Paradigm Completion for Derivational Morphology

Ryan Cotterell[ə] **Ekaterina Vylomova**[fi] **Huda Khayrallah**[ə] **Christo Kirov**[ə] **David Yarowsky**[ə]

[ə]Center for Language and Speech Processing, Johns Hopkins University, USA
[fi]Department of Computing and Information Systems, University of Melbourne, Australia
ryan.cotterell@jhu.edu evylomova@student.unimelb.edu.au
{huda,ckirov,yarowsky}@cs.jhu.edu

Abstract

The generation of complex derived word forms has been an overlooked problem in NLP; we fill this gap by applying neural sequence-to-sequence models to the task. We overview the theoretical motivation for a paradigmatic treatment of derivational morphology, and introduce the task of derivational paradigm completion as a parallel to inflectional paradigm completion. State-of-the-art neural models, adapted from the inflection task, are able to learn a range of derivation patterns, and outperform a non-neural baseline by 16.4%. However, due to semantic, historical, and lexical considerations involved in derivational morphology, future work will be needed to achieve performance parity with inflection-generating systems.

1 Introduction

Unlike inflectional morphology, which produces grammatical variants of the same core lexical item (e.g., *take*↦*takes*), derivational morphology is one of the key processes by which new lemmata are created. For example, the English verb *corrode* can evolve into the noun *corrosion*, the adjective *corrodent*, and numerous other complex derived forms such as *anticorrosive*. Derivational morphology is often highly productive, leading to the ready creation of neologisms such as *Rao-Blackwellize* and *Rao-Blackwellization*, both originating from the Rao-Blackwell theorem. Despite the prevalence of productive derivational morphology, however, there has been little work on its generation. Commonly used derivational resources such as NomBank (Meyers et al., 2004) are still finite. Moreover, the complex phonological and historical changes (e.g., the adjectivization *corrode*↦*corrosive*) and

affix selection (e.g., choosing between English deverbal suffixes *-ment* and *-tion*) make generation of derived forms an interesting and challenging problem for NLP.

In this work, we show that viewing derivational morphological processes as paradigmatic may be fruitful for generation. This means that there are a number of well-defined form-function pairs associated with a core word. For example, a typical English verb may have five forms in its inflectional paradigm, corresponding to its base (*take*), past tense (*took*), past participle (*taken*), progressive (*taking*) and third-person singular (*takes*) forms. These forms are related by a consistent set of relations, such as affixation. Similarly, a verb may have several slots in its derivational paradigm: The form *take* has the agentive nominalization *taker*, and the abilitative adjectivization *takable*. Note there are also consistent patterns associated with each derivational slot, e.g., the *-er* suffix regularly produces the agentive.

Exploiting this paradigmatic characterization of derivational morphology allows us to create a statistical model capable of generating derivationally complex forms. We apply state-of-the-art models for inflection generation, which learn mappings from fixed paradigm slots to derived forms. Empirically, we compare results for two models on the new task of derivational paradigm completion: a neural sequence-to-sequence model and a standard non-neural baseline. Our best neural model for derivation achieves 71.7% accuracy, beating the non-neural baseline by 16.4 points. Nevertheless, we note this is about 25 points lower than the equivalent model on the English inflection task (and even 20 points lower than the model's performance on the harder Finnish inflection generation). These results point to additional complications in derivation that require more elaborate models or data annotation to overcome. While inflection generation is

Semantics	POS	Affix
NEGATION	J→J	*un-, in-, il-, ir-*
ORIGIN	N→J	*-an, -ian, -ish, -ese*
RELATION	N→J	*-ous, -ious, -eous*
DIMINUTIVE	N→N	*-ette*
REPEAT	V→V	*re-*
PATIENT	V→N	*-ee*
RESULT	V→N	*-ment, -ion, -tion, -tion, -al, -ure*
AGENT	V→N	*-er, -or, -ant, -ee*
POTENTIAL	V→J	*-able,-abil, ible*

Table 1: A partial list of derivational transformations in English with corresponding POS changes and semantic labels.

becoming a solved problem (Cotterell et al., 2017), derivation generation is still very much open.

2 Derivational Morphology

The generation of derived forms is structurally similar to the generation of inflectional variants, but presents additional challenges for NLP. Here, we provide linguistic background comparing the two types of morphological processes.

Inflection and Derivation. Inflectional morphology primarily marks semantic features that are necessary for syntax, e.g., gender, tense and aspect. Thus, it follows that in most languages inflection never changes the part of speech of the word and often does not change its basic meaning. The set of inflectional forms for a given lexeme is said to form a paradigm, e.g., the full paradigm for the verb *to take* is ⟨*take*, *taking*, *takes*, *took*, *taken*⟩. Each entry in an inflectional paradigm is termed a slot and is indexed by a syntacto-semantic category, e.g., the PAST form of *take* is *took*. We may reasonably expect that all English verbs—including neologisms— have these five forms.[1] Furthermore, there is typically a fairly regular relationship between a paradigm slot and its form (e.g., add *-s* for the third person singular form). Derivational morphology, on the other hand, often changes the core part of speech of a word and makes more radical changes in meaning. In fact, derivational processes are often subcategorized by the part-of-speech change they engender, e.g., *corrode↦corrosion* is a deverbal nominalization.

Derivational Paradigms. Much like inflection, derivational processes may be organized into

paradigms, with slots corresponding to more abstract lexico-semantic categories for an associated part of speech (Corbin, 1987; Booij, 2008; Štekauer, 2014). Lieber (2004) presents one of the first theoretical frameworks to enumerate a set of derivational paradigm slots, motivated by previous studies of semantic primitives by Wierzbicka (1988). A partial listing of possible derivational paradigm slots for base English adjectives, nouns, and verbs is given in Table 1. The list contains several productive cases. A key difficulty comes from the the fact that the mapping between semantics and suffixes is not always clean; Lieber (2004) points out the category AGENT could be expressed by the suffix *-er* (as in *runner*) or by *-ee* (as in *escapee*). However, both *-er* and *-ee* may have the PATIENT role; consider *burner* ("a cheap phone intended to be disposed of, i.e. burned") and *employee* ("one being employed"), respectively. We flesh out partial derivational paradigms for several English verbs in Table 2.

Unlike in inflectional paradigms, where we expect most cells to be filled for any given base form, derivational paradigms often contain baseslot combinations that are not semantically compatible, leading to the gaps in Table 2.[2] We also observe increased paradigm irregularity due to some derived forms becoming lexicalized at different points in history, differences in the language from which the base word entered the target language (e.g., English roots of Germanic and Latinate origin behave differently (Bauer, 1983)), as well as other factors that are not obvious from the characters in the base word (e.g., gender or number of the resulting noun).

As an example of how difficult these factors can make derivation, consider the wide variety of potential nominalizations corresponding to the RESULT of a verb, e.g., *-ion*, *-al* and *-ment*, (Jackendoff, 1975). While any particular English verb will almost exclusively employ exactly one of these suffixes (e.g., we have *refuse↦refusal* and other candidates *⁎refusion* and *⁎refusement* are illicit),[3] the information required to choose the correct suffix may be both arbitrary or not easily available.

---

[1] Only a handful of English irregulars distinguish between the past tense and the past participle, e.g., *took* and *taken*, and thus have five *unique* forms in their verbal paradigms; most English verbs have four *unique* forms.

[2] For instance, if suffix *-ee* marks a PATIENT it is semantically not compatible with intransitive verbs, i.e., *⁎sneezee* cannot be derived from intransitive *sneeze*.

[3] Note some forms appear to have multiple nominalizations, e.g., *deport↦{deportation,deportment}*, but closer inspection shows there is one regular semantic transformation per word sense: *deportation* is eviction, but *deportment* is behavior.

Base	-er/-or	-ee	-ment/-tion	-able/-ible
POS	V↦N	V↦N	V↦N	V↦J
Semantic	AGENT	PATIENT	RESULT	POTENTIAL
animate	*animator*	—	*animation*	*animatable*
attract	*attractor*	*attractee*	*attraction*	*attractable*
—	*aggressor*	*aggressee*	*aggression*	—
employ	*employer*	*employee*	*employment*	*employable*
place	*placer*	—	*placement*	*placeable*
repel	*repeller*	*repelee*	*repellence*	*repellable*
escape	*escapee*	—	—	*escapable*
corrode	*corroder*	—	*corrosion*	*corrosible*
derive	*deriver*	*derivee*	*derivation*	*derivable*

Table 2: Partial derivational paradigm for several English verbs; semantic gaps are indicated with —.

Productivity. There is a general agreement in linguistics that frequently used complex words become part of the lexicon as wholes, while most others are likely to be constructed from constituents (Bauer, 2001; Aronoff and Lindsay, 2014); the latter ones typically follow derivational patterns, or rules, such as adding *-able* to express potential or ability or applying *-ly* to convert adjectives into adverbs. These patterns typically present two essential properties: productivity and restrictedness. Productivity relates to the ability of a pattern to be applied to any novel base form to create a new word, potentially on-the-fly. One example of such a productive transformation is adding *-less* (privative construction), which may attach to almost any noun to form an adjective. Moreover, the resulting form's meaning is compositional and predictable. Many derivational suffixes in English are of this type. On the other hand, some patterns are subject to semantic, pragmatic, morphological or phonological restrictions. Consider the English patient suffix *-ee*, which cannot be attached to a base ending in /i(:)/, e.g., it cannot be attached to the verb *free* to form *freeee*. Restrictedness is closely related to productivity, i.e., highly productive rules are less restricted. A parsimonious model of derivational morphology would describe forms using productive rules when possible, but may store forms with highly restricted patterns directly as full lexical items.

A Note On Terminology. We would like to make a subtle, but important point regarding terminology: the phrase *morphologically rich* in the NLP community almost exclusively refers to *inflectional*, rather than *derivational* morphology. For example, English is labeled as morphologically impoverished, whereas German and Russian are considered morphologically rich, e.g., see the introduction of Tsarfaty et al. (2010). As regards derivation, English is quite complex and even similar in richness to German or Russian as it contains productive formations from two substrata: Germanic and Latinate. From this perspective, English is very much a morphologically rich language. Indeed, a corpus study on the Brown Corpus showed that the *majority* of English words are morphologically complex when derivation is considered (Light, 1996). Note that there many languages that have exhibit neither rich inflection, nor rich derivational morphology, e.g., Chinese, which most commonly employs compounding for word word formation (Chung et al., 2014).

3 Task and Models

We discuss our two systems for derivational paradigm completion and the results they achieve.

3.1 Data

We experiment on English derivational triples extracted from NomBank (Meyers et al., 2004).[4] Each triple consists of a base form, the semantics of the derivation and a corresponding derived form e.g., ⟨*ameliorate*, RESULT, *amelioration*⟩. Note that in this task we do not predict whether a slot exists, merely what form it would take given the base and the slot. In terms of current study, we consider the following derivational types: verb nominalization such as RESULT, AGENT and PATIENT, adverbalization and adjective-noun transformations. We intentionally avoid zero-derivations. We also

---

[4]There are few resources annotated for derivation in non-English languages, making wider experimentation difficult.

	baseline		1-best seq2seq		10-best seq2seq
	acc	edit	acc	edit	acc
all	55.3%	2.01	71.7%	0.97	84.5%
NOMINAL (J↦N)	23.1%	3.45	35.1%	2.67	70.2%
RESULT (V↦N)	40.0%	2.24	52.9%	1.86	72.6%
AGENT (V↦N)	52.2%	0.94	65.6%	0.78	82.2%
ADVERB (J↦R)	90.0%	0.21	93.3%	0.18	96.5%

Table 3: Results under two metrics (accuracy and Levenshtein distance) comparing the non-neural baseline from the 201 SIG-MORPHON shared task and the neural sequence-to-sequence model (both for 1-best and 10-best output).

exclude overly orthographically distant pairs by filtering out those for which the Levenshtein distance exceeds half the sum of their lengths, which appear to be misannotations in NomBank. The final dataset includes 6,029 derivational samples, which we split into train (70%), development (15%), and test (15%).[5] We also note that NomBank annotations are often semantically more coarse-grained.

3.2 Evaluation Metrics

We evaluate on 3 metrics: accuracy, average edit distance, and F_1. Accuracy measures how often system output exactly matches the gold string. Edit distance, by comparison, measures the Levenshtein distance between system output and the gold string. Finally, we calculate affix F_1 scores for individual derivational affixes. E.g., for *-ment* precision is the number of words where the model correctly predicted *-ment* (out of total predictions) and recall is the number of words where the model correctly predicted out of the number of true words.

3.3 Baseline Transducer

We train a simple transducer for each base-to-paradigm slot mapping in the training set, identical to the baseline described in Cotterell et al. (2016). This uses an averaged perceptron classifier to greedily apply an output transformation (substitution, deletion, or insertion) to each input character given the surrounding characters and previous decisions.

3.4 RNN Encoder-Decoder

Following Kann and Schütze (2016) on the morphological inflection task, we use an encoder-decoder gated recurrent neural network (Bahdanau et al., 2015). First, an encoder network encodes a sequence: the concatenation of the characters of

---

the input word and a tag describing the desired transformation—both represented by embeddings. This encoder is bidirectional and consists of two gated RNNs (Cho et al., 2014), one encoding the input in the forward direction, the other one encoding in the backward direction. The output of the two RNNs is the resulting hidden vectors $\overrightarrow{h_i}$ and $\overleftarrow{h_i}$. The hidden state is a concatenation of the forward and backward hidden vectors, i.e., $h_i = [\overrightarrow{h_i} \overleftarrow{h_i}]$.

The decoder also consists of an RNN, but is additionally equipped with an attention mechanism. The latter computes a weight for each of the encoder hidden vectors for each character or subtag, which can be roughly understood as giving a certain importance to each of the inputs. The probability of the target sequence $y = (y_1, \ldots, y_{|y|})$ given the input sequence $x = (x_1, \ldots, x_{|x|})$ is modeled by

$$p(y \mid x_1, \ldots, x_{|x|}) = \prod_{t=1}^{|y|} p(y_t \mid \{y_1, \ldots, y_{t-1}\}, c_t)$$
$$= \prod_{t=1}^{|y|} g(y_{t-1}, s_t, c_t), \qquad (1)$$

where g is a multi-layer perceptron, s_t is the hidden state of the decoder and c_t is the sum of the encoder states h_i, scored by attention weights $\alpha_i(s_{t-1})$ that depend on the decoder state: $c_t = \sum_i \alpha_i(s_{t-1}) h_i$.

Input Encoding. We model this problem as a character translation problem, with special encodings for the transformation tags that indicate the type of derivation. For example, we treat the triple: ⟨*ameliorate*, RESULT, *amelioration*⟩ as the source string `a m e l i o r a t e RESULT` and target string `a m e l i o r a t i o n`. This is similar to the encoding in Kann and Schütze (2016).

Training. We use the Nematus toolkit (Sennrich et al., 2017).[6] We exactly follow the recipe in Kann and Schütze (2016), the winning submission on the 2016 SIGMORPHON shared task for inflectional morphology. Accordingly, we use a character embedding size of 300, 100 hidden units in both the encoder and decoder, Adadelta (Zeiler, 2012) with a minibatch size of 20, and a beam size of 12. We train for 300 epochs and select the test model based on the performance on the development set.

4 Experimental Results

Table 3 compares the accuracy of our baseline system with the accuracy of our sequence-to-sequence

---

neural network using the data splits discussed in §3.1. In all cases, the network outperforms the baseline. While 1-best performance is not nearly as high as that expected from a state-of-the-art inflectional generation system, the key point is that performance significantly increases when considering the 10-best outputs. This suggests that the network is indeed learning the correct set of possible nominalization patterns. However, the information needed to correctly choose among these patterns for a given input is not necessarily available to the network. In particular, the network is only aware of important disambiguating historical (e.g., is the input of Latin or Greek origin) and lexical-semantic (e.g., is the input verb transitive or intransitive) factors to the extent that they are implicitly encoded in the input character sequence. We speculate that making these additional pieces of information directly available as input features will significantly improve 1-best accuracy.

Unfortunately, NomBank does not provide the necessary annotations in most cases. For instance, there is no way to differentiate *actor* and *actress* without gender. It also does not distinguish the semantics of some adjective nominalizations, e.g., *activism* and *activity*. Future work will reannotate NomBank to make these finer-grained distinctions.

Error Analysis. We observe mistakes on less frequent suffixes, e.g., *-age*—we predict **draination* instead of *drainage*. Also, there are several cases where NomBank only lists one available form, e.g., *complexity*, and our model predicts *complexness*. We also see mistakes on irregular adverbs, e.g., we generate *advancely* from *advance*, rather than *in-advance*, as well as in PATIENT nominalizations, e.g., the model produces *containee* in place of *content*—this last distinction is unpredictable.

5 Related Work

Previous work in unsupervised morphological *segmentation* and has implicitly incorporated derivational morphology. Such systems attempt to segment words into all constituent morphs, treating inflectional and derivational affixes as equivalent. The popular Morfessor tool (Creutz and Lagus, 2007) is one example of such an unsupervised segmentation system, but many others exist, e.g., Poon et al. (2009), Narasimhan et al. (2015) *inter alia*. Supervised segmentation and analysis models in the literature can also break down derivationally complex forms into their morphs, provided pre-

affix	F_1	affix	F_1	affix	F_1
-ly	1.0	-ity	0.54	-ence	0.32
-er	0.86	-ment	0.45	-ure	0.22
-ation	0.78	-ist	0.43	-ee	0.20
-or	0.59	-ness	0.40	-age	0.20

Table 4: F_1 for various suffix attachments with the sequence-to-sequence model

segmented and labeled data is available for training (Ruokolainen et al., 2013; Cotterell et al., 2015; Cotterell and Schütze, 2017). Our work, however, builds directly upon recent efforts in the *generation* of inflectional morphology (Durrett and DeNero, 2013; Nicolai et al., 2015; Ahlberg et al., 2015; Rastogi et al., 2016; Faruqui et al., 2016). We differ in that we focus on derivational morphology. In another recent line of work, Vylomova et al. (2017) predict derivationally complex forms using sentential context. Our work differs from their approach in that we attempt to generate derivational forms divorced from the context, but the underlying neural sequence-to-sequence architecture is quite similar.

6 Conclusion

We have presented a statistical model for the generation of derivationally complex forms, a task that has gone essentially unexplored in the literature. Viewing derivational morphology as paradigmatic, where slots refer to semantic categories, e.g., *corrode*+RESULT↦*corrosion*, we draw upon recent advances in the generation of inflectional morphology. Applying this method works well, achieving an overall accuracy of 71.71%, and beating a non-neural baseline. Performance, however, is lower than on the task of paradigm completion for inflectional morphology, indicating that paradigm completion for derivational morphology is more challenging than its inflectional counterpart.

Acknowledgements

References

Malin Ahlberg, Markus Forsberg, and Mans Hulden. 2015. Paradigm classification in supervised learning of morphology. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*, pages 1024–1029, Denver, Colorado. Association for Computational Linguistics.

Mark Aronoff and Mark Lindsay. 2014. Productivity, blocking and lexicalization. *The Oxford Handbook of Derivational Morphology*, pages 67–83.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *International Conference on Learning Representations (ICLR)*.

Laurie Bauer. 1983. *English Word-formation*. Cambridge University Press.

Laurie Bauer. 2001. *Morphological Productivity*, volume 95. Cambridge University Press.

Geert Booij. 2008. Paradigmatic morphology. *La Raison Morphologique. Hommage à la Mémoire de Danielle Corbin*, pages 29–38.

Kyunghyun Cho, Bart van Merrienboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the properties of neural machine translation: Encoder–decoder approaches. In *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, pages 103–111, Doha, Qatar. Association for Computational Linguistics.

Karen Steffen Chung, Nathan W. Hill, and Sun Jackson T.-S. 2014. Sino-Tibetan. In Rochelle Lieber and Pavol Štekauer, editors, *The Oxford Handbook of Derivational Morphology*, chapter 34, pages 609–650. Oxford University Press, Oxford.

Danielle Corbin. 1987. *Morphologie Dérivationnelle et Structuration du Lexique*, volume 193. Walter de Gruyter.

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, Thomas Müller, Alexander Fraser, and Hinrich Schütze. 2015. Labeled morphological segmentation with semi-Markov models. In *Proceedings of the Nineteenth Conference on Computational Natural Language Learning (CoNLL)*, pages 164–174, Beijing, China. Association for Computational Linguistics.

Ryan Cotterell and Hinrich Schütze. 2017. Joint semantic synthesis and morphological analysis of the derived word. *Transactions of the Association for Computational Linguistics (TACL)*, 5:147–161.

Mathias Creutz and Krista Lagus. 2007. Unsupervised models for morpheme segmentation and morphology learning. *ACM Transactions on Speech and Language Processing (TSLP)*, 4(1):3.

Greg Durrett and John DeNero. 2013. Supervised learning of complete morphological paradigms. In *Proceedings of the 2013 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 2016 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.

Ray Jackendoff. 1975. Morphological and semantic regularities in the lexicon. *Language*, pages 639–671.

Katharina Kann and Hinrich Schütze. 2016. Single-model encoder-decoder with explicit morphological representation for reinflection. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 555–560, Berlin, Germany. Association for Computational Linguistics.

Rochelle Lieber. 2004. *Morphology and Lexical Semantics*, volume 104. Cambridge University Press.

Marc Light. 1996. *Morphological Cues for Lexical Semantics*. Ph.D. thesis, University of Rochester.

A. Meyers, R. Reeves, C. Macleod, R. Szekely, V. Zielinska, B. Young, and R. Grishman. 2004. The nombank project: An interim report. In *HLT-NAACL 2004 Workshop: Frontiers in Corpus Annotation*, pages 24–31, Boston, Massachusetts, USA. Association for Computational Linguistics.

Karthik Narasimhan, Regina Barzilay, and Tommi Jaakkola. 2015. An unsupervised method for uncovering morphological chains. *Transactions of the Association for Computational Linguistics (TACL)*, 3:157–167.

Garrett Nicolai, Colin Cherry, and Grzegorz Kondrak. 2015. Inflection generation as discriminative string transduction. In *Proceedings of the 2015 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.

Hoifung Poon, Colin Cherry, and Kristina Toutanova. 2009. Unsupervised morphological segmentation with log-linear models. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 209–217, Boulder, Colorado. Association for Computational Linguistics.

Pushpendre Rastogi, Ryan Cotterell, and Jason Eisner. 2016. Weighting finite-state transductions with neural context. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*, pages 623–633, San Diego, California. Association for Computational Linguistics.

Teemu Ruokolainen, Oskar Kohonen, Sami Virpioja, and Mikko Kurimo. 2013. Supervised morphological segmentation in a low-resource learning setting using conditional random fields. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning (CoNLL)*, pages 29–37, Sofia, Bulgaria. Association for Computational Linguistics.

Rico Sennrich, Orhan Firat, Kyunghyun Cho, Alexandra Birch, Barry Haddow, Julian Hitschler, Marcin Junczys-Dowmunt, Samuel Läubli, Antonio Valerio Miceli Barone, Jozef Mokry, and Maria Nadejde. 2017. Nematus: a toolkit for neural machine translation. In *Proceedings of the Software Demonstrations of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pages 65–68, Valencia, Spain. Association for Computational Linguistics.

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

Pavol Štekauer. 2014. Derivational paradigms. In Rochelle Lieber and Pavol Štekauer, editors, *The Oxford Handbook of Derivational Morphology*, chapter 12, pages 354–369. Oxford University Press, Oxford.

Ekaterina Vylomova, Ryan Cotterell, Timothy Baldwin, and Trevor Cohn. 2017. Context-aware prediction of derivational word-forms. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 118–124, Valencia, Spain. Association for Computational Linguistics.

Anna Wierzbicka. 1988. *The Semantics of Grammar*, volume 18. John Benjamins Publishing.

Matthew D. Zeiler. 2012. ADADELTA: an adaptive learning rate method. *CoRR*, abs/1212.5701.