Data Augmentation for Low Resource Neural Machine Translation for Sotho-Tswana Languages

Maxwell Mojapelo[0009−0000−5398−4144] and Jan Buys[0000−0003−1994−5832]

Department of Computer Science, University of Cape Town, South Africa
mjpmak001@myuct.ac.za,jbuys@cs.uct.ac.za

Abstract. Neural Machine Translation (NMT) models have achieved remarkable performance on translating between high resource languages. However, translation quality for languages with limited data is much worse. This research focuses on the low resource language of Sepedi and considers two data augmentation techniques to increase the size and diversity of English-Sepedi corpora for training an NMT model. First we consider backtranslation, which makes use of the larger amount of available monolingual Sepedi text. We train a reverse (Sepedi to English) model and generate synthetic English sentences from the monolingual Sepedi sentences. These synthetic translations examples are added to the parallel English-Sepedi sentences. We carry out various experiments to investigate translation quality improvements. The second technique we consider is to generate synthetic data from parallel sentences between English and a closely-related language, Setswana. Setwana word are replacing with Sepedi words through an induced bilingual dictionary, which is created by using a supervised Generative Adversarial Network to align the embeddings of Sepedi and Setswana words. We evaluate our models on the JW300, FLoRes and Autshumato evaluation test sets, finding improvements over the current benchmark BLEU scores across all three datasets.

Keywords: Neural Machine Translation · Data Augmentation · Backtranslation · Word Replacement.

1 Introduction

Machine translation is a natural language processing task for automating the translation of text between two or more natural languages. Despite major advances in natural language processing over the past few decades, research efforts typically focus on a small number of high-resource languages [21], with less attention given to so-called low resource languages due to the limited availability of data. Most African languages fall into this category and remain understudied for machine translation, even though African languages account for 30.15% of all living languages [27]. Neural machine translation (NMT) models require numerous instances of sentence translation pairs covering diverse contexts [14].

Consequently, limited availability of training data reduces models' translation quality.

Data augmentation is a general approach in machine learning to generate additional training examples to complement available data, with the aim of improving the performance of models trained on the augmented training data [39]. In this research we investigate two data augmentation techniques to improve the translation quality of a NMT model, focusing our experiments on Sepedi, a low-resource language which belongs to the Sotho-Tswana language family.

The most widely used data augmentation technique for neural machine translation is backtranslation [32]. This technique remains largely unexplored for South African languages. We investigate backtranslation using Sepedi monolingual corpora from multiple domains. The second method we investigate is replacing some of the words in the training data of one language with words from another language [40] to generate additional synthetic training data. For South African languages this method has only been investigated for a classification task [23]. We apply the word replacement technique to the closely-related language of Setswana.

We start by establishing strong baselines using the Transformer architecture, replicating models from previously published work on NMT for South African languages [24, 2] for translation from English to Sepedi. All our NMT models are evaluated using the Bilingual Evaluation Understudy (BLEU) score, which compares the model's translation output to reference translations. We use hyperparameter tuning to improve performance by tuning model capacity (e.g. increasing the number of attention heads) and regularization (e.g. changing the dropout rate). A reverse (Sepedi to English) NMT model is then trained to backtranslate Sepedi monolingual data to synthetically generated English sentences; these sentences are combined with the available parallel English-Sepedi corpus to train augmented models.

For the word replacement, we train Sepedi and Setswana word embeddings using the continuous bag-of-words (CBOW) and skipgram approaches. Semi-supervised Generative Adversarial Networks are used to align the word embeddings to induce a Sepedi-Setswana bilingual dictionary. A pseudo English-Sepedi corpus is created by replacing each Setswana word in the English-Setswana parallel corpus with a Sepedi word, if a dictionary entry exists for that word. This pseudo English-Sepedi corpus is added to the training data and its effect on translation quality is investigated.

In summary, the contributions of this research are as follows:

1. Improving the current benchmark English-Sepedi BLEU scores by tuning the NMT model's capacity and regularization.
2. Benchmarking backtranslation on an English-Sepedi NMT model and exploring improvements through noising and filtering, resulting in BLEU score improvements ranging from +2 to +8 BLEU across three established test sets.
3. Benchmarking data augmentation by replacing Sepedi words in a English-Setswana corpus, comparing the translation quality improvements due to

using different embedding models for inducing a bilingual dictionary, as well as filtering the synthetic corpus. This results in BLEU score improvements ranging from +0.35 to +1.2 BLEU scores across two of the three test sets.

2 Related Work

2.1 Neural Machine Translation

Autoregressive neural machine translation (NMT) models the conditional probability of a target sentence $\mathbf{y} = \{y_1, y_2, y_3, ..., y_{n-1}, y_n\}$ of length n given a source sentence $\mathbf{x} = \{x_1, x_2, ..., x_{m-1}, x_m\}$ of length m [36, 34]. The chain rule is used to factorize the conditional probability into the product of the conditional probabilities of each next target word y_t consecutively,

$$P(y|x) = \prod_{t=1}^{n} P(y_t|y_0, y_1.y_2, ..., y_{t-1}, \mathbf{x}), \tag{1}$$

for each time step t. The inference problem is finding the most likely translation $\hat{\mathbf{y}}$ for a given input source sentence $\mathbf{x}$, that is $\hat{\mathbf{y}} = \text{argmax}P(\mathbf{y}|\mathbf{x})$ [34].

Earlier approaches [9, 35] mapped the input sequence to a fixed-length vector, and subsequently map that vector to the target sequence. Although this *encoder-decoder* architecture using recurrent neural networks performed well on short sentences, it performed poorly on longer sentences [8]. This led to new approaches being proposed, including the use of the Long-Short Term Memory (LSTM) architecture [35]. Most significantly was the use of an *attention mechanism* where the input sentence is encoded into a sequence of vectors with a weighted combination of these vectors used by the decoder at each time step [5].

This further led to the development of the Transformer model which uses only the attention mechanism to model the relationship between sequence elements [38]. The model applies the attention function multiple times through a projection of the query (Q) and key-value (K,V) vectors with different, learned linear projections. This gives rise to the Transformer architecture which uses multi-head attention and point-wise, fully connected layers. Given that there is no recurrence in the Transformer model, a *positional embedding* is added to each input embedding, capturing the relative or absolute position of tokens in a sequence. The encoder-decoder model for NMT uses self-attention among source and target sequences and cross-attention between them.

Transformer-based NMT models have multiple configurable components (e.g. embedding dimension size, number of layers, learning rates etc.) which enables varying training approaches [28]. NMT models require a fixed vocabulary, with an embeddings for each word in the vocabulary. This gives rise to the *out of vocabulary* words problem, where words not in the vocabulary cannot be handled at test time. Current NMT model predominantly uses Byte-Pair Encoding (BPE) [16], a data compression algorithm that merges frequently occurring pairs of characters repeatedly to construct a vocabulary which may include *subwords*, into which any word can be divided.

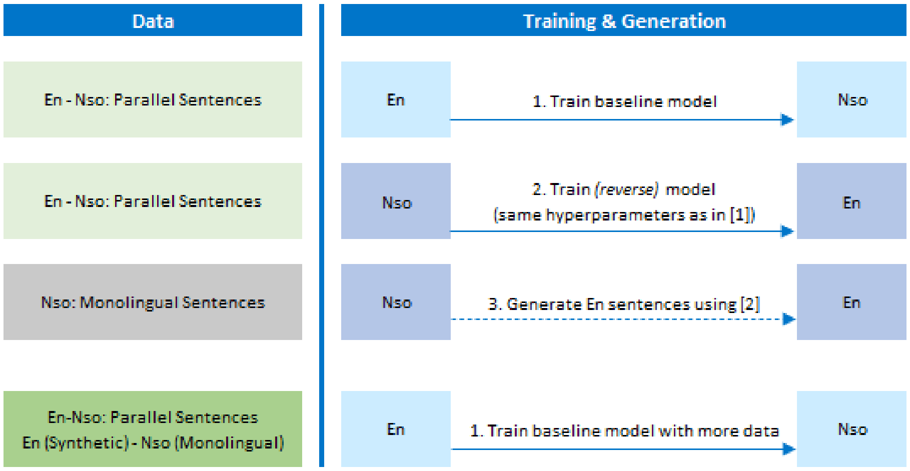
Data
Training & Generation
En - Nso: Parallel Sentences
En
1. Train baseline model
Nso
En - Nso: Parallel Sentences
Nso
2. Train (reverse) model (same hyperparameters as in [1])
En
Nso: Monolingual Sentences
Nso
3. Generate En sentences using [2]
En
En-Nso: Parallel Sentences En (Synthetic) - Nso (Monolingual)
En
1. Train baseline model with more data
Nso

Fig. 1: Overview of backtranslation for machine translation training

The standard automatic evaluation metric for machine translation is the Bilingual Evaluation Understudy (BLEU) score [30] which computes a modified precision of n-grams in the translated sentence compared to one or more reference sentences.

2.2 Data Augmentation

Data augmentation encompasses various methods and techniques that aim to increase the amount and diversity of training data without collecting more data [15]. These techniques attempt to generate extra data points from the empirically observed training set to train subsequent machine learning models and algorithms. The additional data can help to prevent models from performing poorly on new unseen data.

Backtranslation Backtranslation [32] makes use of a trained *target-to-source* NMT model to translate large amounts of monolingual target language data. This creates synthetic parallel sentence pairs which can then be used as additional training data for a source-to-target NMT model. This technique leverages semantic invariances encoded in supervised translation dataset to produce augmented data with similar semantic invariances [33]. The success of backtranslation has led to the development of many extensions [42, 10, 41].

Figure 1 illustrates the backtranslation process using English (En) as source language and Sepedi (Nso) as target language. The steps are as follows: (1) train a baseline NMT model using available parallel En-Nso sentences; (2) train a reverse NMT model *Nso-En* using the same available parallel En-Nso sentences; (3) use the *Nso-En* NMT model to generate *synthetic* En sentences from the

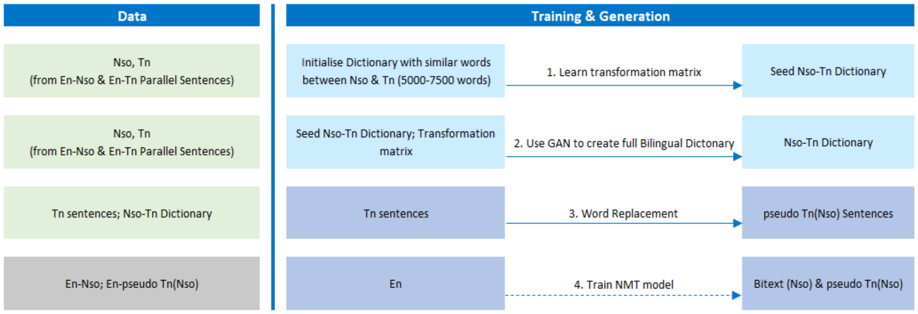
Data
Training & Generation
Nso, Tn (from En-Nso & En-Tn Parallel Sentences)
Initialise Dictionary with similar words between Nso & Tn (5000-7500 words)
1. Learn transformation matrix
Seed Nso-Tn Dictionary
Nso, Tn (from En-Nso & En-Tn Parallel Sentences)
Seed Nso-Tn Dictionary; Transformation matrix
2. Use GAN to create full Bilingual Dictonary
Nso-Tn Dictionary
Tn sentences; Nso-Tn Dictionary
Tn sentences
3. Word Replacement
pseudo Tn(Nso) Sentences
En-Nso; En-pseudo Tn(Nso)
En
4. Train NMT model
Bitext (Nso) & pseudo Tn(Nso)

Fig. 2: Overview of word replacement-based data augmentation for machine translation

monolingual Nso sentences; (4) train an NMT model based on the combined available *En-Nso* parallel sentences and synthetic sentences.

The additional monolingual data is useful for the overall objective of the NMT model as it helps improve the estimation of the prior probability of the target sentence, i.e., the fluency of the output. This exploits the fact that the encoder-decoder of the NMT conditions the probability distribution of the next target word on previous target words. The improvement in translation quality resulting from backtranslation inspired other studies including backtranslating source-side monolingual data [42], improving massively multilingual models [41] and backtranslating on a batch-basis instead of the full corpus [4].

Synthetic sentences can introduce poor translations and noise in the training data. Data filtering methods have been proposed to extract higher-quality sentences which are likely to optimize the translation model, which removing sentences with errors [19].

Word Replacement Data augmentation in the form of word replacement refers to approaches that perturb a dataset by replacing tokens with either different words from the current vocabulary or with new words that did not exist in that dataset. The word replacement technique [40] aims to exploit the semantic relationships that are captured in word vector representations [26].

Figure 2 provides an illustration of the word replacement process using Sepedi (Nso) as target language, Setswana (Tn) as highly related language, and English (En) as source language. The steps are as follows: (1) use words that are the same in Sepedi and Setswana to initialise the transformation matrix between word embeddings of Sepedi and Setswana words; (2) train a GAN model based on the matrix and apply the model on word embeddings for all words in the vocabularies to generate a bilingual dictionary; (3) replace all the aligned Setswana words in the Tn side of the En-Tn parallel corpus with the aligned Nso-words from the bilingual dictionary; (4) train an NMT model based on the combination of En-Nso parallel sentences and En-Tn parallel sentences with Nso word replacements.

The process of creating the bilingual dictionary is based on a transformation matrix W between the embedding spaces of the target and highly related languages, using normalized embeddings with orthogonal transformations. The transformation matrix is learned by aligning similar words between the target language and highly related language.

The unsupervised dictionary [22, 40] is created via a domain-adversarial training technique for all other words [17]. The model is trained to discriminate between the highly related language word embeddings Y and the transformed target word embeddings WX, with the discriminator aiming to make accurate predictions of the origin of the embeddings (i.e. Y or WX) and the generator aiming to make Y and WX as similar as possible.

To refine and translate the mapped embeddings to words in the dictionary, word pairs are added to the dictionary if they are the closest neighbours, where the Cross-Domain Similarity Local Scaling [22] metric is used as measure of closeness for the words in the resulting dictionary. The aim is to ensure that the selected highly related language word is the likeliest nearest neighbour of the source word in the highly related language.

2.3 NMT for South African Languages

NMT research focused specifically on South African and African languages remains limited compared to other languages. The research that pioneered NMT for South African languages [2] focused on Setswana, experimenting with multiple neural network architectures including convolution networks and the Transformer. BLEU scores achieved in this research ranged from 27.77 to 33.53, with the Transformer architecture performing best. An extension to previously uncovered South African languages of Northern Sotho (Sepedi) and Xitsonga using the Autshumato dataset achieved BLEU scores of 10.94 and 17.98 [1]. The current benchmark research carried out full coverage of all South African languages [24], training on the larger JW300 corpus [3]. More recent works have trained large multilingual NMT systems for selected African [13] and South African languages [12]; in this work we are not comparing directly to these approaches.

3 Data and Preprocessing

3.1 English-Sepedi and English-Setswana Parallel Corpora

The statistics of the English-Sepedi parallel corpora used in this work are given in Table 1. We use the JW300 dataset [3] as our primary data source. JW300 is a crawl of publications from the jw.org website consisting of multilingual articles mainly translated from English. The dataset excels in terms of coverage for low-resource languages. This dataset was accessed via Opus [37], using the test set created by the Masakhane project [27].

This dataset does have some issues, the first being the ideological bias emanating from jw.org, as well as domain bias given that the text mainly covers

Table 1: En-Nso parallel corpora for training and evaluation

Dataset	Sentences	Tokens (En)	Tokens (Nso)
JW300 Train	620 474	15.4m	12.7m
JW300 Validation	1 246	31k	25k
JW300 Test	2 711	56k	45k
Autshumato Test	514	9.2k	12.4k
FLoRes Test	2 009	42.8k	55.8k

Table 2: Sepedi Monolingual Corpora [31]

Dataset Name	Sentences	Tokens
NCHLT Sepedi Text Corpora	164k	3.5m
Curriculum Assessment Policy Statements	428k	9.1m
Government Cabinet Meetings Minutes	21 233	528k
Nal'ibali Short Stories	7 925	197k
South African Revenue Services Information	7 196	148k
Other Web Crawl	116k	2.4m
Total	745k	15.87m

biblical subjects. Another issue, though as far we know it does not directly effect the data for our experiments, is inconsistency in the language codes used [7].

We use the Autshumato Machine Translation Evaluation Set [25] for additional evaluation and comparability to other published research. This dataset consists of 500 sentences translated by four different professional human translators across all eleven South African languages.

Lastly, we use the FLoRes-101 Evaluation Benchmark dataset [18] (using both `dev` and `devtest`). This dataset consists of 3001 sentences professionally translated from English Wikipedia covering multiple domains including news, travel and various books, and consists of translations to 101 low-resource languages including Northern Sotho (Sepedi). It is standard practice in machine translation research that the evaluation sets are relatively small, due to the relatively large size of the training data and the challenge of getting high-quality translations with multiple references to ensure reliable evaluation.

The additional bitext dataset for the word replacement experiments is the English to Setswana parallel corpus from JW300. This training data is slightly larger than the English-Sepedi corpus with 862 159 sentences (23m English tokens and 19m Setwana tokens). The sizes of the En-Tn validation and test sets are similar to that of En-Nso.

3.2 Sepedi monolingual corpus

There is a limited amount of publicly available Sepedi monolingual text. A notable dataset is the NCHLT Sepedi Text Corpora [11] consisting of only 164 000 sentences and approximately 1.5 million tokens, based on South African government documents crawled from government websites. This paper makes use of additional monolingual Sepedi data acquired through web crawling [31]. Table 2 presents the dataset statistics. The dataset covers various domains including South African Government Cabinet meetings minutes, Curriculum Assessment Policy Statements, documents from the South African Revenue Services, and short stories from nalibali.org.

3.3 Preprocessing

We follow previous work in separating the global JW300 English-Sepedi test set [27] from the rest of the JW300 data, which is split into training and validation sets. We train BPE tokenizers separately on the English and Sepedi sides of the training data, with a maximum vocabulary of 10 000 for each languages. For backtranslation we found that it worked better to train the BPE tokenizer on the combination of the JW300 corpus and the Sepedi monolingual corpus than on the JW300 corpus only. Applying the JW300-only learnt BPE operations on the Sepedi monolingual corpus introduced between 10.80% and 14.80% *unk* tokens; this was in contrast to using learnt BPE operations based on the combination of the JW300 and Sepedi monolingual corpora which introduced 0.06% to 1.84% *unk* tokens.

For training purposes, we use the FairSeq toolkit [28], an extensible open-source neural machine translation package that is suitable for research purposes, for all the experiments.[1] FairSeq includes implementations of sequence-to-sequence models, LSTMs, convolutional models, and Transformers.

The model is trained for between 50 - 100 epochs or until convergence. All experiments are run on single Tesla V100-PCIE-16GB GPU nodes for 6 hours on average. We make use of 16-bit floating point operations [29] for faster training.

4 Backtranslation: Experiments and Results

We train an English to Sepedi baseline NMT model, following previous work [1]. The baseline configuration consists of 5 hidden layers and learning rate of 0.0003, using the Adam optimizer [20]. We use 4 attention heads for both the encoder and decoder, an embedding dimension of 256, feed-forward dimension of 1024, and a dropout rate of 0.3. We perform hyperparameter tuning to further optimize these settings.

A reverse NMT model (Sepedi to English) is trained based on the same optimised hyperparameters; this model is used to generate synthetic English sentences from monolingual Sepedi sentences. The combined parallel dataset and dataset generated with backtranslation is used to train further models.

In addition to the baseline model hyperparameter settings [24], we consider further combinations of hyperparameter settings focusing on the trade off be-

[1] Available at https://github.com/pytorch/fairseq

Table 3: English-Sepedi Translation Results for Backtranslation (Test Set BLEU Scores)

	JW300	**Autshumato**	**FLoRes**
Benchmark [24, 1, 18]	45.95	10.95	6.76
Baseline	45.69	9.45	6.45
Baseline+Hyperparameter Optimisation	**49.50**	11.63	6.51
Standard Backtranslation	45.71	12.23	8.10
Backtranslation+Noise	42.96	19.30	8.74
Backtranslation+Filtering	48.23	14.31	8.52
Backtranslation+Noise+Filtering	48.82	**19.45**	**9.17**

tween model capacity and regularization. For generating the synthetic backtranslation data we considered three decoding algorithms: beam search, sampling and top-k sampling (with k=10). In all the result reported here, beam search is used as it led to the best performance.

Additionally, we investigate applying noise [10] to the backtranslated sentences, making use of the noisy-text package.[2] The following noising operations are applied: delete words in the synthetic sentence (with probability of 0.1), replace words with a *MASK* token (with probability of 0.1), and swapping words within a range of 10 tokens.

4.1 Results

The backtranslation results are reported in Table 3.

Baseline Our baseline aimed to reproduce previous work that is used as benchmark using the same hyperparameters [1]. On the JW300 test set near-identical results are obtained [24]. On Autshumato the results are slightly lower [1], while on FLoRes the results are again very close to previously reported results [18].

Hyperparameter Tuning For subsequent experiments we performed hyperparameter tuning based on increasing the model capacity, in which we increase the number of attention heads from 4 to 8. This results in a BLEU score increase of +1.54 for the En-Nso direction. The second aspect is the model's regularization, whereby we lower the dropout rate to 0.1, and achieve a BLEU score increase of +2.77 for the En-Nso direction. The best result is due to a combination of both of these approaches, giving a BLEU score increase of +3.81 for the En-Nso direction.

We apply the same hyperparameters to train the reverse translation model *(Nso-En)*. Performance increase from 46.59 to 47.95 BLEU when both changes are applied, compared to the baseline.

[2] https://github.com/valentinmace/noisy-text

Backtranslation The baseline backtranslation results show an improvement of only +0.02 BLEU on the JW300 evaluation set; however the improvements are larger with +2.78 and +1.65 on the Autshumato and FLoRes evaluation sets, respectively.

Backtranslation with noise added during the generation of the synthetic sentences leads to a performance drop on the JW300 test set, but improvements on the other two evaluation sets. This suggests that the added noise makes the model more robust across domains. We also experimented with reducing the noise introduced by the backtranslated data by increasing the ratio of the base training data compared to the backtranslated data to 2:1. This led to a smaller increase on Autshumato (to 15.61 BLEU), but decreased performance on buth JW300 (44.49 BLEU) and FLoRes (7.86 BLEU).

A different approach to lessen translation quality deterioration in backtranslation is filtering. We investigate filtering the synthetic sentences based on the number of tokens. Results are reported for filtering out sentences with less than five tokens. We record an increase in the BLUE score of +2.52 on the JW300 evaluation set.

We also experimented with a higher threshold of ten tokens: The results are very similar, with slight increases on JW300 (48.29 BLEU) and Autshumato (14.68 BLUE) but a decrease on FLoRes (8.03 BLUE). The number of training sentences decreases from 1.36M (backtranslation baseline) to 955k (filtering token length 5) or 828k (filtering token length 10), compared to the bitext size of 620k.

We further combine applying noise to the generated synthetic sentences and filtering the sentences based on the number of tokens. The results are again reported for filtering sentences with less than 5 tokens. This yields the best performance among backtranslation models across all 3 test sets. We also experimented with a filtering length of 10 tokens. This leads to slightly worse performance on JW300 (47.42) and Autshumato (19.22), but slightly higher on FLoRes (9.33).

The final results therefore show that the addition of monolingual data with different domains to the bitext data improves the generalization of the NMT model. However, care needs to be taken to ensure the quality of the synthetic data utilized.

5 Word Replacement: Experiments and Results

We train word embeddings for Sepedi and Setswana from monolingual data using fastText [6].[3] A bilingual dictionary is created by aligning the embedding spaces. Setswana tokens in the English-Setswana parallel corpus are replaced with Sepedi words if they appear in the bilingual dictionary. The resulting pseudo-Sepedi corpus is combined with the base English–Sepedi corpus to train NMT models.

To train word embeddings of size d we experiment with two approaches [26]: Continuous Bag-of-Words (CBOW) and Skipgram. The CBOW architecture uses

[3] https://github.com/facebookresearch/fastText

Table 4: NMT results with word replacement based on CBOW vs. Skipgram Induced Dictionaries

Method	**JW300**	**FLoRes**	**Autshumato**
CBOW *(d = 300)*	30.88	5.10	8.19
CBOW *(d = 500)*	28.51	5.11	7.67
Skipgram *(d = 300)*	**31.97**	**6.20**	**8.81**
Skipgram *(d = 500)*	30.14	5.77	8.52

the combination of the vectors of the words surrounding of a target word (the context words) as context for predicting the target word. Skipgram on the other hand aims to predict the *context words* given the *target word* as input into the model.

The two embedding spaces are aligned using a supervised adversarial training approach. We start by identifying similar tokens between the Sepedi and Setswana vocabularies. These total 5 305 tokens, which are further split into train-test split of 4000 and 1 305 tokens. Word pairs between Setswana and Sepedi are added to the dictionary if they are each other's closest neighbours by using the CSLS similarity measure [40]. The mappings are done from the language with the smaller vocabulary size to ensure a one-to-one mapping. We also apply alignment refinements [22] to improve the mapping of rare words. Additionally, we investigate a filtering approach in which only synthetic sentences in which at least some percentage of tokens (1% to 20%) were replaced are kept.

5.1 Results

Embeddings for dictionary construction The first set of results for word replacement are given in Table 4. We compare the effect of the embedding dimension d using both CBOW and Skipgram. In preliminary experiments the BLEU scores resulting from the various dimension sizes generally increased up to size $d = 300$. The reported results all use a BPE vocabulary of 15k. We also experimented with a BPE size of 10k, but this led to unstable results, with very low BLEU scores (e.g. 4.3) for some of the embeddings models.

The skipgram model outperforms CBOW across all test sets and settings for constructing the bilingual dictionary. Models with an embedding size of 300 outperform using the larger embedding size of 500 across all test sets. However, this approach still consistently performs worse than the baseline.

Word replacement with filtering We aim to improve the word replacement approach further by filtering the pseudo-Sepedi corpus based on the percentage of Setswana tokens replaced in each sentence. Table 5 gives the translation results from training over corpora with various levels of filtering. The embedding model that gave the best results (Skipgram, d = 300, BPE vocabulary 15k) is used. Filtering leads to improvements across all the evaluation sets. The 10% filtered

Table 5: NMT results for word replacement with filtering, controlling the minimum proportion of replaced tokens per sentence

% Replaced	**JW300**	**FLoRes**	**Autshumato**
Baseline	**45.69**	6.45	9.45
No filter	31.97	6.20	8.81
1%	41.29	7.72	**11.30**
5%	40.03	6.08	8.03
10%	42.40	**7.96**	9.06
20%	41.90	7.16	8.81

corpus generally results in best performance across the evaluation sets, with the exception of 1% filtering on Autshumato. Compared to the baseline, improvements are obtained on the FLoRes and Autshumato test sets, but not on the in-domain JW300 test set. The reflects a similar pattern to the backtranslation results, although much smaller improvements are obtained from word replacement than from backtranslation.

6 Conclusion

We performed a comprehensive evaluation of two data augmentation methods, backtranslation and word replacement, for English to Sepedi neural machine translation. The results show that both backtranslation and word replacement improves generalisation over the NMT baseline, as evidenced by improvements in BLEU scores on the Autshumato and FLoRes test sets, which cover different domains than the JW300 training corpus. Compared to previous published results, our best backtranslation models obtained improvements of +3.13 on the JW300, +8.50 on Autshumato, and +2.41 on the FLoRes evaluation set. On the JW300 test set the best performance was however obtained by better hyperparameter tuning of the base model. The results from word replacement were also promising, although the higher levels of noise in the synthetic data creates challenges for this approach in low-resource setups. The choice of BPE vocabulary size, embedding model and embedding dimension had a large effect on the quality of the induced bilingual dictionary. Filtering the synthetic corpus was essential for obtaining good results. The results show smaller improvements over the benchmarks, with +0.35 on the Autshumato evaluation set and +1.20 on the FLoRes evaluation set. No improvements where obtained on the benchmark JW300 evaluation set.

Acknowledgements

This work is based on research supported in part by the National Research Foundation of South Africa (Grant Number: 129850) and the South African Centre for High Performance Computing.

References

1. Abbott, J., Martinus, L.: Benchmarking neural machine translation for Southern African languages. In: Proceedings of the 2019 Workshop on Widening NLP. pp. 98–101. Association for Computational Linguistics, Florence, Italy (Aug 2019), https://www.aclweb.org/anthology/W19-3632
2. Abbott, J.Z., Martinus, L.: Towards neural machine translation for african languages. CoRR **abs/1811.05467** (2018)
3. Agic, Z., Vulic, I.: Jw300: A wide-coverage parallel corpus for low-resource languages. In: Korhonen, A., Traum, D.R., Màrquez, L. (eds.) ACL (1). pp. 3204–3210. Association for Computational Linguistics (2019)
4. Artetxe, M., Labaka, G., Agirre, E., Cho, K.: Unsupervised neural machine translation. ArXiv **abs/1710.11041** (2018)
5. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. In: International Conference on Learning Representations (2015)
6. Bojanowski, P., Grave, E., Joulin, A., Mikolov, T.: Enriching word vectors with subword information. Transactions of the Association for Computational Linguistics **5**, 135–146 (2017), https://aclanthology.org/Q17-1010
7. Caswell, I., Kreutzer, J., Wang, L., Wahab, A., van Esch, D., Ulzii-Orshikh, N., Tapo, A.A., Subramani, N., Sokolov, A., Sikasote, C., Setyawan, M., Sarin, S., Samb, S., Sagot, B., Rivera, C., Gonzales, A.R., Papadimitriou, I., Osei, S., Suarez, P.O., Orife, I., Ogueji, K., Niyongabo, R.A., Nguyen, T.Q., Muller, M., Muller, A., Muhammad, S.H., Muhammad, N.F., Mnyakeni, A., Mirzakhalov, J., Matangira, T., Leong, C., Lawson, N., Kudugunta, S., Jernite, Y., Jenny, M., Firat, O., Dossou, B.F.P., Dlamini, S., de Silva, N., cCabuk Balli, S., Biderman, S.R., Battisti, A., Baruwa, A., Bapna, A., Baljekar, P.N., Azime, I.A., Awokoya, A., Ataman, D., Ahia, O., Ahia, O., Agrawal, S., Adeyemi, M.: Quality at a glance: An audit of web-crawled multilingual datasets. ArXiv **abs/2103.12028** (2021)
8. Cho, K., van Merriënboer, B., Bahdanau, D., Bengio, Y.: On the properties of neural machine translation: Encoder–decoder approaches. In: Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation. pp. 103–111. Association for Computational Linguistics, Doha, Qatar (Oct 2014). https://doi.org/10.3115/v1/W14-4012, https://aclanthology.org/W14-4012
9. Cho, K., van Merrienboer, B., Çaglar Gülçehre, Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using rnn encoder–decoder for statistical machine translation. In: Empirical Methods in Natural Language Processing (2014)
10. Edunov, S., Ott, M., Auli, M., Grangier, D.: Understanding back-translation at scale. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. pp. 489–500. Association for Computational Linguistics, Brussels, Belgium (Oct-Nov 2018). https://doi.org/10.18653/v1/D18-1045, https://aclanthology.org/D18-1045
11. Eiselen, R., Puttkammer, M.: Developing text resources for ten South African languages. In: Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14). pp. 3698–3703. European Language Resources Association (ELRA), Reykjavik, Iceland (May 2014)
12. Elmadani, K.N., Meyer, F., Buys, J.: University of cape town's wmt22 system: Multilingual machine translation for southern african languages. In: Proceedings of the Seventh Conference on Machine Translation (WMT). Association for Computational Linguistics (2022)

13. Emezue, C.C., Dossou, B.F.P.: MMTAfrica: Multilingual machine translation for African languages. In: Proceedings of the Sixth Conference on Machine Translation. pp. 398–411. Association for Computational Linguistics, Online (Nov 2021), https://aclanthology.org/2021.wmt-1.48
14. Fadaee, M., Bisazza, A., Monz, C.: Data augmentation for low-resource neural machine translation. In: Barzilay, R., Kan, M.Y. (eds.) ACL (2). pp. 567–573. Association for Computational Linguistics (2017)
15. Feng, S.Y., Gangal, V., Wei, J., Chandar, S., Vosoughi, S., Mitamura, T., Hovy, E.H.: A survey of data augmentation approaches for nlp. ArXiv **abs/2105.03075** (2021)
16. Gage, P.: A new algorithm for data compression. The C Users Journal archive **12**, 23–38 (1994)
17. Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M., Lempitsky, V.S.: Domain-adversarial training of neural networks. In: J. Mach. Learn. Res. (2016)
18. Goyal, N., Gao, C., Chaudhary, V., Chen, P.J., Wenzek, G., Ju, D., Krishnan, S., Ranzato, M., Guzmán, F., Fan, A.: The flores-101 evaluation benchmark for low-resource and multilingual machine translation. Transactions of the Association for Computational Linguistics **10**, 522–538 (2022)
19. Imankulova, A., Sato, T., Komachi, M.: Improving low-resource neural machine translation with filtered pseudo-parallel corpus. In: WAT@IJCNLP (2017)
20. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. CoRR **abs/1412.6980** (2015)
21. Lakew, S.M., Negri, M., Turchi, M.: Low resource neural machine translation: A benchmark for five african languages. CoRR **abs/2003.14402** (2020), https://arxiv.org/abs/2003.14402
22. Lample, G., Conneau, A., Ranzato, M., Denoyer, L., Jégou, H.: Word translation without parallel data. In: ICLR (Poster). OpenReview.net (2018)
23. Marivate, V., Sefara, T., Modupe, A.: Investigating an approach for low resource language dataset creation, curation and classification: Setswana and sepedi. In: Proceedings of the first workshop on Resources for African Indigenous Languages. pp. 15–20. European Language Resources Association (ELRA) (May 2020), https://www.aclweb.org/anthology/2020.rail-1.3
24. Martinus, L., Webster, J., Moonsamy, J., Jnr, M.S., Moosa, R., Fairon, R.: Neural machine translation for south africa's official languages. CoRR **abs/2005.06609** (2020)
25. McKellar, C.A.: Autshumato machine translation evaluation set (2019), https://hdl.handle.net/20.500.12185/506
26. Mikolov, T., Chen, K., Corrado, G.S., Dean, J.: Efficient estimation of word representations in vector space. In: ICLR (2013)
27. Orife, I., Kreutzer, J., Sibanda, B.K., Whitenack, D., Siminyu, K., Martinus, L., Ali, J.T., Abbott, J.Z., Marivate, V., KABENAMUALU, S.K., Meressa, M., Murhabazi, E., Ahia, O., Biljon, E.V., Ramkilowan, A., Akinfaderin, A., Oktem, A., Akin, W., Kioko, G., Degila, K., Kamper, H., Dossou, B.F.P., Emezue, C.C., Ogueji, K., Bashir, A.M.: Masakhane - machine translation for africa. ArXiv **abs/2003.11529** (2020)
28. Ott, M., Edunov, S., Baevski, A., Fan, A., Gross, S., Ng, N., Grangier, D., Auli, M.: fairseq: A fast, extensible toolkit for sequence modeling. In: Ammar, W., Louis, A., Mostafazadeh, N. (eds.) NAACL-HLT (Demonstrations). pp. 48–53. Association for Computational Linguistics (2019)

29. Ott, M., Edunov, S., Grangier, D., Auli, M.: Scaling neural machine translation. In: WMT (2018)
30. Papineni, K., Roukos, S., Ward, T., Zhu, W.J.: Bleu: a method for automatic evaluation of machine translation. In: Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics. pp. 311–318. Association for Computational Linguistics, Philadelphia, Pennsylvania, USA (Jul 2002). https://doi.org/10.3115/1073083.1073135, https://www.aclweb.org/anthology/P02-1040
31. Ralethe, S.: Sepedi monolingual data (2021)
32. Sennrich, R., Haddow, B., Birch, A.: 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). pp. 86–96. Association for Computational Linguistics, Berlin, Germany (Aug 2016). https://doi.org/10.18653/v1/P16-1009, https://www.aclweb.org/anthology/P16-1009
33. Shorten, C., Khoshgoftaar, T.M., Furht, B.: Text data augmentation for deep learning. Journal of Big Data **8** (2021)
34. Stahlberg, F.: Neural machine translation: A review. J. Artif. Intell. Res. **69**, 343–418 (2020)
35. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks. In: Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N.D., Weinberger, K.Q. (eds.) NIPS. pp. 3104–3112 (2014)
36. Tan, Z., Wang, S., Yang, Z., Chen, G., Huang, X., Sun, M., Liu, Y.: Neural machine translation: A review of methods, resources, and tools. AI Open **1**, 5–21 (2020). https://doi.org/https://doi.org/10.1016/j.aiopen.2020.11.001
37. Tiedemann, J.: Parallel data, tools and interfaces in opus. In: Chair), N.C.C., Choukri, K., Declerck, T., Dogan, M.U., Maegaard, B., Mariani, J., Odijk, J., Piperidis, S. (eds.) Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12). European Language Resources Association (ELRA), Istanbul, Turkey (may 2012)
38. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (eds.) Advances in Neural Information Processing Systems 30, p. 5998–6008. Curran Associates, Inc. (2017), https://papers.nips.cc/paper/7181-attention-is-all-you-need
39. Wang, X., Pham, H., Dai, Z., Neubig, G.: Switchout: an efficient data augmentation algorithm for neural machine translation. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. pp. 856–861. Association for Computational Linguistics, Brussels, Belgium (Oct-Nov 2018). https://doi.org/10.18653/v1/D18-1100, https://www.aclweb.org/anthology/D18-1100
40. Xia, M., Kong, X., Anastasopoulos, A., Neubig, G.: Generalized data augmentation for low-resource translation. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. pp. 5786–5796. Association for Computational Linguistics, Florence, Italy (Jul 2019). https://doi.org/10.18653/v1/P19-1579, https://www.aclweb.org/anthology/P19-1579
41. Zhang, B., Williams, P., Titov, I., Sennrich, R.: Improving massively multilingual neural machine translation and zero-shot translation. ArXiv **abs/2004.11867** (2020)
42. Zhang, J., Zong, C.: Exploiting source-side monolingual data in neural machine translation. In: Empirical Methods in Natural Language Processing (2016)