The University of Helsinki submissions to the WMT19 similar language translation task

Yves Scherrer, Rául Vázquez, Sami Virpioja

University of Helsinki
{name.surname}@helsinki.fi

Abstract

This paper describes the University of Helsinki Language Technology group's participation in the WMT 2019 similar language translation task. We trained neural machine translation models for the language pairs Czech ↔ Polish and Spanish ↔ Portuguese. Our experiments focused on different subword segmentation methods, and in particular on the comparison of a cognate-aware segmentation method, Cognate Morfessor, with character segmentation and unsupervised segmentation methods for which the data from different languages were simply concatenated. We did not observe major benefits from cognate-aware segmentation methods, but further research may be needed to explore larger parts of the parameter space. Character-level models proved to be competitive for translation between Spanish and Portuguese, but they are slower in training and decoding.

1 Introduction

Machine translation between closely related languages is, in principle, less challenging than translation between distantly related ones. Sharing large parts of their grammars and vocabularies reduces the amount of effort needed for a machine translation system to be able to generalize (Pourdamghani and Knight, 2017). Nevertheless, and especially since the languages offered in this shared task are to some extent morphologically complex, we assume that proper subword segmentation will be beneficial for neural machine translation (NMT) performance. In particular, we aim at consistent segmentation across both related languages. While generic subword segmentation methods such as BPE (Sennrich et al., 2016), Morfessor (Creutz and Lagus, 2007; Grönroos et al., 2014), or SentencePiece (Kudo and Richardson, 2018) yield improved consistency by concatenating data from the two languages and training a single segmentation model, the Cognate Morfessor method (Grönroos et al., 2018) explicitly relies on cognate word pairs to enforce consistent segmentation.

The University of Helsinki participated in the similar language translation task for the language pairs Czech ↔ Polish and Spanish ↔ Portuguese, obtaining the following rankings:
– third (out of six) on Portuguese → Spanish,
– fourth (out of five) on Spanish → Portuguese,
– third (out of five) on Czech → Polish,
– first (out of two) on Polish → Czech.

Section 2 describes the different subword segmentation techniques we considered in our work. Section 3 details the training data and our preprocessing pipeline, whereas Section 4 presents the models we evaluated and the models we submitted, together with the results.

2 Subword segmentation

Our experiments focused on four subword segmentation methods, which are summarized shortly in this section.

2.1 Character segmentation

For similar languages, a commonly used segmentation scheme is character-level segmentation, where every character, including the space character, is considered independently. The idea of character-level machine translation for similar languages dates back to SMT times (e.g. Tiedemann, 2009). More recently, character-level NMT has shown promising results for distant languages (Costa-jussà and Fonollosa, 2016; Lee et al., 2017) as well as for similar ones (Costa-jussà et al., 2017).

The advantage of character-level models is that they do not require any other type of preprocessing such as tokenization or truecasing, and that

Proceedings of the Fourth Conference on Machine Translation (WMT), Volume 3: Shared Task Papers (Day 2) pages 236–244
Florence, Italy, August 1-2, 2019.

the segmentation algorithm is free of hyperparameters. However, character-level NMT models tend to be slow due to the greater length of the sequences.

2.2 Morfessor

Morfessor (Creutz and Lagus, 2002, 2007) is a method for unsupervised morphological segmentation. In contrast to the byte-pair encoding (BPE) algorithm widely adopted in neural machine translation (Sennrich et al., 2016), Morfessor defines a proper statistical model and applies maximum a posteriori estimation for the model parameters. The granularity of the segmentation (and thus size of the subword lexicon) is tunable by inserting a hyperparameter for varying the balance between prior and data likelihood (Kohonen et al., 2010). The prior can be considered as a encoding cost for the subword lexicon, and the likelihood as encoding cost for the corpus given the lexicon. In the first Morfessor variant, Morfessor Baseline (Creutz and Lagus, 2002; Virpioja et al., 2013), the statistical model is a unigram language model, i.e., the subword units are assumed to occur independently in words. Under this assumption, the probability of a sequence of tokens is simplified to be the product of the subword occurrence probabilities, which enables an efficient training algorithm.

The Morfessor Baseline method has been widely tested in automatic speech recognition (ASR) for various languages (Kurimo et al., 2006; Creutz et al., 2007). Smit et al. (2017) report that it performs slightly better in Finnish ASR compared to BPE. Morfessor Baseline and BPE segmentations have not been compared so far with respect to the performance in NMT. However, the Morfessor FlatCat variant (Grönroos et al., 2014) have been tested in English-to-Finnish NMT (Grönroos et al., 2017) and Turkish-to-English NMT (Ataman et al., 2017). While the former does not provide comparison to other segmentation methods, Ataman et al. (2017) report significant improvements over BPE segmentation for Turkish.

2.3 Cognate Morfessor

Cognate Morfessor (Grönroos et al., 2018) is a variant of Morfessor designed to optimize subword segmentation for two related languages so that segmentations are consistent especially for cognates, i.e., word pairs that are similar orthographically, semantically, and distributionally. Cognate Morfessor extends the cost function of Morfessor Baseline (consisting of a lexicon and corpus coding costs) by three lexicon and corpus costs: one for each language, and one for edit operations that transform the cognate forms between the languages. Having more components in the cost function means that they can also be weighted separately; the method has one hyper-parameter for the monolingual corpus costs and one for the edit operations.

The goal of Grönroos et al. (2018) was to improve the translation accuracy from a language with less parallel data (e.g. Estonian) using a related language with more data (e.g. Finnish) in the same NMT system. However, Cognate Morfessor is also a sensible segmentation approach for translating between two related languages. For cognates for which the task is similar to transliteration, the method can learn longer subword chunks that can be transliterated in one step, reducing the average number of tokens per word and improving efficiency compared to character-based models.

Moreover, it can improve the consistency of the segmentation compared to the common approach of concatenating the bilingual corpora and optimizing a joint subword lexicon for them. For example, consider that some common inflection produces a slightly different suffix for the two languages. A joint lexicon is likely to have both suffixes as subword units. Then the suffix for language A may interfere with the segmentation of stems of language B that happen to contain the same string, and vice versa. Cognate Morfessor can avoid such problems by keeping the suffixes in separate lexicons.

2.4 SentencePiece unigram model

As discussed in Section 2.2, Morfessor Baseline defines a unigram language model and determines the size of its lexicon by using a prior probability for the lexicon parameters. A more straightforward approach, first proposed by Varjokallio et al. (2013) for application in ASR, is to fix the lexicon size beforehand and try to find the set of units such that they maximize likelihood of the data for a unigram model. Another heuristic search algorithm for this problem has been proposed by Kudo (2018). In addition, he proposes a subword regularization method for NMT: The unigram language model can be used to generate multiple candidate segmentations to emulate noise and segmentation errors in the data, and thus improve the

Dataset	ES ↔ PT	CS ↔ PL
Europarl	1798 k	619 k
JRC-Acquis	1650 k	1311 k
Wikititles	621 k	249 k
News-Commentary	47 k	—
Total	4116 k	2178 k

Table 1: Filtered parallel dataset statistics (sentence pairs).

Direction	Back-trans.	Parallel	Total
PT → ES	3405 k	4116 k	7520 k
ES → PT	2283 k	4116 k	6399 k
PL → CS	765 k	2178 k	2943 k
CS → PL	4273 k	2178 k	6451 k

Table 2: Back-translation and training data statistics (sentence pairs).

robustness of the translation. The unigram method by Kudo (2018) is implemented in the SentencePiece software (Kudo and Richardson, 2018).

2.5 Byte pair encoding

In Sennrich et al. (2016) the authors adapt the byte pair encoding (BPE) data compression algorithm (Gage, 1994) to the task of word segmentation. They use the idea of the original algorithm, iteratively replacing the most frequent pair of bytes in a sequence with a single and unused byte, on word segmentation by merging characters instead of bytes. This allows for the representation of an open vocabulary through a fixed-size vocabulary of variable-length character sequences.

3 Data

The organizers of the similar languages task provided a fixed set of parallel datasets for training. We filtered these datasets minimalistically, removing empty lines, lines with more than 500 tokens, and lines with source-target length ratio higher than 9.[1] Table 1 reports the sizes of these datasets after filtering.

We trained four character-level NMT systems (see Section 4.1) with these parallel data in order to create back-translations.[2] We created back-translations from all provided monolingual datasets, starting from the beginning of each dataset. Table 2 lists the amount of back-translated sentence pairs per translation direction and summarizes the amount of training data for the final systems.

For the models based on Morfessor and Cognate Morfessor, all data was normalized, tokenized and truecased with the Moses tools[3], while the models based on SentencePiece were only truecased in the same way. For the character-level models, a second filtering step was applied to remove sentence pairs with less than 20 or more than 1000 characters.

The development and test sets were processed analogously, and the system outputs were detokenized and detruecased with the Moses tools.

4 Experiments and results

All our NMT models are trained with the same translation toolkit – OpenNMT-py (Klein et al., 2017) –, use the same model architecture – the Transformer (Vaswani et al., 2017) –, and the same hyperparameters[4]. Training data are shuffled beforehand.

We set a threshold in terms of epochs for each translation direction, after which we stop model training.[5] This allows us to compare models fairly, as they have all seen the same amount of training data, which is not guaranteed when relying on training time or number of batches.

Results on the development set are shown in Table 3 and discussed in detail below. We report two word-level metrics, BLEU (Papineni et al., 2002) and TER (Snover et al., 2006), as well as two character-level metrics, CharacTer (Wang et al., 2016) and chrF (Popović, 2016). BLEU and chrF are computed with SacreBLEU (Post, 2018).[6] In order to quantify the impact of pre- and post-processing, we compute BLEU scores with the unprocessed reference as well as with an additional reference that has been normalized,

[1] We used the `clean-corpus-n.perl` script of the Moses SMT distribution. See https://github.com/moses-smt/mosesdecoder/

[2] We chose character-level systems for back-translation in order not to impose any prior decision on preprocessing and segmentation.

[3] https://github.com/moses-smt/mosesdecoder/

[4] http://opennmt.net/OpenNMT-py/FAQ.html

[5] Note however that not all character-level models could be trained sufficiently long due to timing constraints.

[6] Signatures: *BLEU+case.mixed+numrefs.1+smooth.exp +tok.13a+version.1.2.12*; *chrF2+case.mixed+numchars.6 +numrefs.1+space.False+tok.13a+version.1.2.12*

tokenized, truecased and de-truecased and detokenized. Surprisingly, the results with the two references may vary by up to 2 points.

Despite the large amounts of available training data, we chose hyperparameters resulting in rather small vocabulary sizes for all subword splitting schemes, ranging between 2800 and 8900 units per language pair. This choice was guided by three reasons: (1) the competitive performance of character-level models, (2) the desire to force the models to split words across languages, and to do so not only for rare words, and (3) the competitive performance of small vocabulary sizes in related problems such as historical text normalization (Tang et al., 2018).

A general finding, shared by the other participants, is that the scores on the Slavic language pair are much lower than on the Romance language pair. We assume that the Spanish–Portuguese development and test sets are built by translating directly from one language to the other, whereas the Czech–Polish development and test sets had been translated from English independently of each other, leading to much freer translations. If this hypothesis is correct, the automatic evaluation scores for Czech–Polish may in fact underestimate the real translation quality.

4.1 Character-level models

For each translation direction, we train a character-level model on the parallel data only and use this model to create back-translations for the opposing direction. Table 3 show BLEU scores on the development set under the *Characters-Initial* line.

Additional character-level models are trained with included back-translations. Due to their good overall performance, these models were selected as contrastive runs for our submissions . They are referred to as *Characters* in Table 3.

The comparison of development scores shows the impact of back-translations: depending on the translation direction, gains of 2 to 6 BLEU points are observed. There is however no clear correlation between the amount (or proportion) of added back-translations and the scores.

4.2 Morfessor Baseline models

Morfessor Baseline segmentations were trained on the concatenation of the source and language parallel training data using the Morfessor 2.0 software (Virpioja et al., 2013). We used the default parameters[7] except that we applied log-dampening and a minimum frequency threshold of 5. We selected two corpus weight (α) values, 0.03 and 0.05, for our experiments. Models trained on the latter setting were submitted as contrastive runs.

Results are shown in Table 3. All Morfessor models outperform the character-level models on the processed reference, but not necessarily on the raw reference, suggesting that some normalization and tokenization settings might have been harmful. Unfortunately, we became aware of this issue only after submission.

The differences between the two corpus cost settings are marginal – in general, translation quality slightly improves for one direction but decreases for the other one.

4.3 Cognate Morfessor models

The Cognate Morfessor training method requires cognate word pairs as input. We follow the cognate extraction method presented in Grönroos et al. (2018) with some minor modifications:

- Word-align the parallel corpora of the two cognate languages. We use *eflomal* (Östling and Tiedemann, 2016) and symmetrize the alignment with the *grow-diag-final-and* heuristic.
- Remove all word pairs that contain punctuation or occur less than 5 times.
- Filter the list of word pairs based on Levenshtein distance. If either of the words consists of 4 or fewer characters, an exact match is required. Otherwise, a Levenshtein distance up to a third of the mean of the lengths is allowed.
- Further filter the list to remove one-to-many and many-to-one mappings, keeping only the most frequent pairing.

Cognate Morfessor models have to be trained on the full vocabulary, not only the cognate pairs.[8] Therefore, the list of cognate pairs is complemented with unaligned source-only and target-only items. This resulted in a training vocabulary of 140 227 entries for Spanish–Portuguese (63 355 cognate pairs + 35 351 monolingual ES words +

[7] https://morfessor.readthedocs.io/en/latest/cmdtools.html#morfessor

[8] See https://github.com/Waino/morfessor-cognates.

Model	Parameters	Train. epochs	Vocab. size	Proc ref BLEU	Raw reference BLEU	TER	cTER	chrF2
ES → PT								
Characters-Initial		5.0	562	52.46	53.90	27.00	19.61	76.72
‡ Characters		1.8	813	54.62	56.20	25.63	18.07	77.96
Morfessor Baseline	$\alpha = 0.03$	2.5	3090	57.43	56.14	26.38	18.36	77.88
‡ Morfessor Baseline	$\alpha = 0.05$	2.5	5187	56.94	55.28	28.76	18.64	77.43
Cognate Morfessor	$\alpha = 0.001$	2.5	2818	57.26	55.89	27.85	18.76	77.58
∗ Cognate Morfessor	$\alpha = 0.01$	2.5	3884	56.92	55.41	27.60	18.61	77.45
SentencePiece Unigram	$\|V\| = 5000$	2.5	7668	**59.76**	**57.79**	**25.58**	**17.55**	**78.52**
Byte Pair Encoding	$\|V\| = 5000$	2.5	6224	58.79	56.92	26.01	17.86	78.25
PT → ES								
Characters-Initial		4.0	562	55.38	56.20	26.35	18.68	78.24
‡ Characters		2.0	834	60.69	**62.10**	**22.61**	15.68	**81.47**
Morfessor Baseline	$\alpha = 0.03$	2.5	3090	62.78	60.77	23.30	15.81	81.32
‡ Morfessor Baseline	$\alpha = 0.05$	2.5	5187	**62.89**	60.87	23.42	**15.63**	81.34
Cognate Morfessor	$\alpha = 0.001$	2.5	2818	60.05	58.11	27.67	15.91	80.95
∗ Cognate Morfessor	$\alpha = 0.01$	2.5	3884	61.41	59.48	25.67	16.01	81.16
SentencePiece Unigram	$\|V\| = 5000$	2.5	7664	62.06	60.27	24.68	16.75	80.05
Byte Pair Encoding	$\|V\| = 5000$	2.5	6225	61.52	59.77	25.22	17.18	79.58
CS → PL								
Characters-Initial		11.1	419	8.51	8.64	79.16	68.33	35.97
‡ Characters		5.5	486	10.45	10.60	76.91	61.89	39.75
Morfessor Baseline	$\alpha = 0.03$	5.5	4181	**12.17**	**11.90**	75.27	61.83	40.72
‡ Morfessor Baseline	$\alpha = 0.05$	5.5	7255	11.93	11.71	76.12	62.29	40.46
Cognate Morfessor	$\alpha = 0.001$	5.5	2884	12.13	11.88	**75.24**	61.65	40.88
∗ Cognate Morfessor	$\alpha = 0.01$	5.5	4186	11.90	11.66	75.76	**61.00**	**40.96**
SentencePiece Unigram	$\|V\| = 5000$	5.5	8841	9.98	9.74	77.25	66.37	37.39
Byte Pair Encoding	$\|V\| = 5000$	5.5	6264	10.01	9.80	77.10	66.32	37.39
PL → CS								
Characters-Initial		11.2	419	11.14	11.34	71.06	71.77	34.39
‡ Characters		3.0	868	14.98	15.33	66.69	64.77	38.35
Morfessor Baseline	$\alpha = 0.03$	3.0	4181	15.68	15.39	66.06	64.55	39.22
‡ Morfessor Baseline	$\alpha = 0.05$	3.0	7255	15.80	15.52	66.45	64.36	39.30
Cognate Morfessor	$\alpha = 0.001$	3.0	2884	**16.02**	**15.73**	**65.82**	**64.12**	39.56
∗ Cognate Morfessor	$\alpha = 0.01$	3.0	4186	15.75	15.48	66.09	64.71	**39.20**
SentencePiece Unigram	$\|V\| = 5000$	3.0	8682	13.56	13.28	67.44	69.03	36.93
Byte Pair Encoding	$\|V\| = 5000$	3.0	5939	14.29	14.08	67.39	68.30	37.49

Table 3: Key figures and results of our experiments on the development set. All scores are percentage values. *Proc ref* refers to a preprocessed and postprocessed version of the reference. Primary submissions are marked with ∗, contrastive submissions with ‡.

Model	BLEU	TER
ES → PT		
Characters	**52.8**	**28.6**
Morfessor Baseline ($\alpha = 0.05$)	51.0	33.1
Cognate Morfessor ($\alpha = 0.01$)	52.0	29.4
PT → ES		
Characters	**59.1**	25.5
Morfessor Baseline ($\alpha = 0.05$)	58.6	**25.1**
Cognate Morfessor ($\alpha = 0.01$)	58.4	25.3
CS → PL		
Characters	5.9	88.4
Morfessor Baseline ($\alpha = 0.05$)	7.0	**87.3**
Cognate Morfessor ($\alpha = 0.01$)	**7.1**	87.4
PL → CS		
Characters	6.6	80.2
Morfessor Baseline ($\alpha = 0.05$)	**7.2**	79.6
Cognate Morfessor ($\alpha = 0.01$)	7.0	**79.4**

Table 4: Official results of the submitted systems. BLEU scores are based on *mt-eval-v13b*. The Cognate Morfessor systems are primary submissions.

41 521 monolingual PT words) and 183 706 entries for Czech–Polish (34 291 cognate pairs + 71 416 monolingual CS words + 77 999 monolingual PL words). It clearly appears that the number of cognate pairs is proportionally much lower for Czech–Polish than for Spanish–Portuguese, and further experiments will be required to quantify the impact of the cognate extraction heuristics on these results.

Cognate Morfessor has two hyper-parameters: the monolingual corpus cost (α) and the edit operation weight. We keep the recommended value of 10 for the edit operation and experiment with two values of α, 0.01 and 0.001. Moreover, we disable the word-final epsilon symbol, which had been introduced by Grönroos et al. (2018) to account for situations where two aligned words do not have the same number of morphs. An inspection of our data showed that this configuration occurred very rarely in both language families.

The *Cognate Morfessor* lines in Table 3 show the NMT results obtained with these models. Again, the choice of α value does not have a consistent impact on the results. The cognate Morfessor models consistently outperform the character models when evaluated against the processed reference, but not when evaluated against the raw reference. They obtain very similar results compared to the standard Morfessor approach.

Based on the results obtained on the development data and the ability to specifically simulate the conditions of closely related morphologically rich languages, we selected the Cognate Morfessor models with $\alpha = 0.01$ as our primary systems.

4.4 SentencePiece unigram models

We trained the segmentation models only on the available parallel datasets for each language pair, following the findings of our submission to the WMT18 translation task (Raganato et al., 2018). We specified a vocabulary size of 5,000 tokens for each language and we took advantage from the tokenizer integrated in the SentencePiece implementation (Kudo and Richardson, 2018) by training the models on non-tokenized data. We applied the same truecasing models as before.

Results reported in Table 3 show that the models trained on SentencePiece-encoded data are consistently behind the Morfessor Baseline and Cognate Morfessor ones, except for the Spanish–Portuguese translation direction. This might be caused by the choice of vocabulary size used and the selected epoch in the table. These models had not converged at the reported time, results were chosen such that different models could be comparable. Once converged, they achieved better BLEU scores, but still fall behind the Cognate Morfessor models.

4.5 Byte pair encoding models

We ran further contrastive experiments using the well-known BPE segmentation (Sennrich et al., 2016). Since the BPE models serve here only for comparison purposes, we set them to be as comparable as possible to the other experiments. For this reason, we jointly trained them on the parallel datasets for each language pair and specified them to have 5,000 merge operations. Said segmentation models were trained on previously tokenized and truecased data.

5 Test results

We submitted three systems per language pair. The official results are reproduced in Table 4. The good performance of the character-level models on Spanish–Portuguese and Portuguese–Spanish can be attributed to the absence of pre- and post-processing, as illustrated in Table 3, rather than to

the underlying model architecture. The two Morfessor systems can be considered equivalent, as no clear winner emerges. The two official evaluation metrics BLEU and TER do not rank the systems consistently.

Character-level metrics were not provided by the organizers, but follow-up experiments showed that chrF2 yields the same rankings as BLEU, whereas CharacTer deviates from BLEU and TER.

The results of our submissions – and of many competitors in this shared task – lie very closely together. Before drawing any conclusions, it would therefore be useful to perform statistical significance testing. MultEval (Clark et al., 2011) provides significance scores through bootstrap resampling, but requires the output from multiple training runs of the same translation system. Unfortunately, we were not able to complete multiple training runs of our models due to time constraints.

6 Conclusions

The University of Helsinki participation focused on a single aspect of neural machine translation, namely subword segmentation. Subword segmentation that is consistent across the two languages has shown numerous benefits in translation quality, especially with respect to morphologically complex languages and for the translation (or transliteration) of rare words.

One of the investigated subword segmentation algorithms, Cognate Morfessor, was previously used successfully in a multilingual setting (translating from English to two related languages, Finnish and Estonian), and it seemed appealing to us to test this approach on similar language pairs from the Romance and Slavic language families. We contrasted the Cognate Morfessor models with three generic segmentation approaches: character segmentation, Morfessor Baseline, and SentencePiece. Our results did not show conclusive evidence that Cognate Morfessor would outperform the segmentation algorithms that did not use the information on cognates, but we have only explored a small area of the parameter space. In particular, the impact of the vocabulary size – independently of the segmentation method – on translation quality should be investigated further.

One rather surprising finding is the competitiveness of character-based models in the test evaluation for the Romance languages. This suggests that rule-based preprocessing and postprocessing scripts such as tokenization, punctuation normalization etc. can have a significant impact on the resulting output and penalize systems that rely on these scripts. Note, however, that models with a few thousand vocabulary units are typically much more efficient than pure character-level models in terms of training and decoding.[9]

It is obvious that other aspects than subword segmentation may have a decisive impact on translation quality: parallel corpus filtering methods, the amount and quality of back-translations, as well as fine-tuning towards the target domain are known to be important factors. We have not considered these factors in our submissions, but the shared task setup provides an interesting test bed for further experiments.

Acknowledgments

We would like to thank Stig-Arne Grönroos for the help with Cognate Morfessor.

The authors gratefully acknowledge the support of the Academy of Finland through project 314062 from the ICT 2023 call on Computation, Machine Learning and Artificial Intelligence. The authors also acknowledge CSC – IT Center for Science, Finland, for computational resources.

This work is part of the FoTran project, funded by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 771113).

References

Duygu Ataman, Matteo Negri, Marco Turchi, and Marcello Federico. 2017. Linguistically motivated vocabulary reduction for neural machine translation from Turkish to English. *The Prague Bulletin of Mathematical Linguistics*, 108(1):331–342.

Jonathan H. Clark, Chris Dyer, Alon Lavie, and Noah A. Smith. 2011. Better hypothesis testing for statistical machine translation: Controlling for optimizer instability. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 176–181, Portland, Oregon, USA. Association for Computational Linguistics.

[9]For instance, the PL → CS Cognate Morfessor model took 66 hours of training on a single GPU to complete three full epochs, whereas the character-level model took 116 hours for three epochs. Decoding of both development and test set took about 20 minutes with the former and 45 minutes with the latter.

Marta R. Costa-jussà, Carlos Escolano, and José A. R. Fonollosa. 2017. Byte-based neural machine translation. In *Proceedings of the First Workshop on Subword and Character Level Models in NLP*, pages 154–158, Copenhagen, Denmark. Association for Computational Linguistics.

Marta R. Costa-jussà and José A. R. Fonollosa. 2016. Character-based neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 357–361, Berlin, Germany. Association for Computational Linguistics.

Mathias Creutz, Teemu Hirsimäki, Mikko Kurimo, Antti Puurula, Janne Pylkkönen, Vesa Siivola, Matti Varjokallio, Ebru Arisoy, Murat Saraçlar, and Andreas Stolcke. 2007. Morph-based speech recognition and modeling of out-of-vocabulary words across languages. *ACM Transactions on Speech and Language Processing*, 5(1):3:1–3:29.

Mathias Creutz and Krista Lagus. 2002. Unsupervised discovery of morphemes. In *Proceedings of the ACL 2002 Workshop on Morphological and Phonological Learning*, volume 6 of *MPL '02*, pages 21–30, Stroudsburg, PA, USA. Association for Computational Linguistics.

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

Philip Gage. 1994. A new algorithm for data compression. In *C Users J*, pages 23–28.

Stig-Arne Grönroos, Sami Virpioja, and Mikko Kurimo. 2017. Extending hybrid word-character neural machine translation with multi-task learning of morphological analysis. In *Proceedings of the Second Conference on Machine Translation*, pages 296–302, Copenhagen, Denmark. Association for Computational Linguistics.

Stig-Arne Grönroos, Sami Virpioja, and Mikko Kurimo. 2018. Cognate-aware morphological segmentation for multilingual neural translation. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 386–393, Belgium, Brussels. Association for Computational Linguistics.

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: Technical Papers*, pages 1177–1185, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.

Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander Rush. 2017. OpenNMT: Open-source toolkit for neural machine translation. In *Proceedings of ACL 2017, System Demonstrations*, pages 67–72, Vancouver, Canada. 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*, pages 78–86, Uppsala, Sweden. Association for Computational Linguistics.

Taku Kudo. 2018. Subword regularization: Improving neural network translation models with multiple subword candidates. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 66–75, Melbourne, Australia. 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.

Mikko Kurimo, Antti Puurula, Ebru Arisoy, Vesa Siivola, Teemu Hirsimäki, Janne Pylkkönen, Tanel Alumäe, and Murat Saraclar. 2006. Unlimited vocabulary speech recognition for agglutinative languages. In *Proceedings of the 2006 Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics (HLT-NAACL)*, HLT-NAACL '06, pages 487–494, Stroudsburg, PA, USA. Association for Computational Linguistics.

Jason Lee, Kyunghyun Cho, and Thomas Hofmann. 2017. Fully character-level neural machine translation without explicit segmentation. *Transactions of the Association for Computational Linguistics*, 5:365–378.

Robert Östling and Jörg Tiedemann. 2016. Efficient word alignment with Markov Chain Monte Carlo. *Prague Bulletin of Mathematical Linguistics*, 106:125–146.

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

Maja Popović. 2016. chrF deconstructed: beta parameters and n-gram weights. In *Proceedings of the First Conference on Machine Translation*, pages 499–504, Berlin, Germany. Association for Computational Linguistics.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Belgium, Brussels. Association for Computational Linguistics.

Nima Pourdamghani and Kevin Knight. 2017. Deciphering related languages. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2513–2518, Copenhagen, Denmark. Association for Computational Linguistics.

Alessandro Raganato, Yves Scherrer, Tommi Nieminen, Arvi Hurskainen, and Jörg Tiedemann. 2018. The University of Helsinki submissions to the WMT18 news task. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 488–495, Belgium, Brussels. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Peter Smit, Sami Virpioja, and Mikko Kurimo. 2017. Improved subword modeling for WFST-based speech recognition. In *Proc. Interspeech 2017*, pages 2551–2555.

Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of Association for Machine Translation in the Americas*.

Gongbo Tang, Fabienne Cap, Eva Pettersson, and Joakim Nivre. 2018. An evaluation of neural machine translation models on historical spelling normalization. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1320–1331, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Jörg Tiedemann. 2009. Character-based PSMT for closely related languages. In *Proceedings of EAMT 2009*, page 12–19, Barcelona, Spain.

Matti Varjokallio, Mikko Kurimo, and Sami Virpioja. 2013. Learning a subword vocabulary based on unigram likelihood. In *IEEE Automatic Speech Recognition and Understanding Workshop, (ASRU 2013), Olomouc, Czech Republic, December 8-12, 2013*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.

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.

Weiyue Wang, Jan-Thorsten Peter, Hendrik Rosendahl, and Hermann Ney. 2016. CharacTer: Translation edit rate on character level. In *Proceedings of the First Conference on Machine Translation*, pages 505–510, Berlin, Germany. Association for Computational Linguistics.