dalvi, Hassan Sajjad, and James Glass. 2017. What do neural machine translation models learn about morphology? In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 861–872.

Anja Belz, Michael White, Dominic Espinosa, Eric Kow, Deirdre Hogan, and Amanda Stent. 2011. The First Surface Realisation Shared Task: Overview and evaluation results. In *Proceedings of the 13th European Workshop on Natural Language Generation*, pages 217–226.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Geert Booij. 1996. Inherent versus contextual inflection and the split morphology hypothesis authors. In *Yearbook of Morphology 1995*, pages 1–16. Springer, Netherlands.

Ryan Cotterell, Christo Kirov, John Sylak-Glassman, Gėraldine Walther, Ekaterina Vylomova, Arya D McCarthy, Katharina Kann, Sebastian Mielke, Garrett Nicolai, Miikka Silfverberg, David Yarowsky, Jason Eisner, and Mans Hulden. 2018. The CoNLL–SIGMORPHON 2018 Shared Task: Universal morphological reinflection. *Proceedings of the CoNLL SIGMORPHON 2018 Shared Task: Universal Morphological Reinflection*, pages 1–27.

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*.

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.

Alex Graves, Santiago Fernández, and Jürgen Schmidhuber. 2005. Bidirectional LSTM networks for improved phoneme classification and recognition. In *Artificial Neural Networks: Formal Models and Their Applications ICANN 2005*, pages 799–804.

Alex Graves and Jürgen Schmidhuber. 2005. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks*, 18(5-6):602–610.

Jan Hajic, Martin Cmejrek, Bonnie Dorr, Yuan Ding, Jason Eisner, Daniel Gildea, Terry Koo, Kristen Parton, Gerald Penn, Dragomir Radev, et al. 2002. Natural language generation in the context of machine translation. In *Summer workshop final report, Johns Hopkins University*.

Yova Kementchedjhieva, Johannes Bjerva, and Isabelle Augenstein. 2018. Copenhagen at CoNLL–SIGMORPHON 2018: Multilingual inflection in context with explicit morphosyntactic decoding. *CoNLL–SIGMORPHON*, page 93.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Daphne Koller and Nir Friedman. 2009. *Probabilistic Graphical Models: Principles and Techniques*. The MIT Press.

John Lafferty, Andrew McCallum, and Fernando CN Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the International Conference on Machine Learning*.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 260–270.

Wang Ling, Chris Dyer, Alan W. Black, Isabel Trancoso, Ramon Fermandez, Silvio Amir, Luis Marujo, and Tiago Luis. 2015. Finding function in form: Compositional character models for open vocabulary word representation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1520–1530.

Tal Linzen, Emmanuel Dupoux, and Yoav Goldberg. 2016. Assessing the ability of LSTMs to learn syntax-sensitive dependencies. *Transactions of the Association for Computational Linguistics*, 4:521–535.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajic, Christopher D Manning, Ryan T McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, et al. 2016. Universal Dependencies v1: A multilingual treebank collection. In *Proceedings of LREC 2016*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318.

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

Jan Ptáček and Zdeněk Žabokrtský. 2006. Synthesis of Czech sentences from tectogrammatical trees. In *International Conference on Text, Speech and Dialogue*, pages 221–228.

Linfeng Song, Xiaochang Peng, Yue Zhang, Zhiguo Wang, and Daniel Gildea. 2017. AMR-to-text generation with synchronous node replacement grammar. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 7–13.

Andrew J. Viterbi. 1967. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions Information Theory*, 13(2):260–269.