Translation on Linguistically Segmented Data (Focus: Upper Sorbian ↔ German)

Nils Imdahl and Sebastian Ehmanns and Harshvardan Patel

Abstract

We investigate the effects of linguistically informed subword segmentation on neural machine translation between Upper Sorbian and German. Using a standard Transformer setup, we compare different segmentation methods such as Morfessor, augmentation with additional monolingual Sorbian data and Part-Of-Speech Tags against a simple BPE baseline. Our results show that Morfessor-based segmentation consistently speeds up model convergence and lowers perplexity, though the final translation quality does not surpass the BPE baseline. Adding monolingual Upper Sorbian data improved convergence but did not consistently improve BLEU/chrF2++, did not improve and in some cases hindered performance. These findings highlight important trade-offs between linguistically informed and purely statistical segmentation in low-resource machine translation.

1 Introduction

Morphologically rich languages pose several challenges for conventional NLP systems such as Neural Machine Translation (NMT), as combinations of morphemes and roots result in extreme type sparsity (Mager et al., 2022). Such challenges are particularly pronounced in low-resource settings, where limited parallel data amplifies the difficulty of handling complex morphology. Upper Sorbian serves as a natural testbed for this setting with fewer than 10,000 speakers (Bleakly, 2023) and a very rich inflectional morphology, which includes:

- Inflection of endings: čitać (to read) → čitam (I read), čitaš (you read)
- Ablaut (Vowel Alternations): brać (to take, impf.) → sym brał (I was taking) sym wzał (I took, perf.)
- Consonant Alternations: ruka (hand, nom. sg.) → ruce (dat. sg.)
- Suppletion: być (to be) → sym (I am), je (he/she/it is)
- Aspectual Inflection (Addition of Prefixes): čitać (to read) → přečitać (to finish reading, perfective), dočitać (to read up to a point)

For modern machine translation an important step is typically tokenization, i.e. splitting words into subwords. This is needed due to the otherwise extremely large vocabulary of individual words and their inflections in almost any natural language. The most extreme form of tokenization is character level tokenization where words are split into individual characters. This addresses the vocabulary size problem but introduces another: the resulting splits are often not semantically meaningful. Hence a good subword tradeoff is needed which keeps a low vocabulary size but preserves meaning.

One such approach, which has become the default tokenization method in NMT, is Byte Pair Encoding (BPE) (Gage, 1994). However, BPE is fundamentally frequency-driven and not linguistically informed: it often produces splits that ignore morpheme boundaries, resulting in segments that neither reflect meaningful units of the language nor capture its morphological structure, especially for morphologically rich languages like Upper Sorbian. Consider the verb *čitać* 'to read', which has the inflection: *čitam* (I read). With BPE, there is no guarantee of a linguistically correct split (*čita* + *m*). Instead it might get split as *čit* + *am*. Furthermore, BPE struggles with ablaut and suppletion, for example for *brać* (to take) we have *sym brał* (I was taking) and *sym wzał* (I took) for which BPE might treat *brał* and *wzał* as completely unrelated tokens.

Recent work (Machácek et al., 2018; Banerjee and Bhattacharyya, 2018; Mager et al., 2022) highlights these limitations and shows that alternative approaches to segmentation are needed for polysynthetic and morphologically complex languages. In

this work, we explore whether unsupervised morphological segmentation via Morfessor (Virpioja et al., 2013) can provide tangible benefits over joint BPE for Upper Sorbian–German translation. We further examine the role of additional monolingual data and the incorporation of Part-Of-Speech (POS) Tags via a deep learning based morphological tagger (Schmid, 2019) as auxiliary features. Our goal is to identify segmentation and feature integration strategies that improve both translation quality and training efficiency.

2 Methods

2.1 Data

We relied on the data provided by the WMT22 shared task in Unsupervised MT and Very Low Resource Supervised MT, which is publicly available [1]. This dataset includes parallel and monolingual resources for Upper Sorbian and German, with already existing training, validation and test splits. This included around 500k lines of parallel data and about 1.5M lines of Upper Sorbian Monolingual Data, see Table 1. Each line is usually a single

Split	German	Upper Sorbian
Training	449,058	449,058
Validation	4,001	4001
Test	2,000	2,000
Monolingual	–	1,447,486

Table 1: Dataset for Upper Sorbian-German translation task (line count)

sentence. The distributions of number of words per line facetted by language and split can be seen in Figure 1. All distributions have a median between 11 and 12 words per line and are right skewed due to some longer sentences outliers.

2.1.1 Preprocessing

Before tokenization and training we followed a standard preprocessing pipeline roughly based on the steps used with Moses SMT (Koehn et al., 2007). Specifically, we employed the following pipeline:

1. Standardize punctuation to prevent duplicate tokens with identical meaning. E.g. conversion of curly quotes to straight quotes.
2. Segmentation of text into words and punctuation tokens to ensure consistent token boundaries.
3. Corpus Cleaning, which includes removal of empty lines, too long and too short sentences as well as enforcing a maximum ratio between source and target language.
4. Truecasing

2.2 Segmentation Strategies

We employed and compared several different segmentation strategies. We compared several segmentation strategies, all ending with BPE (16k merges) for comparability. Our BPE implementation of choice was suword-nmt (Sennrich et al., 2016, 2017). In the following we describe each of the different strategies:

- **BPE (p)**: BPE trained only on the (training) parallel data and applied consistently to all splits. This is our main baseline.
- **BPE (p + m)**: BPE trained jointly on the (training) parallel data plus the additional Upper Sorbian monolingual data. This allows BPE merges to take into account a much larger distribution of Sorbian forms.
- **Morfessor (p)**: Unsupervised morphological segmentation using Morfessor (Virpioja et al., 2013), trained on the parallel Upper Sorbian side. After segmentation, subword BPE with 16k merges is applied on top of the Morfessor output to maintain comparability of vocabulary sizes with the baseline.
- **Morfessor (p + m)**: Same as above, but Morfessor is trained on both the parallel Upper Sorbian data and the additional 1.5M lines of Sorbian monolingual data. This aims to produce morphologically informed segmentations that better reflect the true inflectional variety of the language.
- **RNN Tagger**: Each Upper Sorbian token is annotated with a POS/morphological tag using a pre-trained RNN-based tagger (Schmid, 2019). The tags are concatenated to the token stream as special symbols (e.g., *čitam_VBP*) before BPE training. This gives the model explicit access to morphosyntactic features during training.

[1] https://github.com/mariondimarco/WMT22_UnsupVeryLowResMT_Data

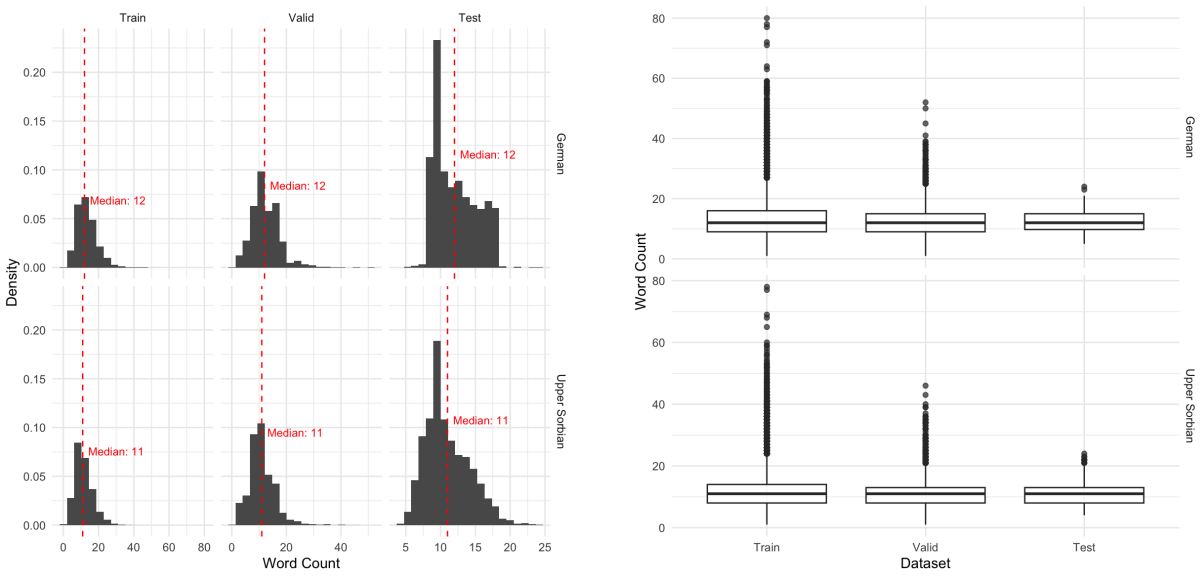

Figure 1: The word count distributions of the parallel Upper Sorbian and German Parallel Data Split by the dataset. The left plot shows a grid of histograms with 20 bins and the median value indicated via a dashed red vertical line. The right plot shows a grid of boxplots.

- **RNN Tagger + Morfessor**: Combination of the above two methods. Upper Sorbian tokens are first segmented using Morfessor and then tagged with the POS/morphological categories. Both segmentation units and tags are passed to BPE. This strategy integrates both morphological boundary information and syntactic features into the subword representation.

For the first four strategies we trained models in both translation directions, while for the last two we only trained models translating from Upper Sorbian into German. In total, we trained 10 different models.

2.3 Model And Training

Given a sentence in the source language $x_{1:i}$ with I tokens and a sentence in the target language $y_{1:j}$ with J tokens, Neural Machine Translation can be framed as modeling the following parameterized probability distribution:

$$p_\theta(y \mid x) = \prod_{j=1}^{J} p_\theta(y_j \mid y_{1:j-1}, x_{1:i})$$

To learn the parameters θ we trained an encoder-decoder Transformer architecture (Vaswani et al., 2017) with the fairseq framework (Ott et al., 2019). For details on the exact architecture see Table 2.

Each model was trained for 50 epochs using a cross-entropy objective with early stopping and a patience of 10 epochs. The early stopping criterion was SacreBLEU (see Section 2.5) score improvement. Optimization was done with Adam (Kingma and Ba, 2014) using a weight decay of 1×10^{-4}. An inverse square root learning rate scheduler with a starting learning rate of 5×10^{-4} was used. Additionally we applied a dropout of 0.3 and label smoothing of 0.1

Feature	Configuration
Embedding Dim.	512
Feedforward Dim.	1,024
Attention Heads	4
Encoder Layers	6
Decoder Layers	6
Total Parameters	49M

Table 2: The configuration of the trained Transformer model.

2.4 Training Efficiency

To complement our main evaluation metrics (cf. Section 2.5), we additionally tracked convergence behavior and perplexity throughout training.

Convergence. We define convergence as the point at which model performance on the validation set stabilizes, i.e., further training no longer yields improvements. In practice, this is monitored by observing the validation loss and stopping training early if no improvement occurs within a fixed

patience window (cf. Section 2.3). Measuring convergence speed allows us to assess how different segmentation strategies affect training efficiency, which is especially relevant in low-resource setups where computational resources are limited.

Perplexity. Perplexity is a standard measure of language model and NMT uncertainty and is directly derived from the cross-entropy loss. Given a reference sentence $y_{1:J}$ of length J and a model with parameters θ, the perplexity is defined as:

$$\text{PPL}(y) = \exp\left(-\frac{1}{J} \sum_{j=1}^{J} \log p_\theta(y_j \mid y_{1:j-1}, x) \right)$$

where $x_{1:I}$ is the source sentence of length I. Intuitively, perplexity measures the average branching factor of the model's probability distribution: lower values indicate that the model assigns higher probability mass to the correct sequence.

Usage in this study. For each segmentation strategy, we recorded perplexity on both the training and validation sets at every epoch. This allowed us to compare models not only in terms of final translation quality but also in terms of learning dynamics (speed of convergence, stability of training, and overall fit to the data distribution). However since perplexity measures only probability it is not necessarily an adequate measure of translation quality, which is why we define two automatic evaluation criteria in the following section.

2.5 Automatic Evaluation

Evaluation of the best trained model checkpoints was done on the unseen test set. We used BLEU (Papineni et al., 2002) and the chrF++ (Popović, 2017) score as our two main evaluation metrics.

BLEU is calculated as the weighted (w_n) geometric mean of the n-gram precision (p_n) but with a brevity penalty (BP) to punish translations shorter than the reference:

$$BLEU = BP \cdot exp(\sum_{n=1}^{\infty} w_n \ln p_n)$$

However, crucially this score is dependent on preprocessing and tokenization, which is why we used the standardized implementation SacreBLEU (Post, 2018).

The chrF++ score is tokenization independent and has been shown to track human judgement better than BLEU, especially for morphologically rich languages. As opposed to its predecessor chrF (Popović, 2015) which only used a character level F-score, chrF++ calculates and F-Score of the Word N-grams as well and takes the arithmetic mean. In our case we specifically use the chrF2++ score which refers to the fact that we calculate the F2-score (i.e we set $\beta = 2$ in the F-Score calculation).

3 Results

Figure 2 shows the validation loss curves for all segmentation strategies. Table 3 shows the final perplexity scores of the best model checkpoint for each segmentation strategy. Figure 3 presents the BLEU and chrF++ scores on the test set.

Strategy	HSB → DE	DE → HSB
BPE (p)	1.57	1.55
BPE (p+m)	1.52	1.55
Morfessor (p)	1.52	**1.52**
Morfessor (p+m)	**1.49**	1.58
RNN Tagger	1.59	–
RNN Tagger + Morfessor	1.96	–

Table 3: Final validation perplexities of all trained models. The first column shows the segmentation strategy applied (cf. Section 2.2). The second column (HSB → DE) shows the perplexities for the Upper Sorbian → German translation direction, while the third (DE → HSB) shows the reverse direction. The lowest (and therefore best) perplexities for each direction are highlighted in bold.

Upper Sorbian → German. Both the **BPE (p + m)** and **Morfessor (p)** approaches yielded very similar convergence and an improvement over the baseline **BPE (p)** strategy of 0.05 in final perplexity. The combination of Morfessor and Monolingual Data yielded the best result with the fastest convergence and a final perplexity of 1.49. Both RNN-Tagger strategies showed slower convergence than the baseline with the simple **RNN-Tagger** [PPL = 1.57] approach outperforming the **RNN-Tagger + Morfessor** [PPL = 1.96] one in final perplexity. In line with this perplexity, both **RNN-Tagger** approaches yielded worse scores compared to the baseline, with the simple **RNN-Tagger** [BLEU = 40.2, chrF2++ = 59.4] approach outperforming the combined **RNN-Tagger + Morfessor** [BLEU = 34.1, chrF2++ = 53.1] model. However, the **BPE (p)** baseline remained competitive, with [BLEU = 61.4, chrF2++ = 79.6] with both **Morfessor** models when it came to Automatic Evaluation. The **Morfessor (p)** and **Morfessor (p+m)** systems reached

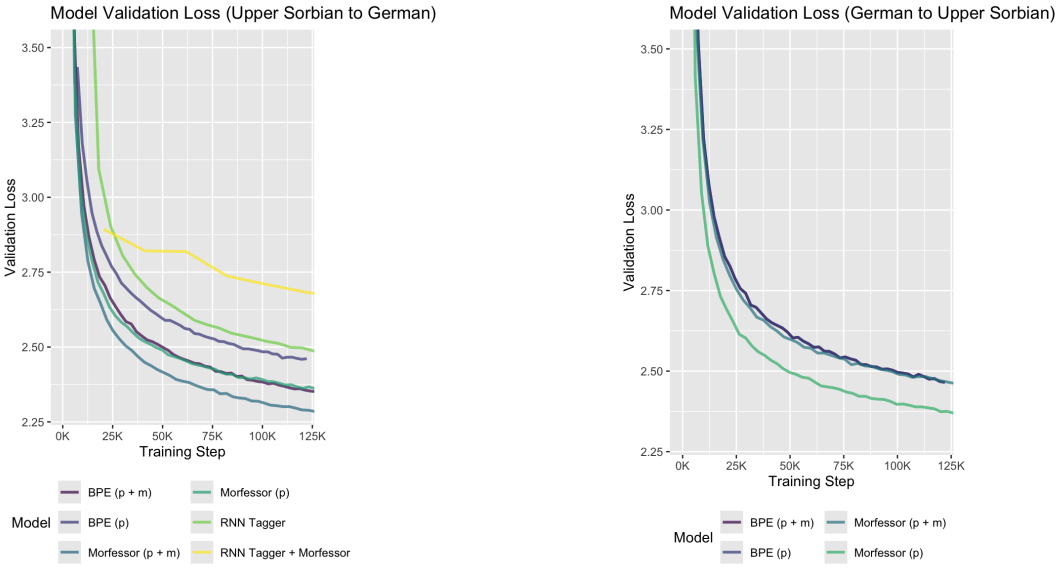

Figure 2: The validation set loss curves of the different segmentation strategies (c.f 2.2). The left plot shows the Upper Sorbian to German direction and a curve for each of the 6 trained models. The right plot shows the German to Upper Sorbian direction for each of the 4 trained models.

comparable chrF2++ and BLEU scores, but did not surpass the baseline. Adding monolingual data with **BPE (p+m)** did not improve the final BLEU or charF2++ scores either.

German → Upper Sorbian. In the reverse direction (DE → HSB), convergence and final perplexity is best for the **Morfessor (p)** [PPL = 1.52], outpeforming both the baseline and both monolingual data approaches. However once again in terms of BLEU and chrF2++ the simple **BPE (p)** [BLEU = 60.5, chrF2++ = 79.0] strategy did not get surpassed by any of the three other segmentation strategies.

4 Discussion

Our experiments show that **Morfessor-based segmentation consistently accelerates training**: models converge in fewer epochs and reach lower validation perplexity compared to BPE-only baselines. This suggests that linguistically motivated subword units can help the model learn more efficiently, which is particularly valuable in low-resource scenarios where training budgets and data are limited.

However, these efficiency gains do **not translate into higher final translation quality** as measured by BLEU or chrF++. In fact, a carefully tuned BPE baseline remains highly competitive. This highlights an important trade-off: while morphological segmentation provides faster convergence and lower perplexity, the downstream benefits to translation quality are modest at best.

The integration of **POS/morphological tags did not improve performance**. Instead, adding tags often hindered convergence and degraded BLEU/chrF++ scores. A likely reason is that concatenating coarse-grained tags directly into the token stream introduces noise and increases input complexity without providing sufficient disambiguating power. This suggests that more sophisticated ways of incorporating morphosyntactic information-such as feature embeddings or structured conditioning-may be required.

Taken together, our findings emphasize that **linguistically informed segmentation improves efficiency but does not guarantee translation quality improvements**. While it improves training dynamics, its impact on translation quality remains limited compared to strong statistical baselines. For practical deployment, the additional engineering and preprocessing required for Morfessor or POS integration may not justify the gains, unless training efficiency is the primary constraint.

5 Outlook And Future Work

Our study highlights both the promise and the limitations of linguistically informed segmentation in low-resource neural machine translation. Several directions remain open for future exploration:

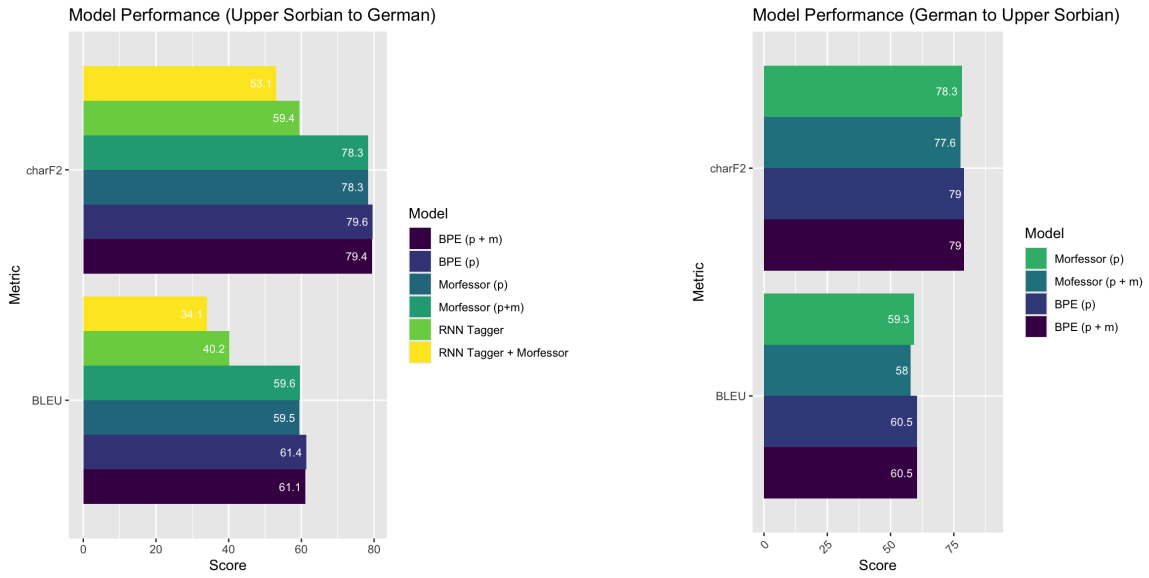

Figure 3: The BLEU and charF2++ scores for both translation directions and all segmentation strategies. Left: Upper Sorbian to German. Right: German to Upper Sorbian.

From basic to fine-tuned Morfessor models. In this work, we employed the standard unsupervised Morfessor model (Virpioja et al., 2013). However, more recent approaches allow for semi-supervised or language-specific adaptations where small amounts of annotated morphological resources guide segmentation (Creutz and Lagus, 2007; Grönroos et al., 2014). For Upper Sorbian, a fine-tuned Morfessor model trained with limited expert-provided segmentations or dictionaries could yield more linguistically consistent boundaries. Such an approach may mitigate cases where purely unsupervised segmentation diverges from true morpheme structure.

Canonical representations and morphological features. Another promising avenue is the use of canonical forms and feature-based abstraction. Instead of learning separate embeddings for every inflected form, models can map tokens to a normalized lemma with associated morphological features (Cotterell et al., 2016). For Upper Sorbian, this would mean abstracting over forms like *čitam*, *čitaš*, and *čitaće* into a canonical representation of *čitać* + person=1/2/3, tense=pres/fut. Such representations have the potential to reduce data sparsity and allow models to better generalize across inflectional paradigms.

Impact of model size. We restricted our experiments to a relatively small Transformer model of approximately 49M parameters for reasons of comparability and training efficiency. However, scaling laws suggest that larger models often achieve lower perplexity and higher downstream performance when sufficient data is available (Kaplan et al., 2020; Hoffmann et al., 2022). It remains an open question whether Upper Sorbian–German NMT benefits more from scaling model size or from improving data representations. Future work should systematically investigate how capacity interacts with segmentation strategies, particularly in low-resource but morphologically complex settings.

Back-translation with inflectional control. Back-translation (Sennrich et al., 2015) has become a standard method to leverage target-side monolingual data by generating synthetic source sentences. For Upper Sorbian–German, one promising variant would be to use an inflection-aware or inflection-controlled MT system to generate source-side sentences. By explicitly controlling or diversifying inflected forms in synthetic data, back-translation could not only increase training volume but also provide richer morphological coverage. This could address the sparsity of specific inflectional patterns in the parallel corpus and complement segmentation-based approaches.

6 Contributions

First and foremost we thank our supervisor Marion DiMarco for her guidance and support throughout this entire practical course. While the team collaborated closely on the project the contributions in terms of main areas of focus are listed below.

- **Nils Imdahl**: Implementation of the full integrated pipeline for model training and evaluation. Training and Evaluation of the models themselves, including visualization of the results. Main author of this paper. Small additions to and proofreading of the poster.
- **Sebastian Ehmanns**: Experimentation with an initial pipeline. Research of existing reference papers. Full RNN-Tagger implementation. Main creator and designer of the poster. Initial draft of this paper, followed by iterative proofreading.
- **Harshvardan Patel**: Creation of the Dataset based on the WMT22 Shared Task. Implementation of initial experiments. Representation of the team at the poster session. Proofreading of and small additions to both poster and paper.

References

Tamali Banerjee and Pushpak Bhattacharyya. 2018. Meaningless yet meaningful: Morphology grounded subword-level NMT. In *Proceedings of the Second Workshop on Subword/Character LEvel Models*, pages 55–60, New Orleans. Association for Computational Linguistics.

Evan Bleakly. 2023. Upper sorbian in budyšin / bautzen: Examples from bautzen's linguistic landscape. *LANGUAGE: Codification, Competence, Communication*, 1-2:20–39.

Ryan Cotterell, Christo Kirov, John Sylak-Glassman, David Yarowsky, Jason Eisner, and Mans Hulden. 2016. The sigmorphon 2016 shared task—morphological reinflection. In *The SIGMORPHON 2016 Shared Task—Morphological Reinflection*, pages 10–22.

Mathias Creutz and Krista Lagus. 2007. Unsupervised models for morpheme segmentation and morphology learning. *ACM Trans. Speech Lang. Process.*, 4(1).

Philip Gage. 1994. A new algorithm for data compression. *C Users J.*, 12(2):23–38.

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.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, and 3 others. 2022. Training compute-optimal large language models. *Preprint*, arXiv:2203.15556.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *CoRR*, abs/2001.08361.

Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *International Conference on Learning Representations*.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180. Association for Computational Linguistics.

Dominik Machácek, Jonás Vidra, and Ondrej Bojar. 2018. Morphological and language-agnostic word segmentation for NMT. *CoRR*, abs/1806.05482.

Manuel Mager, Arturo Oncevay, Elisabeth Mager, Katharina Kann, and Thang Vu. 2022. BPE vs. morphological segmentation: A case study on machine translation of four polysynthetic languages. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 961–971, Dublin, Ireland. Association for Computational Linguistics.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of NAACL-HLT 2019: Demonstrations*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, page 311–318, USA. Association for Computational Linguistics.

Maja Popović. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395, Lisbon, Portugal. Association for Computational Linguistics.

Maja Popović. 2017. chrF++: words helping character n-grams. In *Proceedings of the Second Conference on Machine Translation*, pages 612–618, Copenhagen, Denmark. 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.

Helmut Schmid. 2019. Deep learning-based morphological taggers and lemmatizers for annotating historical texts. In *Proceedings of the 3rd International Conference on Digital Access to Textual Cultural Heritage*, DATeCH2019, page 133–137, New York, NY, USA. Association for Computing Machinery.

Rico Sennrich, Alexandra Birch, Anna Currey, Ulrich Germann, Barry Haddow, Kenneth Heafield, Antonio Valerio Miceli Barone, and Philip Williams. 2017. The university of Edinburgh's neural MT systems for WMT17. In *Proceedings of the Second Conference on Machine Translation*, pages 389–399, Copenhagen, Denmark. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015. Improving neural machine translation models with monolingual data. *CoRR*, abs/1511.06709.

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.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *CoRR*, abs/1706.03762.

Sami Virpioja, Peter Smit, Stig-Arne Grönroos, and Mikko Kurimo. 2013. Morfessor 2.0: Python implementation and extensions for morfessor baseline. Aalto University publication series SCIENCE + TECHNOLOGY 25/2013, Aalto University, Helsinki.