NRC Systems for Low Resource German-Upper Sorbian Machine Translation 2020: Transfer Learning with Lexical Modifications

Rebecca Knowles and **Samuel Larkin** and **Darlene Stewart** and **Patrick Littell**
National Research Council Canada
{Rebecca.Knowles, Samuel.Larkin, Darlene.Stewart, Patrick.Littell}@nrc-cnrc.gc.ca

Abstract

We describe the National Research Council of Canada (NRC) neural machine translation systems for the German–Upper Sorbian supervised track of the 2020 shared task on Unsupervised MT and Very Low Resource Supervised MT. Our models are ensembles of Transformer models, built using combinations of BPE-dropout, lexical modifications, and backtranslation.

1 Introduction

We describe the National Research Council of Canada (NRC) neural machine translation systems for the shared task on Unsupervised MT and Very Low Resource Supervised MT. We participated in the supervised track of the low resource task, building Upper Sorbian–German neural machine translation (NMT) systems in both translation directions. Upper Sorbian is a minority language spoken in Germany. We built baseline systems (standard Transformer (Vaswani et al., 2017) with a byte-pair encoding vocabulary (BPE; Sennrich et al., 2016b)) trained on all available parallel data (60,000 lines), which resulted in unusually high BLEU scores for a language pair with such limited data.

In order to improve upon this baseline, we used transfer learning with modifications to the training lexicon. We did this in two ways: by experimenting with the application of BPE-dropout (Provilkov et al., 2020) to the transfer learning setting (Section 2.3), and by modifying Czech data used for training parent systems with word and character replacements in order to make it more "Upper Sorbian-like" (Section 2.4).

Our final systems were ensembles of systems built using transfer learning and these two approaches to lexicon modification, along with iterative backtranslation.

2 Approaches

2.1 General System Notes

In both translation directions, our final systems consist of ensembles of multiple systems, built using transfer learning (Section 2.2), BPE-Dropout (Section 2.3), alternative preprocessing of Czech data (Section 2.4), and backtranslation (Section 2.5). We describe these approaches and related work in the following sections, providing implementation details for reproducibility in Sections 3, 4 and 5.

2.2 Transfer Learning

Zoph et al. (2016) proposed a transfer learning approach for neural machine translation, using language pairs with larger amounts of data to pre-train a parent system, followed by finetuning a child system on the language pair of interest. Nguyen and Chiang (2017) expand on that, showing improved performance using BPE and shared vocabularies between the parent and child. We follow this approach: we build disjoint source and target BPE models and vocabularies, with one vocabulary for German (DE) and one for the combination of Czech (CS) and Upper Sorbian (HSB); see Section 4.

We chose to use Czech–German data as the parent language pair due to the task suggestions, relative abundance of data, and the close relationship between Czech and Upper Sorbian (cf. Lin et al., 2019; Kocmi and Bojar, 2018). While Czech and Upper Sorbian cognates are often not identical at the character level (Table 1), there is a high level of character-level overlap; trying to take advantage of that overlap without assuming complete character-level identity is a motivation for the explorations in subsequent sections (Section 2.3, Section 2.4). Another relatively high-resource language related to Upper Sorbian is Polish, but while the Czech and Upper Sorbian orthographies are fairly similar, mostly using the same characters for the same sounds (with a few notable exceptions), Polish orthography is more distinct. This, combined with the lack of a direct Polish–German parallel dataset in the constrained condition, led us to choose Czech as our transfer language for these experiments.

Czech	Upper Sorbian
analyzovat	analyzować
donesl	donjesł
externích	eksternych
hospodářská	hospodarsce
kreativní	kreatiwne
okres	wokrjes
potom	potym
projekt	projekt
sémantická	semantisku
velkým	wulkim

Table 1: A sample of probable Czech–Upper Sorbian cognates and shared loanwords, mined from the Czech–German and German–Upper Sorbian parallel corpora and filtered by Levenshtein distance.

Other work on transfer learning for low-resource machine translation includes multilingual seed models (Neubig and Hu, 2018), dynamically adding to the vocabulary when adding languages (Lakew et al., 2018), and using a hierarchical architecture to use multiple language pairs (Luo et al., 2019).

2.3 BPE-Dropout

We apply the recently-proposed approach of performing BPE-dropout (Provilkov et al., 2020), which takes an existing BPE model and randomly drops some merges at each merge step when applying the model to text. The goal of this, beside leading to more robust subword representations in general, is to produce subword representations that are more likely to overlap between the pre-training (Czech–German) and finetuning (Upper Sorbian–German) stages. We hypothesized that, in the same way that BPE-Dropout leads to robustness against accidental spelling errors and variant spellings (Provilkov et al., 2020), it could likewise lead to robustness to the kind of spelling variations we see between two related languages.

For example, consider the putative Czech–Upper Sorbian cognates and shared loanwords presented in Table 1. Sometimes a fixed BPE segmentation happens to separate shared characters into shared subwords (e.g. CS `analy@@ z@@ ovat` vs. HSB `analy@@ z@@ ować`), such that the presence of the former during pre-training can initialize at least some of the subwords that the model will later see in Upper Sorbian. However, other times the character-level differences lead to segmentations where no subwords are shared (e.g. CS `hospodář@@ ská` vs. HSB `hospodar@@ sce` or `potom` vs. HSB `po@@ tym`). Considering a wider variety of segmentations would, we hypothesized, mean that Upper Sorbian subwords would have more chance of being initialized during Czech pre-training (see Appendix C).

Rather than modifying the NMT system itself to reapply BPE-dropout during training, we treated BPE-dropout as a preprocessing step. Additionally, we experimented with BPE-dropout in the context of transfer learning, examining the effects of using source-side, both-sides, or no dropout in both parent and child systems.

2.4 Pseudo-Sorbian

For the Upper Sorbian–German direction, we also experimented with two techniques for modifying the Czech–German parallel data so that the Czech side is more like Upper Sorbian. In particular, we concentrated on modification methods that require neither large amounts of data, nor in-depth knowledge of the historical relationships between the languages, since both of these are often lacking for the lower-resourced language.

We considered two variations of this idea:

- *word-level* modification, in which some frequent Czech words (e.g. prepositions) are replaced by likely Upper Sorbian equivalents, and
- *character-level* modification, where we attempt to convert Czech words at the character level to forms that may more closely resemble Upper Sorbian words.

Note that in neither case do we know what particular conversions are *correct*; we ourselves do not know enough about historical Western Slavic to predict the actual Upper Sorbian cognates of Czech words. Rather, we took inspiration from stochastic segmentation methods like BPE-Dropout (Provilkov et al., 2020) and SentencePiece (Kudo and Richardson, 2018): when we have an idea of the *possible* solutions to the segmentation problem but do not know which one is the *correct* one, we can sample randomly from the possible segmentations as a sort of regularization, with the goal of discouraging the model from relying too heavily on a single segmentation scheme and giving it some exposure to a variety of possible segmentations. Whereas BPE-dropout and SentencePiece focus on possible segmentations of the word, our pseudo-Sorbian experiments focus on possible word- and character-level replacements. The goal was to discourage the parent Czech–German model from relying too heavily on regularities in Czech (e.g. the presence of particular frequent words, the presence of particular Czech character *n*-grams) and perhaps also gain some prior exposure to Upper Sorbian words and characters that will occur in the genuine Upper Sorbian data; we can also think of this as a form of low-resource data augmentation (Fadaee et al., 2017; Wang et al., 2018). See Appendix C for an analysis of increased subword overlap between pseudo-Sorbian and test data, as compared to BPE-dropout and the baseline approach.

2.4.1 Word-level pseudo-Sorbian

To generate the word-level pseudo-Sorbian, we ran `fast_align` (Dyer et al., 2013) on the Czech–German and German–Upper Sorbian parallel corpora, and took the product of the resulting word correspondences, to generate candidate Czech-Upper Sorbian word correspondences. As this process produces many unlikely correspondences, particularly for words that occur only a few times in the corpora, we filtered this list so that any Czech–German word correspondence that occurred fewer than 500 times in the aligned corpus was ineligible, and likewise any German–Upper Sorbian correspondence that occurred fewer than 50 times. We then used these correspondences to randomly replace 10% of eligible Czech words in the Czech-German corpus with one of their putative equivalents in Upper Sorbian. The result is a language that is *mostly* still Czech, but in which some high-frequency words (especially prepositions) are Upper Sorbian.

2.4.2 Character-level pseudo-Sorbian

To generate the character-level pseudo-Sorbian, we began with the same list of putative Czech-Upper Sorbian word correspondences, calculated the Levenshtein distances (normalized by length) between them, and filtered out pairs that exceeded 0.5 distance. This gave a list of words that were likely cognates, from which we hand-selected a development set of about 200; a sample of these is seen in Table 1. Using this set to identify character-level correspondences (e.g. CS `v` to HSB `w`, CS `d` to HSB `dź` before front vowels, etc.), we wrote a program to randomly replace the appropriate Czech character sequences with possible correspondences in Upper Sorbian. Again, as Czech-Upper Sorbian correspondences are not entirely predictable (CS `e` might happen to correspond, in a particular cognate, to HSB `e` or `ej` or `i` or `a` or `o`, etc.), we cannot expect that any given result is correct Upper Sorbian. Rather, we can think of this process as attempting to train a system that can respond to inputs from a variety of possible (but not necessarily actual) Western Slavic languages, rather than just a system that can respond to precisely-spelled Czech and only Czech.

2.4.3 Combined pseudo-Sorbian

In initial testing, we determined that a combination of word-level and character-level modification performed best; we ran each process on the Czech–German corpus separately, then concatenated the resulting corpora and trained a parent model on it. Due to time constraints we did not run the full set of ablation experiments. Subsequent finetuning on genuine Upper Sorbian–German data proceeded as normal, without any modification.

For all pseudo-Sorbian systems, we used the BPE vocabulary trained on the original Czech and Upper Sorbian data, rather than the modified data, so that systems trained on pseudo-Sorbian data could still be ensembled with systems trained only on the original data (Section 2.6).

2.5 Backtranslation

We used backtranslation (Sennrich et al., 2016a) to incorporate monolingual German and Upper Sorbian data into training. We backtranslated all Upper Sorbian monolingual data (after filtering as described in Section 3). We backtranslated the German monolingual news-commentary data and 1.2M randomly sampled lines of 2019 German news.

We experiment with iterative backtranslation: backtranslating data using systems without backtranslation, and then using the new systems built using the backtranslated text to perform a second iteration of backtranslation (Hoang et al., 2018; Niu et al., 2018; Zhang et al., 2018). Like Caswell et al. (2019), we use source-side tags at the start of backtranslated sentences to indicate to the models which sentences are the product of backtranslation.

2.6 Ensembling

Our final systems are ensembles of several systems. Because all systems used the same vocabulary sets and same model sizes, we could decode using Sockeye's (Hieber et al., 2018) default ensembling mechanism.

		Child Dropout		
		None	Source	Both
Parent Dropout	None	54.6	54.5	54.3
	Source	55.0	**55.5**	54.2
	Both	54.9	**55.5**	55.0

Table 2: Comparison of BPE-dropout use in both parent and child systems for 10k vocabulary DE-HSB translation (measured on devel_test set), without backtranslation. All parent systems were trained on the German-Czech data, while child systems trained on the parallel DE-HSB data. *None* involves no BPE-dropout, *source* applies BPE-dropout to the source side only, and *both* applies it to both the source and the target.

3 Data

We used all provided parallel German–Upper Sorbian data and all monolingual Upper Sorbian data (after filtering), along with German–Czech parallel data from Open Subtitles (Lison and Tiedemann, 2016),[1] DGT (Tiedemann, 2012; Steinberger et al., 2012), JW300 (Agić and Vulić, 2019), Europarl v10 (Koehn, 2005), News-Commentary v15, and WMT-News[2] for building the BPE vocabularies. The monolingual Upper Sorbian Web and Witaj datasets[3] were filtered to remove lines containing characters that had not been observed in the Upper Sorbian parallel data or in the Czech data; this removed sentences that contained text in other scripts and other languages. The Czech–German data was used for training parent models, while monolingual German and Upper Sorbian were used (along with parallel German–Upper Sorbian data) for training child models. A table of data sizes and how they were used is shown in Appendix A.

4 Preprocessing

We build BPE vocabularies of size 2k, 5k, 10k, 15k, and 20k using `subword-nmt`[4] (Sennrich et al., 2016b). After building the vocabulary, we add a set of 25 generic tags, plus a special backtranslation tag "<BT>", which we use in future experiments for indicating when training data has been backtranslated (Caswell et al., 2019). We also add all Moses and Sockeye special tags (ampersand, <unk> etc.) to a glossary file used for applying BPE, which prevents them from being segmented.

Because there is so much more Czech data than Upper Sorbian data, we duplicate the in-domain parallel hsb-de data and the monolingual HSB data 25 times when training BPE in order to make sure that HSB data is adequately represented (and not overwhelmed by Czech data) in training the encoding. After training BPE, we extract (and fix for the remainder of our experiments) a single DE vocabulary and a single HSB-CS vocabulary, covering all the relevant data used to train BPE for that language pair.

We ran BPE-dropout with a rate of 0.1 over the training data 5 times using the same BPE merge operations, vocabularies and glossaries as before, concatenating these variants to form an extended training set.

5 Software and Systems

We used Sockeye's (Hieber et al., 2018) implementation of Transformer (Vaswani et al., 2017) with 6 layers, 8 attention heads, network size of 512 units, and feedforward size of 2048 units. We have changed the default gradient clipping type to absolute, used the whole validation set during validation, an initial learning rate of 0.0001, batches of ∼8192 tokens/words, maximum sentence length of 200 tokens, optimizing for BLEU. Parent systems used checkpoint intervals of 2500 and 4000. Child system checkpoint intervals varied from 65 to 4000 to balance frequent checkpointing with efficiency. Decoding was performed with beam size 5.

6 Results and Discussion

6.1 BPE-Dropout in Transfer Learning

Provilkov et al. (2020) examine BPE-dropout when building translation systems for individual language pairs. Here we apply it in a transfer learning setting, raising the question of whether BPE-dropout should be applied to the parent system, the child system, or both, as well as the question of using source-side BPE-dropout or both source- and target-side BPE-dropout.

[1] http://www.opensubtitles.com

[2] http://www.statmt.org/wmt20/translation-task.html

[3] http://www.statmt.org/wmt20/unsup_and_very_low_res/

[4] https://github.com/rsennrich/subword-nmt

Our results for this are somewhat mixed, owing in part to the relatively small BLEU gains produced by BPE-dropout (as compared to backtranslation). In Table 2 we show BLEU scores for German–Upper Sorbian translation with a 10k vocabulary and no backtranslation. The most promising systems in that experiment are those with source-side BPE-dropout in the child system, with either both side or source-side dropout in the parent. In the 20k vocabulary DE-HSB setting with second iteration backtranslation, we saw a similar effect, with source BPE-dropout for both parent and child having a BLEU score of 58.4 on devel_test, +1.1 above the second-best system (no BPE-dropout in parent or child). Results in the other translation direction were more ambiguous, leaving room for future analysis of BPE-dropout in the transfer learning setting.

As a result of these experiments, many of the systems we used in our final ensembles were trained with source-side BPE-dropout, though when it appeared promising we also ensembled with systems without BPE-dropout.

6.2 Iterative Backtranslation

We performed two rounds of backtranslation of Upper Sorbian monolingual data and German monolingual data described in Section 2.5. The first round (BT1) used our strongest system without backtranslation, while the second round (BT2) used our strongest system including backtranslated data from the first round. We ran experiments sweeping BPE vocabulary sizes and backtranslated corpora; for German news we experimented 300k and 600k subsets as well as the full 1.2M line random subselection. In all experiments the 60k sentence-pair parallel HSB-DE corpus was replicated a number of times to approximately match the included backtranslated data in number of lines.

The second round of backtranslation of the Upper Sorbian monolingual data improved the BLEU score by 0.7 BLEU points for the best configuration, with the vocabulary size of the best configuration increasing to 20k from 15k. However, the second round of backtranslation of the German monolingual data did not improve the subsequent HSB-DE systems, instead showing a drop of 0.1 BLEU points; our final system (Section 6.5) uses a mix of systems trained using BT1 and BT2. For full details of the systems used for backtranslation, see Appendix B.

System	DE-HSB	HSB-DE
Baseline	44.2	44.1
Base. + BPE-Dr.	44.4	44.7
Base. + BT2	54.9	54.7
Base. + BT2 + BPE-Dr.	56.1	55.0
Child	54.7	53.4
Child + BPE-Dr.	55.5	54.1
Child + BT2	57.7	56.5
Child + BT2 + BPE-Dr.	58.4	56.8
Final Submitted Systems	59.4	58.9

Table 3: Ablation experiments showing performance of baseline systems, BPE-dropout, backtranslation, transfer learning, and their combination. All systems shown here do not use pseudo-Sorbian. DE-HSB systems here have a 20k vocabulary, while HSB-DE have a 10k vocabulary. BLEU score is reported on devel_test set. The final line shows the submitted primary systems and their performance on devel_test.

Generating multiple translation for backtranslation (i.e. multiple source sentences for each authentic target sentence) is known to improve translation quality (Imamura et al., 2018; Imamura and Sumita, 2018); all of the systems we have described here used a single backtranslation per target sentence. After the submission of our final systems, we experimented with backtranslation using *n*-best translations of the monolingual text. In both directions, we found that building student systems using the 10-best backtranslation list generated with sampling from the softmax's top-10 vocabulary (rather than taking the max), but without BPE-dropout, produced improvements of around 0.2-0.8 BLEU.[5] The resulting systems had comparable BLEU scores to the systems trained with single variant backtranslation and BPE-dropout; we leave as future work an examination of the result of combining multiple backtranslations with BPE-dropout.

6.3 Ablation

Here we first discuss the impact of our non-pseudo-Sorbian approaches: BPE-dropout, backtranslation, and transfer learning, showing how each contributed to the final systems used for ensembling.

Table 3 shows ablation experiments for DE-HSB (20k vocabulary) and HSB-DE (10k vocabulary).[6] In the first four lines, we consider training a system without transfer learning, starting from a base-

[5] Authentic bitext was upsampled to keep the ratio identical to our prior experiments.

[6] Smaller vocabulary sizes perform better on the baseline experiments, but the trends remain the same, so we show results for our final vocabulary sizes.

line built using only the parallel Upper Sorbian–German data. Despite the small data size, and perhaps due to the close match between training and test data, this baseline has high BLEU scores on the devel_test set: 44.2 (DE-HSB) and 44.1 (HSB-DE). Adding BPE-dropout to this setting (with 5 runs of the algorithm) results in a modest improvement (+0.2 BLEU for DE-HSB and +0.6 BLEU for DE-HSB). If we instead add backtranslated data (translated in our second iteration of backtranslation), we see a much larger jump of +10.7 and +10.6 BLEU respectively over the baselines; note that this also represents a huge increase in available data for training. Combining the two approaches adds an additional +1.2 and 0.3 BLEU, respectively.

In fact, these systems outperform both a parent-child baseline and a parent-child system with BPE-dropout, highlighting the importance of incorporating additional target-side monolingual data in the low-resource setting. Once we combine backtranslation we see a moderate improvement over the child systems with BPE-dropout (+2.6 and +2.4 BLEU, respectively). Again, combining BPE-dropout and backtranslation still produces more improvement, as does eventual ensembling.

Due to time constraints, we did not run a full ablation study of word, character and combined pseudo-Sorbian. Our initial results (run with an earlier version of character pseudo-Sorbian, and a differently extracted BPE vocabulary) found for the HSB-DE direction that word pseudo-Sorbian slightly outperformed (on the order of 0.5 BLEU) character pseudo-Sorbian for 10k vocabulary, but was comparable for 2k and 5k vocabulary sizes; these results are given in Appendix C. The combination of the two had slightly higher scores across those three vocabulary sizes (ranging from +0.1 to +0.6 BLEU) than either of the two individual approaches, so we used the combination for the remaining experiments.

6.4 Final German–Upper Sorbian System

System	BLEU
1. Child + BT2	57.7
2. Child + Src. BPE-Dr. + BT2	58.4
3. Pseudo-Sorbian + Child + BT2	57.8
4. Pseudo. + Child + Src. BPE-Dr. + BT2	58.2
Ensemble	**59.4**

Table 4: Primary German–Upper Sorbian ensemble submission BLEU score on devel_test, with scores of each of its individual component systems. The system numbers correspond to the list in Section 6.4.

Our final German–Upper Sorbian system is an ensemble of four systems, with vocabulary size of 20k merges. All child models ensembled were trained on second iteration backtranslated monolingual HSB data (all available, filtered) and 12 replications of the de–hsb parallel text, with backtranslation tags.

1. Child without BPE-dropout, de–cs parent without BPE-dropout.
2. Child with source side BPE-dropout, de–cs parent with source side BPE-dropout
3. Child without BPE-dropout, pseudo-hsb–de parent without BPE-dropout.
4. Child with source side BPE-dropout, pseudo-hsb–de parent with source side BPE-dropout

The system scores on devel_test are shown in Table 4. The best scoring individual systems were transfer learning systems with source-side BPE-dropout, with the one using pseudo-Sorbian falling slightly behind the non-pseudo-Sorbian by 0.2 BLEU points. Without BPE-dropout, the best pseudo-Sorbian system shown here outperforms its corresponding non-pseudo-Sorbian system by approximately 0.1 BLEU. On the test set, this system had scores of (as computed by the Matrix submission) 57.3 BLEU-cased, TER (Snover et al., 2006) of 0.3, BEER 2.0 (Stanojević and Sima'an, 2014) of 0.754, and CharacTER (Wang et al., 2016) of 0.255. This was 3.4 BLEU-cased behind the best-scoring system (SJTU-NICT), but within 0.6 BLEU of the second- and third-highest scoring systems (University of Helsinki); it was also tied with the third-highest scoring system (University of Helsinki) in terms of CharactTER.

6.5 Final Upper Sorbian–German System

System	BLEU
1. Child + BPE-Dr. + BT1	57.2
2. Child + BT2	57.1
3. Pseudo. + Child + BT1	57.2
4. Pseudo. + Child + BPE-dr. + BT1	57.1
5. Pseudo. + Child + BT2	57.1
Ensemble	**58.9**

Table 5: Primary Upper Sorbian–German ensemble submission BLEU score on devel_test, with scores of each of its individual component systems. The numbers correspond to the list in Section 6.5.

The final Upper Sorbian–German system is an ensemble of systems with a BPE vocabulary of 10k merges.

1. Child with source side BPE-dropout, 20 times hsb–de data, 1.2M lines of first iteration backtranslated news data; cs–de parent with source side BPE-dropout
2. Child without BPE-dropout, 25 times hsb–de data, news commentary (NC) and 1.2M lines of news second iteration backtranslated;[7] cs-de parent without BPE-dropout
3. Child without BPE-dropout, 25 times hsb–de data, NC and 1.2M lines of news first iteration backtranslated; pseudo-hsb–de parent without BPE-dropout
4. Child with source side BPE-dropout, 25 times hsb–de data, NC and 1.2M lines of news first iteration backtranslated; pseudo-hsb–de parent with source side BPE-dropout
5. Child without BPE-dropout, 20 times hsb–de data and 1.2M lines of second iteration backtranslated news data; pseudo-hsb–de parent without BPE-dropout

Table 5 shows that the five systems combined were very comparable in BLEU scores (57.1 and 57.2), but their ensembled BLEU score showed an improvement of $\geq$1.7 BLEU over each individual score. The final ensemble had a BLEU-cased score of 58.9 on the test data (calculated by the Matrix submission systems), a TER of 0.29, a BEER 2.0 of 0.579, and a CharacTER score of 0.268. This represented a -0.7 BLEU-cased difference off of the best system (University of Helsinki), but only a -0.001 CharactTER difference.

6.6 Discussion

We experimented with a variety of ensembles, and found that our strongest ensembles were those that included both the pseudo-Sorbian systems and those built without pseudo-Sorbian. In initial experiments with Upper Sorbian-German systems, with vocabulary size 5k, we found that adding pseudo-Sorbian systems to ensembles produced improvements even if the pseudo-Sorbian system did not have quite as high of a BLEU score as the systems built without it. For example, combining the top three systems without pseudo-Sorbian (BLEU scores of 57.3, 57.2, and 57.0, respectively) or the top two of those systems resulted in ensemble system BLEU scores of 57.9. Replacing the third-best system with a pseudo-Sorbian system with a BLEU score of 56.6 resulted in an improved ensemble BLEU score of 58.5. Diverse ensembles (e.g., different architectures or runs) are known to outperform less diverse ensembles (e.g., ensembles of checkpoints) for neural machine translation (Farajian et al., 2016; Denkowski and Neubig, 2017; Liu et al., 2018). While diversity of models for ensembling is usually discussed in terms of model architecture or seeding of multiple runs, we could argue that the use of lexically modified training data could constitute another form of model diversity, contributing to a stronger ensembled model.

For baseline systems trained only on the parallel data, smaller vocabulary sizes performed best, as expected (given only 60,000 lines of text, large vocabulary sizes may contain many tokens that are only observed a small number of times). As we added transfer learning, backtranslation, and eventually ensembling, the best systems were those with slightly larger vocabulary sizes. In the Upper Sorbian–German translation direction, some of our best performing systems that did not use pseudo-Sorbian were found with a 5k vocabulary size, while 10k was generally better for the pseudo-Sorbian systems. We tried ensembles with both 5k and 10k that included pseudo-Sorbian and non-pseudo-Sorbian systems, and found the best results with 10k.

7 Conclusions

In this work, we demonstrated that transfer learning, BPE-dropout, and backtranslation all provide improvements for this low-resource setting. Our experiments on lexical modifications, building pseudo-Sorbian text for training parent models, performed approximately on-par with standard transfer learning approaches, and could be trivially combined with BPE-dropout. While the lexical modification approach did not outperform the standard transfer learning setup, we found that it still improved ensembles, possibly due to the increase in system diversity.

Acknowledgments

We thank the reviewers for their comments and suggestions. We thank Yunli Wang, Chi-kiu Lo, Sowmya Vajjala, and Huda Khayrallah for discussion and feedback.

[7] This version of the second iteration backtranslation differs slightly from that used in the remainder of the experiments, in that UNKs (tokens representing unknown words) were not filtered out.

References

Željko Agić and Ivan Vulić. 2019. JW300: A wide-coverage parallel corpus for low-resource languages. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3204–3210, Florence, Italy. Association for Computational Linguistics.

Isaac Caswell, Ciprian Chelba, and David Grangier. 2019. Tagged back-translation. In *Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers)*, pages 53–63, Florence, Italy. Association for Computational Linguistics.

Michael Denkowski and Graham Neubig. 2017. Stronger baselines for trustable results in neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation*, pages 18–27, Vancouver. Association for Computational Linguistics.

Chris Dyer, Victor Chahuneau, and Noah A. Smith. 2013. A simple, fast, and effective reparameterization of IBM model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–648, Atlanta, Georgia. Association for Computational Linguistics.

Marzieh Fadaee, Arianna Bisazza, and Christof Monz. 2017. Data augmentation for low-resource neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 567–573, Vancouver, Canada. Association for Computational Linguistics.

M Amin Farajian, Rajen Chatterjee, Costanza Conforti, Shahab Jalalvand, Vevake Balaraman, Mattia A Di Gangi, Duygu Ataman, Marco Turchi, Matteo Negri, and Marcello Federico. 2016. FBK's neural machine translation systems for IWSLT 2016. In *Proceedings of the ninth International Workshop on Spoken Language Translation, USA*.

Felix Hieber, Tobias Domhan, Michael Denkowski, David Vilar, Artem Sokolov, Ann Clifton, and Matt Post. 2018. The sockeye neural machine translation toolkit at AMTA 2018. In *Proceedings of the 13th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Papers)*, pages 200–207, Boston, MA. Association for Machine Translation in the Americas.

Vu Cong Duy Hoang, Philipp Koehn, Gholamreza Haffari, and Trevor Cohn. 2018. Iterative back-translation for neural machine translation. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pages 18–24, Melbourne, Australia. Association for Computational Linguistics.

Kenji Imamura, Atsushi Fujita, and Eiichiro Sumita. 2018. Enhancement of encoder and attention using target monolingual corpora in neural machine translation. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pages 55–63, Melbourne, Australia. Association for Computational Linguistics.

Kenji Imamura and Eiichiro Sumita. 2018. NICT self-training approach to neural machine translation at NMT-2018. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pages 110–115, Melbourne, Australia. Association for Computational Linguistics.

Tom Kocmi and Ondřej Bojar. 2018. Trivial transfer learning for low-resource neural machine translation. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 244–252, Belgium, Brussels. Association for Computational Linguistics.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. *Proceedings of the 10th Machine Translation Summit (MT Summit)*, pages 79–86.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing.

Surafel M Lakew, Aliia Erofeeva, Matteo Negri, Marcello Federico, and Marco Turchi. 2018. Transfer learning in multilingual neural machine translation with dynamic vocabulary. In *International Workshop on Spoken Language Translation*.

Yu-Hsiang Lin, Chian-Yu Chen, Jean Lee, Zirui Li, Yuyan Zhang, Mengzhou Xia, Shruti Rijhwani, Junxian He, Zhisong Zhang, Xuezhe Ma, Antonios Anastasopoulos, Patrick Littell, and Graham Neubig. 2019. Choosing transfer languages for cross-lingual learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3125–3135, Florence, Italy. Association for Computational Linguistics.

Pierre Lison and Jörg Tiedemann. 2016. OpenSubtitles2016: Extracting large parallel corpora from movie and TV subtitles. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 923–929, Portorož, Slovenia. European Language Resources Association (ELRA).

Yuchen Liu, Long Zhou, Yining Wang, Yang Zhao, Jiajun Zhang, and Chengqing Zong. 2018. A comparable study on model averaging, ensembling and reranking in nmt. In *Natural Language Processing and Chinese Computing*, pages 299–308, Cham. Springer International Publishing.

G. Luo, Y. Yang, Y. Yuan, Z. Chen, and A. Ainiwaer. 2019. Hierarchical transfer learning architecture for low-resource neural machine translation. *IEEE Access*, 7:154157–154166.

Graham Neubig and Junjie Hu. 2018. Rapid adaptation of neural machine translation to new languages. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 875–880, Brussels, Belgium. Association for Computational Linguistics.

Toan Q. Nguyen and David Chiang. 2017. Transfer learning across low-resource, related languages for neural machine translation. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 296–301, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Xing Niu, Michael Denkowski, and Marine Carpuat. 2018. Bi-directional neural machine translation with synthetic parallel data. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pages 84–91, Melbourne, Australia. Association for Computational Linguistics.

Ivan Provilkov, Dmitrii Emelianenko, and Elena Voita. 2020. BPE-dropout: Simple and effective subword regularization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1882–1892, Online. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016a. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96, Berlin, Germany. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016b. 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.

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*, volume 200. Cambridge, MA.

Miloš Stanojević and Khalil Sima'an. 2014. Fitting sentence level translation evaluation with many dense features. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 202–206, Doha, Qatar. Association for Computational Linguistics.

Ralf Steinberger, Andreas Eisele, Szymon Klocek, Spyridon Pilos, and Patrick Schlüter. 2012. DGT-TM: A freely available translation memory in 22 languages. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*, pages 454–459, Istanbul, Turkey. European Languages Resources Association (ELRA).

Jörg Tiedemann. 2012. Parallel data, tools and interfaces in OPUS. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*, pages 2214–2218, Istanbul, Turkey. European Languages Resources Association (ELRA).

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 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pages 5998–6008.

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: Volume 2, Shared Task Papers*, pages 505–510, Berlin, Germany. Association for Computational Linguistics.

Xinyi Wang, Hieu Pham, Zihang Dai, and Graham Neubig. 2018. SwitchOut: an efficient data augmentation algorithm for neural machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 856–861, Brussels, Belgium. Association for Computational Linguistics.

Zhirui Zhang, Shujie Liu, Mu Li, Ming Zhou, and Enhong Chen. 2018. Joint training for neural machine translation models with monolingual data. In *AAAI*, pages 555–562.

Barret Zoph, Deniz Yuret, Jonathan May, and Kevin Knight. 2016. Transfer learning for low-resource neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1568–1575, Austin, Texas. Association for Computational Linguistics.

A Data

Table 6 shows the data sizes, including the size after filtering for the monolingual Upper Sorbian data, as well as how each dataset was used for BPE training and vocabulary extraction, parent training, and/or child training.

B Backtranslation Details

The configurations used to backtranslate the first round were:

- For monolingual Upper Sorbian, the HSB–DE child system with 5k vocabulary size and both source and target side BPE-dropout for both the HSB–DE system and its CS–DE parent (53.4 BLEU on devel_test)

- For monolingual German, the DE–HSB child with 10k vocabulary size and both source and target side BPE-dropout for both the DE–HSB system and its DE–CS parent (55.0 BLEU on devel_test).

The following configurations were used to backtranslate the second round:

- For monolingual Upper Sorbian, the HSB–DE child system with 5k vocabulary size and source side BPE-dropout for both the HSB–DE system and its CS–DE parent; 25 times hsb–de data, DE news commentary and 1.2M lines of DE news backtranslated (57.25 BLEU on devel_test)

- For monolingual German, the DE–HSB system with 15k vocabulary size and source side BPE-dropout for both the DE–HSB system and its DE–CS parent; 12 times hsb–de data, HSB Sorbian Institute, Witaj, and Web data backtranslated (57.7 BLEU on devel_test).

After the second round of backtranslation, the top configurations were:

- For HSB–DE, the 5k vocabulary size child with source side BPE-dropout for both the HSB–DE system and its CS–DE parent; 20 times hsb–de data, 1.2M lines of (second round) backtranslated DE news (57.15 BLEU on devel_test)

- For monolingual German, the 20k vocabulary size child with source side BPE-dropout for both the DE–HSB system and its DE–CS parent; 12 times hsb–de data, backtranslated (second round) HSB Sorbian Institute, Witaj, and Web data (58.4 BLEU on devel_test).

We note that the second round of backtranslating the German monolingual news data into Upper Sorbian did not improve the BLEU score for the subsequent HSB–DE systems, with the best configuration dropping by 0.1 BLEU points. However, the second round of backtranslation of the Upper Sorbian monolingual data did improve the BLEU score by 0.7 BLEU points for the best configuration, with the vocabulary size of the best configuration increasing to 20k from 15k.

C Pseudo-Sorbian Comparisons and Analysis

Table 7 presents the results of our pseudo-Sorbian comparison discussed in Sections 2.4 and 6.3; as mentioned; we find that both word- and character-level modifications are similar at small vocabulary sizes, but that word-level modification outperforms at a higher vocabulary size. However, at all vocabulary sizes a combination of the two improves over either approach on its own.

It should be noted again that these preliminary results are not directly comparable to other results in this paper (having trained on a smaller corpus, lacking the JW300 documents) and are also not technically constrained (as the word list used to create the character-level replacement was from bilingual dictionaries, not the constrained corpora). In our submitted systems, we created a new character-level system using only the constrained corpora.

As pseudo-Sorbian lexical modification creates a new training corpus, this raises questions of how to appropriately create BPE vocabularies, in particular when the character-level version is used. In word-level pseudo-Sorbian, the resulting corpus still only consists of words found in the original Czech and Upper Sorbian corpora, although the resulting n-gram frequencies will differ somewhat because of some Czech words being replaced by Upper Sorbian ones. Character-level pseudo-Sorbian, however, can create words and character-level n-grams that do not appear in the original corpus at all.[8]

[8] In future work, it would probably be beneficial to guide the output of the modification with a character-level language model trained on target-language data, to better avoid the generation of n-grams that are unlikely or unattested in the target language.

Data	Lines	BPE/Voc.	Parent	Child
train.hsb-de.{de,hsb}	60,000	Y ×25	N	Y
sorbian_institute_monolingual.hsb	339,822	Y ×25	N	Y
web_monolingual_filtered.hsb	131,047	Y ×25	N	Y
witaj_monolingual_filtered.hsb	220,564	Y ×25	N	Y
OpenSubtitles.cs-de.{de,cs}	16,378,674	Y	Y	N
DGT.cs-de.{de,cs}	4,853,298	Y	Y	N
JW300.{de,cs}	1,155,056	Y	Y	N
Europarl.cs-de.{de,cs}	568,572	Y	Y	N
News-Commentary.cs-de.{de,cs}	185,127	Y	Y	N
WMT-News.cs-de.{de,cs}	20,567	Y	Y	N
news.2019.de.shuffled.deduped.de	57,622,797	N	N	Y
news-commentary-v15.dedup.de	233,111	N	N	Y

Table 6: Data and how it was used, whether for BPE training and vocabulary extraction, parent model training, or child model training. Note that the monolingual German news.2019 data was subsampled, and the number of lines shown represents the full set from which the subsample was drawn.

Pseudo-Sorbian	BPE 2k	BPE 5k	BPE 10k
Word-level	51.8	52.6	52.6
Character-level	51.9	52.6	52.1
Both	52.4	52.7	**52.8**

Table 7: Comparison of approaches to create Pseudo-Sorbian corpora for pre-training, by word-level or character-level replacement of Czech text, at different vocabulary sizes. All scores represent BLEU scores on dev-test, in the HSB–DE direction.

The systems in Table 7 use system-specific BPE; that is, the BPE operations and vocabulary are constructed for each specific {pseudo-Sorbian, Upper Sorbian} training corpus. However, in the final submitted systems, we used a fixed vocabulary from the original {Czech, Upper Sorbian} corpus, which made it possible to ensemble pseudo-Sorbian systems with our other systems, giving us better results than either type of system alone. We do not know what effect (negative or positive) this may have on the quality of the pseudo-Sorbian-trained systems (since they would be using a BPE vocabulary for a different set of "languages", and thus may be over-segmented).[9] This raises a number of questions about appropriate choices of BPE models, which increases the complexity of ablation studies beyond what we are able to address in the scope of this paper.

Setting aside the complications of various BPE training schemes, we return to the BPE segmentations used in our final systems to analyze whether pseudo-Sorbian and BPE-dropout do indeed achieve their goals of producing more overlap between the pseudo-Sorbian or Czech training data and the Upper Sorbian data. We consider the devel_test portion of the Upper Sorbian data. With a 10k BPE vocabulary, that test set contains 4540 unique subword types. 62.6% of those types (2840) are observed in the baseline Czech parent model training data, and 52.9% of the training tokens are in that set. After applying BPE-dropout to the Czech parent training data, the percentage of observed types increases slightly, to 63.4% (2878), with 58.9% of the training tokens in that set. With the pseudo-Sorbian combined system, however, we see a much bigger increase in type overlap: 89.0% of the Upper Sorbian devel_test types (4041) were observed at least once in the pseudo-Sorbian parent data, making up 70.9% of the training tokens. Increased coverage of Upper Sorbian devel_test subword tokens during parent training means that embeddings for those subword tokens will be updated during parent model training, hopefully in a way that improves their warm start in the Upper Sorbian student training.[10]

[9] Using our final BPE segmentation does result in a slightly higher number of segmentations per token than a BPE model trained directly on the pseudo-Sorbian (combined version) data.

[10] While we could imagine that in some situations, they might end up with inappropriate representations, we expect those to be improved when the tokens are observed in student model training.