Morfessor-enriched features and multilingual training for canonical morphological segmentation

Rouhe, Aku

The Association for Computational Linguistics
2022-06

Rouhe , A , Grönroos , S-A , Virpioja , S , Creutz , M & Kurimo , M 2022 , Morfessor-enriched features and multilingual training for canonical morphological segmentation . in G Nicolai & E Chodroff (eds) , Proceedings of the 19th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology . The Association for Computational Linguistics , Stroudsburg , pp. 144-151 , Workshop on Computational Research in Phonetics, Phonology, and Morphology , Seattle , Washington , United States , 14/07/2022 . https://doi.org/10.18653/v1/2022.sigmorphon-1.16

http://hdl.handle.net/10138/348400
https://doi.org/10.18653/v1/2022.sigmorphon-1.16

Morfessor-enriched features and multilingual training for canonical morphological segmentation

Aku Rouhe[♢] **Stig-Arne Grönroos**[♡♠] **Sami Virpioja**[♡]
Mathias Creutz[♡] **Mikko Kurimo**[♢]
[♢] Department of Signal Processing and Acoustics, Aalto University, Finland
[♡] Department of Digital Humanities, University of Helsinki, Finland
[♠] Silo.AI, Finland
[♡] name.surname@helsinki.fi
[♢] name.surname@aalto.fi

Abstract

In our submission to the SIGMORPHON 2022 Shared Task on Morpheme Segmentation, we study whether an unsupervised morphological segmentation method, Morfessor, can help in a supervised setting. Previous research has shown the effectiveness of the approach in semi-supervised settings with small amounts of labeled data. The current tasks vary in data size: the amount of word-level annotated training data is much larger, but the amount of sentence-level annotated training data remains small. Our approach is to pre-segment the input data for a neural sequence-to-sequence model with the unsupervised method. As the unsupervised method can be trained with raw text data, we use Wikipedia to increase the amount of training data. In addition, we train multilingual models for the sentence-level task. The results for the Morfessor-enriched features are mixed, showing benefit for all three sentence-level tasks but only some of the word-level tasks. The multilingual training yields considerable improvements over the monolingual sentence-level models, but it negates the effect of the enriched features.

1 Introduction

Current use of subword segmentation in neural natural language processing (NLP) with unsupervised segmentation methods such as BPE (Sennrich et al., 2015), SentencePiece (Kudo and Richardson, 2018), and Morfessor (Creutz and Lagus, 2002; Virpioja et al., 2013) mainly focuses on finding short and frequent subwords that give good performance in the NLP application, while putting less weight on linguistic correctness. The level of segmentation varies by the frequency of the word: frequent words retain their affixes, while rare words, such as rare proper names, are heavily segmented into syllable-like units or even characters. These methods typically perform *surface* segmentation, meaning that the subwords can be concatenated back into the surface form of the word without any transformation to account for phonological processes

e.g. $profibrotic \mapsto pro +\!\!+ fibr +\!\!+ ot +\!\!+ ic.$

However, when linguistic fidelity is of importance—for example because the segments are analyzed statistically as opposed to using a neural model—a supervised segmentation method may be more suitable. The goal is to output morphemes, the smallest meaning-bearing linguistic units. In *canonical morphological segmentation* (Kann et al., 2016), instead of segmenting into surface forms of morphemes, the different allomorphs are mapped into a single canonical form, reversing any phonological changes.

e.g. $profibrotic \mapsto pro +\!\!+ fibre +\!\!+ osis +\!\!+ ic.$

It is not always possible to give a single correct analysis for any particular surface form. A surface form may be homonymous, with inflections or derivations from two or more lemmas. In order to disambiguate the meanings to choose a single analysis from several alternatives, it is necessary to use the surrounding sentence context. In Task 2 of this shared task, such sentence level segmentation is performed.

e.g. $she \quad rose \quad up \mapsto she \quad rise +\!\!+ ed \quad up$
$a \quad red \quad rose \mapsto a \quad red \quad rose.$

Word-level morpheme segmentation is more widely studied than sentence-level morpheme segmentation. In part, the focus on word level segmentation is due to the historically limited ability of models to exploit all of the available context. With neural sequence to sequence (seq2seq) models, this limitation can easily be lifted. Limited availability of labeled data for the sentence level task provides

19th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology, pages 144 - 151

a second reason for the popularity of word-level segmentation.

This work presents the AUUH (Aalto University - University of Helsinki) team submission to the SIGMORPHON 2022 Shared Task on Morpheme Segmentation (Batsuren et al., 2022). In this shared task, the imbalance of training data persists. For the word-level Task 1, there is ample training data, ranging from 15 000 labeled words for the lowest resourced language, Mongolian, to hundreds of thousands of words for the higher resourced languages. Task 1 has between 3 and 30 times as much data as in sentence-level Task 2. In addition to the labeled data, an order of magnitude more unlabeled data can easily be sourced.

Considering that these types of data are available in very different amounts, there is an opportunity to improve especially the sentence-level performance by exploiting the other types of data. In this work, we use large amounts of unlabeled data to enrich the input with features from an unsupervised segmentation model. This feature set augmentation approach, which combines the strengths of generative and discriminative models, has previously been applied for word-level surface segmentation (Ruokolainen et al., 2014; Grönroos et al., 2019). Additionally, we use the word-level labeled data through multi-task and multi-lingual training.

Our systems are fully data-driven and language-independent, requiring no linguistic resources beyond the training data. All the software used in the systems has open-source implementations.

2 Methods

Our approach for the shared tasks consists of a neural seq2seq model, enrichment of data with features learned in an unsupervised manner, and multi-task and multilingual training. We submitted six different configurations, which we refer to as Systems A–F in the following.

2.1 Seq2seq model

We apply a sequence-to-sequence (seq2seq) model to map from character sequences to character sequences. In our baseline models, the input is the character sequence of the surface form of the word. In our enriched models, the surface form is augmented with predicted segmentation boundary symbols. In all cases, the output is the sequence of canonical morphemes and segmentation boundary symbols, decoded on character level. We treat the boundary marker "@@" as a single symbol[1]. In the original output format, the morphemes are separated by a space, which we simply ignore in the seq2seq data and add back in the detokenization step. Our seq2seq models are implemented using the Marian NMT (Junczys-Dowmunt et al., 2018) Neural Machine Translation framework.

Even though the amount of data is of a standard size for segmentation, it is small compared to typical machine translation data sets. Therefore, when designing the neural network architectures, we experiment with neural architectures from the literature on low-resource neural machine translation.

Following Sennrich and Zhang (2019), our models C–F use a bidirectional GRU bideep (Miceli Barone et al., 2017) architecture. We modify the architecture slightly by lowering the embedding dimension from 512 to 128, as we have a character-level model instead of a subword model.

Inspired by Araabi and Monz (2020), we try reducing the capacity of Transformer-base (Vaswani et al., 2017) to better suit the small data setting, reducing the number of layers in both encoder and decoder to 5, reducing the feed-forward dimension to 512, reducing the number of attention heads to 2, increasing dropout to 0.3, adding 0.1 target dropout (and in our implementation 0.1 source dropout as well), and increasing label smoothing to 0.5. However, in preliminary experiments this performed worse than Transformer-base. Instead, a smaller Transformer-base modification, which we title Transformer-base$_{mod}$, where we reduce the feed-forward dimension to 1024, and add 0.1 source and target dropout, yields our best Transformer results in preliminary experiments.

For the monolingual word-level tasks we use the bideep GRU architecture, as that architecture worked reliably even with limited data. For the multi-task, multi-lingual models A–B, which are trained with considerably more data overall, we use the Transformer-base$_{mod}$ architecture.

The seq2seq models are trained for 50 epochs with the cross-entropy loss, with early stopping based on validation criterion improvement stalling. As a validation criterion, we use the official evaluation F-measure. This choice yielded consistent improvements over the cross-entropy criterion in preliminary experiments.

[1]For clarity, represented later in the paper as a single symbol @.

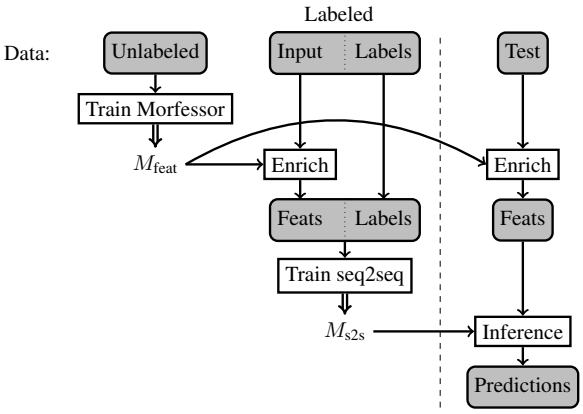

Figure 1: Feature enrichment process.

2.2 Enrichment with unsupervised features

The feature enrichment process is shown in Figure 1. For training the unsupervised features, the training data consists of a large word list extracted from an unlabeled corpus. Morfessor Baseline (Creutz and Lagus, 2002; Virpioja et al., 2013), an unsupervised generative model, is trained using the unlabeled data only.

The words in the labeled training set are first presegmented using the Morfessor Baseline model. The predicted segmentation is turned into features by adding a reserved unicode character at the predicted segmentation boundaries, and then concatenating to form the new input string.

For example, the input string *"subneural"* is segmented by Morfessor as

$$subneural \mapsto s\ u\ b\ \sqcup\ n\ e\ u\ r\ a\ l.$$

The seq2seq model then takes this feature representation as input, and outputs the canonical segmentation:

$$s\ u\ b \sqcup n\ e\ u\ r\ a\ l \mapsto s\ u\ b\ @\ n\ e\ u\ r\ o\ n\ @\ a\ l.$$

At decoding time a two-step procedure is used: first the features for the desired words are produced using the Morfessor Baseline model. The final segmentation can then be decoded from the seq2seq model.

The idea is that the features from the unsupervised generative model allow the statistical patterns found in the large unannotated data to be exploited. Two tasks remain for the seq2seq model to learn: determining when the predictions of Morfessor are reliable in order to correct its mistakes, and finding the mapping from predicted surface morphemes to the canonical forms of morphemes. We hypothesize that these two tasks are easier to learn as part of a pipeline system, compared to learning the mapping from the unsegmented surface form into canonical morphemes directly as an end-to-end task.

2.2.1 Morfessor

Morfessor is a family of language-independent unsupervised and semi-supervised morpheme segmentation models. The first variant, later called Morfessor Baseline, was introduced by Creutz and Lagus (2002). It is an unsupervised algorithm that makes use of a context-insensitive maximization criterion based on unigram probabilities. A Python implementation and extensions were provided by Virpioja et al. (2013) with further improvements by Grönroos et al. (2020). Further unsupervised variants introduce context-sensitive segmentation, identifying possible prefixes, stems and suffixes as a biproduct. The so-called Morfessor Categories-MAP model (Creutz and Lagus, 2005, 2007) produces a hierarchical segmentation structure, which later evolved into a flat structure in Morfessor FlatCat (Grönroos et al., 2014). Kohonen et al. (2010) extended to semi-supervised learning for situations where small amounts of linguistic gold standard analyses are available.

In this work, we focus on using Morfessor Baseline, leaving comparison of different Morfessor variants for future work.

2.2.2 Training data

For training the Morfessor models, we use the official word-level training sets, sentence-level training sets for the languages that had them available, and, in addition, Wikipedia dumps from 2022-04-01. The word-level data is added as is. From the sentence-level data, we include tokens that contained only letters in a script suitable for the language (Cyrillic for Mongolian, and Latin for English and Czech). Wikipedia dumps are processed with `wikiextractor` (Attardi, 2015). Only those tokens that have the correct script (Cyrillic for Mongolian and Russian, Latin for the rest) are included. In addition, to further reduce non-words and foreign words, we restrict word length to 40, word frequency to 3 for English and 2 for the rest, and either include only lowercase words (English) or lowercase the words (rest).

Finally, the words from the different sources are combined together for training Morfessor. The

	Wikipedia	Task 1	Task 2	total
labels	unlabeled	word-level	sentence-level	
ces	1097041	30694	4890	1107515
eng	466490	458692	15700	779878
fra	1502818	252671	0	1649688
hun	1356328	742239	0	1937213
ita	1171105	369208	0	1417499
lat	224277	705862	0	914135
mon	101136	15171	4961	108668
rus	2148379	627367	0	2483749
spa	1402977	688672	0	1942361

Table 1: Numbers of unique word forms in the training data sets.

frequencies of the words are ignored in training. Table 1 shows the numbers of unique word forms in the data sets.

We observe that with the exception of the Czech language, all subtasks of this shared task consist of canonical segmentation. For some words, the label sequence concatenates directly into the surface form, i.e. the canonicalization mapping of each morpheme is the identity function. The proportion of training words having this property vary by language, from 7.6% for Italian to 99.7% for Latin. However, for the Czech language, all the words in the training data have this property of concatenating directly into the surface form. As the Czech language does exhibit allomorphy (see e.g. Ševčíková, 2018), we conclude that the task for Czech was surface segmentation rather than canonical segmentation.

2.2.3 Hyper-parameter tuning

We use grid search to find the optimal corpus weight hyper-parameter for the Morfessor models. We test values in the range from 0.001 to 2.0. The word-level development sets are used for evaluation. However, the official evaluation scripts expect canonical segmentation, while Morfessor produces surface segmentation. Thus we rely on the EMMA-2 evaluation method and maximize the F_1-score between the model and reference segmentations.[2] EMMA-2, proposed by Virpioja et al. (2011), is a variant of the EMMA (Evaluation Metric for Morphological Analysis) introduced by Spiegler and Monson (2010). Both methods solve the problem of comparison of two different label sets by creating a mapping between the predicted and reference labels. The original EMMA method finds one-to-one assignment between the labels using the Hungarian algorithm, but the computational complexity prevents using it for large test sets. In contrast, EMMA-2 makes separate one-to-many assignments when calculating the precision and recall.

[2]Implementation available at https://github.com/svirpioj/morphometrics.

2.3 Multi-task and multilingual training

We train models that use two types of multi-task objectives. In the first one, we combine the word-level Task 1 with the sentence-level Task 2. In the second one, we train a multilingual model with the concatenation of all languages available in Task 2.

To distinguish tasks from each other, we use task selector tokens prefixed to the input, similar to Johnson et al. (2017). The language selector token is first, if used, and then in word tasks a special token is used. Sentence tasks do not have a separate selector token: no selector token implies a sentence task.

The multilingual model is then finetuned for an additional 50 epochs on each individual language. In a preliminary experiment, the additional training time did not by itself yield a better model. In finetuning, the sentence-level and word-level multi-task objective was kept. We finetuned models separately with word- and sentence-level validation data.

2.4 Systems

Table 2 lists the differences between the systems.

In the official competition, some of our submitted systems were trained on slightly different data than we intended, due to human error, and some

	Morfessor features	Architecture	Multilingual	Multitask
System A	✓	Transformer-base$_{mod}$	✓	✓
System B	–	Transformer-base$_{mod}$	✓	✓
System C	✓	Bideep GRU	–	✓
System D	–	Bideep GRU	–	✓
System E	✓	Bideep GRU	–	–
System F	–	Bideep GRU	–	–

Table 2: Differences between the six submitted systems.

systems were missing simply due to running out of time. The results in this description paper have been produced with corrected systems. The results that changed, or were added after the competition deadline, are marked with the symbol $\star$ in the tables.

3 Results

Tables 3 and 4 list the results of Tasks 1 and 2 respectively. Systems A and B, C and D, and E and F each form comparable pairs, where the former (e.g. System A) uses Morfessor-enriched features, and the latter (e.g. System B) is the same system without enriched features. In the result tables, these comparable pairs are separated with horizontal divider lines.

Some of our systems have the highest score of all shared task participants in specific subcategories of the evaluation. Our system B has the highest F_1-score (96.31%) and lowest Levenshtein distance (1.39) for the English sentence-level task. Our system A has the highest F_1-score (93.23%) for the English word-level evaluation category 001, i.e. compound words without inflectional or derivational affixes.

Tables 5 and 6 show Task 1 results by morphological category, for systems A–B and E–F respectively. For English, Russian, and Hungarian, the system using the Morfessor-enriched features performs better for most categories involving compounding, in particular the 001 category (only compounding). Of the languages in this shared task, only Hungarian and English vocabularies contain a substantial portion of compound words (17.32% and 6.79% respectively).

4 Discussion

The multilingual model without Morfessor-enriched features (System B) gives the best results in both tasks for the three languages (ces, eng, mon) for which we trained such a system. When using multilingual training, the Morfessor-enriched features are not beneficial. The unsupervised features may be less useful with the increased amount of training data in the multilingual setup, and varying granularities of the unsupervised segmentations for the different languages could confuse the multilingual model.

Without multilingual training, the results for enriched features are inconclusive for the word-level task, but clearly beneficial for the sentence-level task. The enriched features give better results for 5 languages (ces, eng, rus, mon, hun) in Task 1 and all three languages (ces, eng, mon) in Task 2.

Consistent with previous work (Grönroos et al., 2019), we find that Morfessor-features are useful for modeling the boundary between compound parts, which is challenging for supervised discriminative models on their own.

Except for the corpus weight hyper-parameter of the Morfessor model, we did not tune many parameters of the setup, such as thresholds for the words in the Wikipedia dumps, different weightings for the corpora, or use of the word frequencies in Morfessor training. More extensive optimization could lead to some improvements for the unsupervised features. It would also be possible to use the part of the data, for which the canonical morphemes correspond to surface morphemes as annotations for training semi-supervised Morfessor variants (Kohonen et al., 2010).

It is possible that using a different β for the F_β-score may result in better tuning. Finding the optimal value for β is left for future work. While computationally more burdensome, instead of searching for the best F_β-score of EMMA-2 for Morfessor's output, some parameters could also be optimized on the results of the final seq2seq model.

	ces	eng	fra	ita	lat	rus	mon	hun	spa
System A†	93.65	92.32	-	-	-	-	98.19	-	-
System B	⋆**93.68**	⋆**93.24**	-	-	-	-	⋆**98.29**	-	-
System E†	**90.71**	**87.10**	90.78	92.39	98.71	**94.33**	**96.06**	⋆**98.36**	⋆96.22
System F	90.28	86.40	**90.81**	**92.56**	**98.85**	93.68	95.32	98.34	**97.25**

Table 3: Word-level (Task 1) results (F1-measure [%]) on the official test sets. Results marked with ⋆ were not submitted to the official competition. Systems marked with † use Morfessor features.

	ces	eng	mon
System A†	88.60	96.22	82.19
System B	**90.42**	**96.31**	**82.59**
System C†	⋆**59.77**	⋆**93.44**	⋆**74.08**
System D	⋆59.08	88.07	⋆71.82
System E†	**61.92**	**85.04**	**72.67**
System F	51.47	82.34	66.38

Table 4: Sentence-level (Task 2) results (F1-measure [%]) on the official test sets. Results marked with ⋆ were not submitted to the official competition. Systems marked with † use Morfessor features.

5 Conclusions

We find that Morfessor-enriched features are beneficial for the sentence-level tasks, but see mixed results for the word-level tasks. The multilingual training yields considerable improvements for both tasks, but it negates the effect of the enriched features.

Acknowledgments

This work was funded by the European Research Council (ERC) under the EU's Horizon 2020 research and innovation program (FoTran project, agreement № 771113) and the Academy of Finland grant 345790 in ICT 2023 programme's project "Understanding speech and scene with ears and eyes". We also thank the CSC-IT Center for Science Ltd., for computational resources.

Category	Inflection	Derivation	Compounding	System	eng	mon
000	–	–	–	A†	80.39	87.16
000	–	–	–	B	⋆**82.63**	⋆**87.76**
001	–	–	✓	A†	**93.23**	100.00
001	–	–	✓	B	⋆93.02	⋆100.00
010	–	✓	–	A†	93.35	91.39
010	–	✓	–	B	⋆**93.86**	⋆**91.46**
011	–	✓	✓	A†	**95.60**	–
011	–	✓	✓	B	⋆94.98	–
100	✓	–	–	A†	89.27	99.35
100	✓	–	–	B	⋆**89.97**	⋆**99.69**
101	✓	–	✓	A†	94.03	100.00
101	✓	–	✓	B	⋆**95.76**	⋆100.00
110	✓	✓	–	A†	95.51	**99.56**
110	✓	✓	–	B	⋆**96.91**	⋆99.50
111	✓	✓	✓	A†	92.51	–
111	✓	✓	✓	B	⋆**94.33**	–

Table 5: Task 1 results for Systems A and B by morphological category (subsets of words containing inflection, derivation, compounding, or combinations of these). System A marked with † uses Morfessor features.

References

Ali Araabi and Christof Monz. 2020. Optimizing transformer for low-resource neural machine translation. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3429–3435.

Giuseppe Attardi. 2015. Wikiextractor. https://github.com/attardi/wikiextractor.

Khuyagbaatar Batsuren, Gábor Bella, Aryaman Arora, Viktor Martinović, Kyle Gorman, Zdeněk Žabokrtský, Amarsanaa Ganbold, Šárka Dohnalová, Magda Ševčíková, Kateřina Pelegrinová, Fausto Giunchiglia, Ryan Cotterell, and Ekaterina Vylomova. 2022. The sigmorphon 2022 shared task on morpheme segmentation. In *19th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*.

Mathias Creutz and Krista Lagus. 2002. Unsupervised discovery of morphemes. In *Proceedings of the ACL-02 Workshop on Morphological and Phonological Learning (MPL)*, volume 6, pages 21–30, Philadelphia, Pennsylvania, USA. Association for Computa-

Category	Inflection	Derivation	Compounding	System	eng	fra	ita	rus	mon	hun	spa
000	–	–	–	E†	**76.34**	65.96	63.35	**63.38**	83.78	⋆**98.29**	⋆56.95
000	–	–	–	F	72.59	**66.32**	**64.34**	59.85	**87.76**	81.41	**63.69**
001	–	–	✓	E†	**90.85**	74.40	39.53	**65.35**	100.00	⋆**80.25**	⋆**17.14**
001	–	–	✓	F	89.61	**78.01**	**41.38**	57.73	100.00	80.09	14.95
010	–	✓	–	E†	**87.56**	**78.88**	84.11	80.88	**87.63**	⋆**93.21**	⋆67.10
010	–	✓	–	F	87.07	78.43	**84.75**	**80.96**	84.02	92.62	**75.87**
011	–	✓	✓	E†	**92.50**	**76.60**	50.67	**83.33**	–	⋆**86.52**	⋆41.38
011	–	✓	✓	F	90.79	73.43	**54.55**	78.48	–	85.55	**43.75**
100	✓	–	–	E†	84.48	91.21	**90.70**	**93.75**	**98.62**	⋆97.83	⋆96.52
100	✓	–	–	F	**84.91**	**91.22**	90.28	93.02	97.87	**97.87**	**97.12**
101	✓	–	✓	E†	**95.09**	77.46	59.04	**80.31**	100.00	⋆98.39	⋆44.44
101	✓	–	✓	F	91.03	**79.30**	**66.67**	78.86	100.00	**98.47**	**83.95**
110	✓	✓	–	E†	**89.37**	96.05	95.82	**95.83**	**97.09**	⋆99.29	⋆97.71
110	✓	✓	–	F	89.07	**96.25**	**96.22**	95.15	96.57	**99.30**	**98.64**
111	✓	✓	✓	E†	89.72	**89.89**	**72.94**	83.07	–	⋆**99.09**	⋆88.20
111	✓	✓	✓	F	**85.77**	85.55	67.47	**84.14**	–	98.81	**89.57**

Table 6: Task 1 results for Systems E and F by morphological category (subsets of words containing inflection, derivation, compounding, or combinations of these). System E marked with † uses Morfessor features. Results marked with ⋆ were not submitted to the official competition.

tional Linguistics.

Mathias Creutz and Krista Lagus. 2005. Inducing the morphological lexicon of a natural language from unannotated text. In *Proceedings of the International and Interdisciplinary Conference on Adaptive Knowledge Representation and Reasoning (AKRR'05)*, pages 106–113, Espoo, Finland.

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

Stig-Arne Grönroos, Sami Virpioja, and Mikko Kurimo. 2019. North Sámi morphological segmentation with low-resource semi-supervised sequence labeling. In *Proceedings of the Fifth International Workshop on Computational Linguistics for Uralic Languages*, pages 15–26.

Stig-Arne Grönroos, Sami Virpioja, and Mikko Kurimo. 2020. Morfessor EM+Prune: Improved subword segmentation with expectation maximization and pruning. In *Proceedings of the 12th Language Resources and Evaluation Conference*, Marseilles, France. ELRA.

Stig-Arne Grönroos, Sami Virpioja, Peter Smit, and Mikko Kurimo. 2014. Morfessor FlatCat: An HMM-based method for unsupervised and semi-supervised learning of morphology. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics*, pages 1177–1185, Dublin, Ireland. Association for Computational Linguistics.

Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2017. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351.

Marcin Junczys-Dowmunt, Roman Grundkiewicz, Tomasz Dwojak, Hieu Hoang, Kenneth Heafield, Tom Neckermann, Frank Seide, Ulrich Germann, Alham Fikri Aji, Nikolay Bogoychev, André F. T. Martins, and Alexandra Birch. 2018. Marian: Fast neural machine translation in C++. In *Proceedings of ACL 2018, System Demonstrations*, pages 116–121, Melbourne, Australia. Association for Computational Linguistics.

Katharina Kann, Ryan Cotterell, and Hinrich Schütze. 2016. Neural morphological analysis: Encoding-decoding canonical segments. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 961–967. Association for Computational Linguistics.

Oskar Kohonen, Sami Virpioja, and Krista Lagus. 2010. Semi-supervised learning of concatenative morphology. In *Proceedings of the 11th Meeting of the ACL Special Interest Group on Computational Morphology and Phonology*, SIGMORPHON '10, page 78–86, USA. Association for Computational Linguistics.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Antonio Valerio Miceli Barone, Jindrich Helcl, Rico Sennrich, Barry Haddow, and Alexandra Birch. 2017. Deep architectures for neural machine translation. In *Second Conference on Machine Translation*, pages 99–107. Association for Computational Linguistics.

Teemu Ruokolainen, Oskar Kohonen, Sami Virpioja, and Mikko Kurimo. 2014. Painless semi-supervised morphological segmentation using conditional random fields. In *European Chapter of the Association for Computational Linguistics (EACL)*, pages 84–89, Gothenburg, Sweden. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015. Neural machine translation of rare words with subword units. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

Rico Sennrich and Biao Zhang. 2019. Revisiting low-resource neural machine translation: A case study. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 211–221.

Magda Ševčíková. 2018. Modelling morphographemic alternations in derivation of Czech. *The Prague Bulletin of Mathematical Linguistics*, 110(1):7–42.

Sebastian Spiegler and Christian Monson. 2010. EMMA: A novel evaluation metric for morphological analysis. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 1029–1037, Beijing, China. Coling 2010 Organizing Committee.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 6000–6010.

Sami Virpioja, Peter Smit, Stig-Arne Grönroos, and Mikko Kurimo. 2013. Morfessor 2.0: Python implementation and extensions for Morfessor Baseline. Report 25/2013 in Aalto University publication series SCIENCE + TECHNOLOGY, Department of Signal Processing and Acoustics, Aalto University, Helsinki, Finland.

Sami Virpioja, Ville T Turunen, Sebastian Spiegler, Oskar Kohonen, and Mikko Kurimo. 2011. Empirical comparison of evaluation methods for unsupervised learning of morphology. *Traitement Automatique des Langues*, 52(2):45–90.