Back-Translation Sampling by Targeting Difficult Words in Neural Machine Translation

Marzieh Fadaee and **Christof Monz**
Informatics Institute, University of Amsterdam
Science Park 904, 1098 XH Amsterdam, The Netherlands
{m.fadaee,c.monz}@uva.nl

Abstract

Neural Machine Translation has achieved state-of-the-art performance for several language pairs using a combination of parallel and synthetic data. Synthetic data is often generated by back-translating sentences randomly sampled from monolingual data using a reverse translation model. While back-translation has been shown to be very effective in many cases, it is not entirely clear why. In this work, we explore different aspects of back-translation, and show that words with high prediction loss during training benefit most from the addition of synthetic data. We introduce several variations of sampling strategies targeting difficult-to-predict words using prediction losses and frequencies of words. In addition, we also target the contexts of difficult words and sample sentences that are similar in context. Experimental results for the WMT news translation task show that our method improves translation quality by up to 1.7 and 1.2 BLEU points over back-translation using random sampling for German→English and English→German, respectively.

1 Introduction

Neural machine translation (NMT) using a sequence-to-sequence model has achieved state-of-the-art performance for several language pairs (Bahdanau et al., 2015; Sutskever et al., 2014; Cho et al., 2014). The availability of large-scale training data for these sequence-to-sequence models is essential for achieving good translation quality.

Previous approaches have focused on leveraging monolingual data which is available in much larger quantities than parallel data (Lambert et al., 2011). Gulcehre et al. (2017) proposed two methods, shallow and deep fusion, for integrating a neural language model into the NMT system. They observe improvements by combining the scores of a neural language model trained on target monolingual data with the NMT system.

Sennrich et al. (2016a) proposed back-translation of monolingual target sentences to the source language and adding the synthetic sentences to the parallel data. In this approach a reverse model trained on parallel data is used to translate sentences from target-side monolingual data into the source language. This *synthetic* parallel data is then used in combination with the actual parallel data to re-train the model. This approach yields state-of-the-art results even when large parallel data are available and has become common practice in NMT (Sennrich et al., 2017; García-Martínez et al., 2017; Ha et al., 2017).

While back-translation has been shown to be very effective to improve translation quality, it is not exactly clear why it helps. Generally speaking, it mitigates the problem of overfitting and fluency by exploiting additional data in the target language. An important question in this context is how to select the monolingual data in the target language that is to be back-translated into the source language to optimally benefit translation quality. Pham et al. (2017) experimented with using domain adaptation methods to select monolingual data based on the cross-entropy between the monolingual data and in-domain corpus (Axelrod et al., 2015) but did not find any improvements over random sampling as originally proposed by Sennrich et al. (2016a). Earlier work has explored to what extent data selection of parallel corpora can benefit translation quality (Axelrod et al., 2011; van der Wees et al., 2017), but such selection techniques have not been investigated in the context of back-translation.

In this work, we explore different aspects of the back-translation method to gain a better understanding of its performance. Our analyses show that the quality of the synthetic data acquired with

Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 436–446
Brussels, Belgium, October 31 - November 4, 2018. ©2018 Association for Computational Linguistics

a reasonably good model has a small impact on the effectiveness of back-translation, but the ratio of synthetic to real training data plays a more important role. With a higher ratio, the model gets biased towards noises in synthetic data and unlearns the parameters completely. Our findings show that it is mostly words that are difficult to predict in the target language that benefit from additional back-translated data. These are the words with high prediction loss during training when the translation model converges. We further investigate these difficult words and explore alternatives to random sampling of sentences with a focus on increasing occurrences of such words.

Our proposed approach is twofold: identifying difficult words and sampling with the objective of increasing occurrences of these words, and identifying contexts where these words are difficult to predict and sample sentences similar to the difficult contexts. With targeted sampling of sentences for back-translation we achieve improvements of up to 1.7 BLEU points over back-translation using random sampling.

2 Back-Translation for NMT

In this section, we briefly review a sequence-to-sequence NMT system and describe our experimental settings. We then investigate different aspects and modeling challenges of integrating the back-translation method into the NMT pipeline.

2.1 Neural Machine Translation

The NMT system used for our experiments is an encoder-decoder network with recurrent architecture (Luong et al., 2015). For training the NMT system, two sequences of tokens, $X = \big[x_1, \ldots, x_n\big]$ and $Y = \big[y_1, \ldots, y_m\big]$, are given in the source and target language, respectively.

The source sequence is the input to the encoder which is a bidirectional long short-term memory network generating a representation $\mathbf{s}_n$. Using an attention mechanism (Bahdanau et al., 2015), the attentional hidden state is:

$$\widetilde{\mathbf{h}}_t = \tanh(\mathbf{W}_c[\mathbf{c}_t; \mathbf{h}_t])$$

where $\mathbf{h}_t$ is the target hidden state at time step t and $\mathbf{c}_t$ is the context vector which is a weighted average of $\mathbf{s}_n$.

The decoder predicts each target token y_t by computing the probability:

$$p(y_t|y_{<t}, \mathbf{s}_n) = \mathrm{softmax}(\mathbf{W}_o\widetilde{\mathbf{h}}_t)$$

For the token y_t, the conditional probability $p(y_t|y_{<t}, \mathbf{s}_n)$ during training quantifies the difficulty of predicting that token in the context $y_{<t}$. The prediction loss of token y_t is the negative log-likelihood of this probability.

During training on a parallel corpus $\mathbb{D}$, the cross-entropy objective function is defined as:

$$\mathcal{L} = \sum_{(X,Y)\in\mathbb{D}} \sum_{i=1}^{m} -\log p(y_i|y_{<i}, \mathbf{s}_n)$$

The objective of this function is to improve the model's estimation of predicting target words given the source sentence and the target context.

The model is trained end-to-end by minimizing the negative log likelihood of the target words.

2.2 Experimental Setup

For the translation experiments, we use English↔German WMT17 training data and report results on newstest 2014, 2015, 2016, and 2017 (Bojar et al., 2017).

As NMT system, we use a 2-layer attention-based encoder-decoder model implemented in OpenNMT (Klein et al., 2017) trained with embedding size 512, hidden dimension size 1024, and batch size 64. We pre-process the training data with Byte-Pair Encoding (BPE) using 32K merge operations (Sennrich et al., 2016b).

We compare the results to Sennrich et al. (2016a) by back-translating random sentences from the monolingual data and combine them with the parallel training data. We perform the random selection and re-training 3 times and report the averaged outcomes for the 3 models. In all experiments the sentence pairs are shuffled before each epoch.

We measure translation quality by single-reference case-sensitive BLEU (Papineni et al., 2002) computed with the `multi-bleu.perl` script from Moses.

2.3 Size of the Synthetic Data in Back-Translation

One selection criterion for using back-translation is the ratio of real to synthetic data. Sennrich et al. (2016a) showed that higher ratios of synthetic data leads to decreases in translation performance.

In order to investigate whether the improvements in translation performance increases with higher ratios of synthetic data, we perform three experiments with different sizes of synthetic data.

	Size	2014	2015	2016	2017
Baseline	4.5M	26.7	27.6	32.5	28.1
+ synthetic (1:1)	9M	28.7	29.7	36.3	30.8
+ synthetic (1:4)	23M	29.1	30.0	36.9	31.1
+ synthetic (1:10)	50M	22.8	23.6	29.2	23.9

Table 1: German→English translation quality (BLEU) of systems with different ratios of *real:syn* data.

	Size	2014	2015	2016	2017
Baseline	4.5M	21.2	23.3	28.0	22.4
+ synthetic tgt	9M	22.4	25.3	29.8	23.7
+ synthetic src	9M	24.0	26.0	30.7	24.8

Table 2: English→German translation quality (BLEU) of systems using forward and reverse models for generating synthetic data.

Figure 1 shows the perplexity as a function of training time for different sizes of synthetic data. One can see that all systems perform similarly in the beginning and converge after observing increasingly more training instances. However, the model with the ratio of (1:10) synthetic data gets increasingly biased towards the noisy data after 1M instances. Decreases in performance with more synthetic than real data is also inline with findings of Poncelas et al. (2018).

Comparing the systems using ratios of (1:1) and (1:4), we see that the latter achieves lower perplexity during training. Table 1 presents the performance of these systems on the German→English translation task. The BLEU scores show that the translation quality does not improve linearly with the size of the synthetic data. The model trained on (1:4) real to synthetic ratio of training data achieves the best results, however, the performance is close to the model trained on (1:1) training data.

2.4 Direction of Back-Translation

Adding monolingual data in the target language to the training data has the benefit of introducing new context and improving the fluency of the translation model. The automatically generated translations in the source language while being erroneous, introduce new context for the source words and will not affect the translation model significantly.

Monolingual data is available in large quantities for many languages. The decision of the direction of back-translation is subsequently not based on the monolingual data available, but on the advantage of having more fluent source or target sentences.

Lambert et al. (2011) show that adding synthetic source and real target data achieves improvements in traditional phrase-based machine translation (PBMT). Similarly in previous works in NMT, back-translation is performed on monolingual data in the target language.

We perform a small experiment to measure whether back-translating from source to target is also beneficial for improving translation quality. Table 2 shows that in both directions the performance of the translation system improves over the baseline. This is in contrast to the findings of Lambert et al. (2011) for PBMT systems where they show that using synthetic target data does not lead to improvements in translation quality.

Still, when adding monolingual data in the target language the BLEU scores are slightly higher than when using monolingual data in the source language. This indicates the moderate importance of having fluent sentences in the target language.

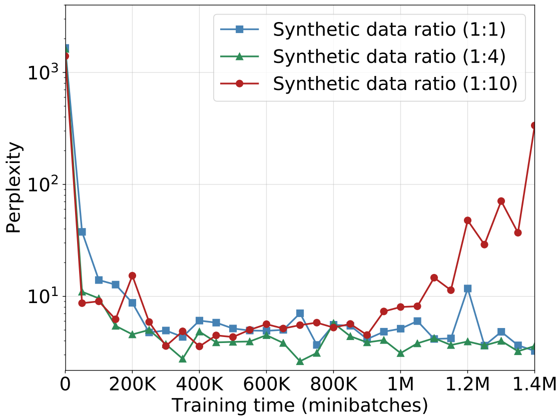

Figure 1: Training plots for systems with different ratios of $(real : syn)$ training data, showing perplexity on development set.

2.5 Quality of the Synthetic Data in Back-Translation

One selection criterion for back-translation is the quality of the synthetic data. Khayrallah and Koehn (2018) studied the effects of noise in the training data on a translation model and discovered that NMT models are less robust to many types of noise than PBMT models. In order for the NMT model to learn from the parallel data, the data should be fluent and close to the manually generated translations. However, automatically generating sentences using back-translation is not as accurate as manual translations.

	Size	2014	2015	2016	2017
Baseline	2.25M	24.3	24.9	29.5	25.6
+ synthetic	4.5M	26.0	26.9	32.2	27.5
+ ground truth	4.5M	26.7	27.6	32.5	28.1

Table 3: German→English translation quality (BLEU).

To investigate the *oracle gap* between the performance of manually generated and back-translated sentences, we perform a simple experiment using the existing parallel training data. In this experiment, we divide the parallel data into two parts, train the reverse model on the first half of the data and use this model to back-translate the second half. The manually translated sentences of the second half are considered as ground truth for the synthetic data.

Table 3 shows the BLEU scores of the experiments. As to be expected, re-training with additional parallel data yields higher performance than re-training with additional synthetic data. However, the differences between the BLEU scores of these two models are surprisingly small. This indicates that performing back-translation with a reasonably good reverse model already achieves results that are close to a system that uses additional manually translated data. This is inline with findings of Sennrich et al. (2016a) who observed that the same monolingual data translated with three translation systems of different quality and used in re-training the translation model yields similar results.

3 Back-Translation and Token Prediction

In the previous section, we observed that the reverse model used to back-translate achieves results comparable to manually translated sentences. Also, there is a limit in learning from synthetic data, and with more synthetic data the model unlearns its parameters completely.

In this section, we investigate the influence of the sampled sentences on the learning model. Fadaee et al. (2017) showed that targeting specific words during data augmentation improves the generation of these words in the right context. Specifically, adding synthetic data to the training data has an impact on the prediction probabilities of individual words. In this section, we further examine the effects of the back-translated synthetic data on the prediction of target tokens.

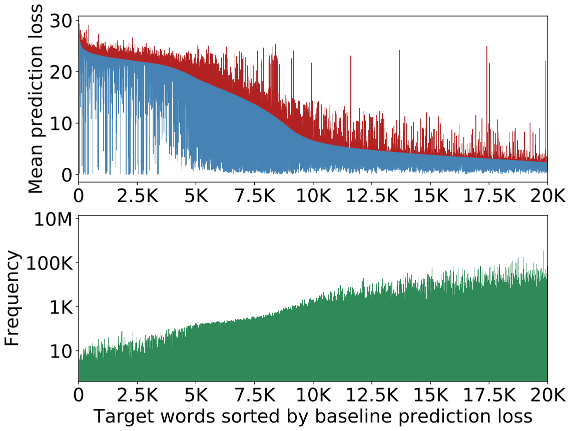

Figure 2: Top: Changes in mean token prediction loss after re-training with synthetic data sorted by mean prediction loss of the baseline system. Decreases and increases in values are marked blue and red, respectively. Bottom: Frequencies (log) of target tokens in the baseline training data.

As mentioned in Section 2.1, the objective function of training an NMT system is to minimize $\mathcal{L}$ by minimizing the prediction loss $-\log p(y_t|y_{<t}, \mathbf{s}_n)$ for each target token in the training data. The addition of monolingual data in the target language improves the estimation of the probability $p(Y)$ and consequently the model generates more fluent sentences.

Sennrich et al. (2016a) show that by using back-translation, the system with target-side monolingual data reaches a lower perplexity on the development set. This is expected since the domain of the monolingual data is similar to the domain of the development set. To investigate the model's accuracy independent from the domains of the data, we collect statistics of the target token prediction loss during training.

Figure 2 shows changes of token prediction loss when training converges and the weights are verging on being stable. The values are sorted by mean token prediction loss of the system trained on real parallel data.

We observe an effect similar to distributional smoothing (Chen and Goodman, 1996): While prediction loss increases slightly for most tokens, the largest decrease in loss occurs for tokens with high prediction loss values. This indicates that by randomly sampling sentences for back-translation, the model improves its estimation of tokens that were originally more difficult to predict, i.e., tokens that had a high prediction loss. Note that we compute the token prediction loss in just one pass over the training corpus with the final model and as a result it is not biased towards the order of the data.

This finding motivates us to further explore sampling criteria for back-translation that contribute considerably to the parameter estimation of the translation model. We propose that by oversampling sentences containing difficult-to-predict tokens we can maximize the impact of using the monolingual data. After translating sentences containing such tokens and including them in the training data, the model becomes more robust in predicting these tokens.

In the next two sections, we propose several methods of using the target token prediction loss to identify the most rewarding sentences for back-translating and re-training the translation model.

4 Targeted Sampling for Difficult Words

One of the main outcomes of using synthetic data is better estimation of words that were originally difficult to predict as measured by their high prediction losses during training. In this section, we propose three variations of how to identify these words and perform sampling to target these words.

Algorithm 1 Sampling for difficult words

```
Input: Difficult tokens $\mathfrak{D} = \{y_i\}_{i=1}^{D}$, monolingual corpus $\mathbb{M}$, number of required samples $N$
Output: Sampled sentences $S = \{S_i\}_{i=1}^{N}$ where each sentence $S_i$ is sampled from $\mathbb{M}$
procedure DIFFSAMPLING ($\mathfrak{D}, \mathbb{M}, N$):
    Initialize $S = \{\}$
    repeat
        Sample $S_c$ from $\mathbb{M}$
        for all tokens $y$ in $S_c$ do
            if $y \in \mathfrak{D}$ then
                Add $S_c$ to $S$
    until $|S| = N$
    return $S$
```

4.1 Token Frequency as a Feature of Difficulty

Figure 2 shows that the majority of tokens with high mean prediction losses have low frequencies in the training data. Additionally, the majority of decreases in prediction loss after adding synthetic sentence pairs to the training data occurs with less frequent tokens.

Note that these tokens are not necessarily *rare* in the traditional sense; in Figure 2 the $10k$ less frequent tokens in the target vocabulary benefit most from back-translated data.

Sampling new contexts from monolingual data provides context diversity proportional to the token frequencies and less frequent tokens benefit most from new contexts. Algorithm 1 presents this approach where the list of difficult tokens is defined as:

$$\mathfrak{D} = \{\forall y_i \in V_t \colon \mathit{freq}(y_i) < \eta\}$$

V_t is the target vocabulary and η is the frequency threshold for deciding on the difficulty of the token.

4.2 Difficult Words with High Mean Prediction Losses

In this approach, we use the mean losses to identify difficult-to-predict tokens. The mean prediction loss $\hat{\ell}(y)$ of token y during training is defined as follows:

$$\hat{\ell}(y) = \frac{1}{n_y} \sum_{n=1}^{N} \sum_{t=1}^{|Y^n|} -\log p(y_t^n | y_{<t}^n, \mathbf{s}_n) \delta(y_t^n, y)$$

where n_y is the number of times token y is observed during training, i.e., the token frequency of y, N is the number of sentences in the training data, $|Y^n|$ is the length of target sentence n, and $\delta(y_t^n, y)$ is the Kronecker delta function, which is 1 if $y_t^n = y$ and 0 otherwise.

By specifically providing more sentences for difficult words, we improve the model's estimation and decrease the model's uncertainty in prediction. During sampling from the monolingual data, we select sentences that contain difficult words.

Algorithm 1 presents this approach where the list of difficult tokens is defined as:

$$\mathfrak{D} = \{\forall y_i \in V_t \colon \hat{\ell}(y_i) > \mu\}$$

V_t is the vocabulary of the target language and μ is the threshold on the difficulty of the token.

System	De-En				En-De			
	test2014	test2015	test2016	test2017	test2014	test2015	test2016	test2017
BASELINE†	26.7	27.6	32.5	28.1	21.2	23.3	28.0	22.4
RANDOM†	28.7	29.7	36.3	30.8	24.0	26.0	30.7	24.8
FREQ	29.7	30.5	37.5	31.4	24.2	27.0	31.7	25.2
MEANPREDLOSS†	29.9	**30.9**	**37.8**	**32.1**	**24.7**	26.8	31.5	**25.5**
MEANPREDLOSS + STDPREDLOSS	**30.0**	30.9	37.7	31.9	24.1	26.9	31.0	25.3
PRESERVE PREDLOSS RATIO	29.8	30.9	37.4	31.6	24.5	**27.2**	**31.8**	25.5

Table 4: German↔English translation quality (BLEU). Experiments marked † are averaged over 3 runs. MEANPREDLOSS and FREQ are difficulty criteria based on mean token prediction loss and token frequency respectively. MEANPREDLOSS + STDPREDLOSS is experiments favoring tokens with skewed prediction losses. PRESERVE PREDLOSS RATIO preserves the ratio of the distribution of difficult contexts.

4.3 Difficult Words with Skewed Prediction Losses

By using the mean loss for target tokens as defined above, we do not discriminate between differences in prediction loss for occurrences in different contexts. This lack of discrimination can be problematic for tokens with high loss variations. For instance, there can be a token with ten occurrences, out of which two have high and eight have low prediction loss values.

We hypothesize that if a particular token is easier to predict in some contexts and harder in others, the sampling strategy should be context sensitive, allowing to target specific contexts in which a token has a high prediction loss. In order to distinguish between tokens with a skewed and tokens with a more uniform prediction loss distribution, we use both mean and standard deviation of token prediction losses to identify difficult tokens.

Algorithm 1 formalizes this approach where the list of the difficult tokens is defined as:

$$\mathfrak{D} = \{\forall y_i \in V_t \colon \hat{\ell}(y_i) > \mu \wedge \sigma(\ell(y_i)) > \rho\}$$

$\hat{\ell}(y_i)$ is the mean and $\sigma(\ell(y_i))$ is the standard deviation of prediction loss of token y_i, V_t is the vocabulary list of the target language, and μ and ρ are the thresholds for deciding on the difficulty of the token.

4.4 Preserving Sampling Ratio of Difficult Occurrences

Above we examined the mean of prediction loss for each token over all occurrences, in order to identify difficult-to-predict tokens. However, the uncertainty of the model in predicting a difficult token varies for different occurrences of the token: one token can be easy to predict in one context, and hard in another. While the sampling step in the previous approaches targets these tokens, it does not ensure that the distribution of sampled sentences is similar to the distribution of problematic tokens in difficult contexts.

To address this issue, we propose an approach where we target the number of times a token occurs in difficult-to-predict contexts and sample sentences accordingly, thereby ensuring the same ratio as the distribution of difficult contexts. If token y_1 is difficult to predict in two contexts and token y_2 is difficult to predict in four contexts, the number of sampled sentences containing y_2 is double the number of sampled sentences containing y_1. Algorithm 2 formalizes this approach.

Algorithm 2 Sampling with ratio preservation

```
Input: Difficult tokens and the corresponding sentences in the bitext $\mathfrak{D} = \{y_t, Y_{y_t} = [y_1, \ldots, y_t, \ldots, y_m]\}$, monolingual corpus $\mathbb{M}$, number of required samples $N$
Output: Sampled sentences $S = \{S_i\}_{i=1}^{N}$ where each sentence $S_i$ is sampled from $\mathbb{M}$
procedure PREDLOSSRATIOSAMPLING($\mathfrak{D}, \mathbb{M}, N$):
    Initialize $S = \{\}$
    $H(y_t) = \frac{N \times |(y_t, \cdot) \in \mathfrak{D}|}{|(y_\cdot, \cdot) \in \mathfrak{D}|}$
    repeat
        Sample $S_c$ from $\mathbb{M}$
        for all tokens $y$ in $S_c$ do
            if $|y \in S| < H(y)$ then
                Add $S_c$ to $S$
    until $|S| = N$
    return $S$
```

4.5 Results

We measure the translation quality of various models for German→English and English→German translation tasks. The results of the translation experiments are presented in Table 4. As baseline we compare our approach to Sennrich et al. (2016a). For all experiments we sample and back-translate sentences from the monolingual data, keeping a one-to-one ratio of back-translated versus original data (1:1).

We set the hyperparameters μ, ρ, and η to 5, 10, and 5000 respectively. The values of the hyperparameters are chosen on a small sample of the parallel data based on the token loss distribution.

As expected using random sampling for back-translation improves the translation quality over the baseline. However, all targeted sampling variants in turn outperform random sampling. Specifically, the best performing model for German→English, MEANPREDLOSS, uses the mean of prediction loss for the target vocabulary to oversample sentences including these tokens.

For the English→German experiments we obtain the best translation performance when we preserve the prediction loss ratio during sampling.

We also observe that even though the model targeting tokens with skewed prediction loss distributions (MEANPREDLOSS + STDPREDLOSS) improves over random selection of sentences, it does not outperform the model using only mean prediction losses.

5 Context-Aware Targeted Sampling

In the previous section, we proposed methods for identifying difficult-to-predict tokens and performed targeted sampling from monolingual data. While the objective was to increase the occurrences of difficult tokens, we ignored the context of these tokens in the sampled sentences.

Arguably, if a word is difficult to predict in a given context, providing more examples of the same or similar context can aid the learning process. In this section, we focus on the context of difficult-to-predict words and aim to sample sentences that are similar to the corresponding difficult context.

The general algorithm is described in Algorithm 3. In the following sections, we discuss different definitions of the local context (line 7 and line 9) and similarity measures (line 10) in this algorithm, and report the results.

5.1 Definition of Local Context

Prediction loss is a function of the source sentence and the target context. One reason that a token has high prediction loss is that the occurrence of the word is a deviation from what occurs more frequently in other occurrences of the context in the training data. This indicates an infrequent event, in particular a rare sense of the word, a domain that is different from other occurrences of the word, or an idiomatic expression.

source	wer glaube, dass das ende, sobald sie in Deutschland ank|ä|men, ir|re, erzählt **B**|**ahr**.
reference	if you think that this stops as soon as they arrive in Germany, you'd be wrong, says **B**|**ahr**.
NMT output	who believe that the end, as soon as they go to Germany, tells **B**|**risk**.

Table 5: An example from the synthetic data where the word *B|ahr* is incorrectly translated to *B|risk*. Subword unit boundaries are marked with '|'.

We identify *pairs* of tokens and sentences from parallel data where in each pair the NMT model assigns high prediction loss to the token in the given context. Note that a token can occur several times in this list, since it can be considered as difficult-to-predict in different sentences.

We propose two approaches to define the local context of the difficult token:

Neighboring tokens A straightforward way is to use positional context: tokens that precede and follow the target token, typically in a window of w tokens to each side. For sentence S containing a difficult token at index i the *context* function in Algorithm 3 is:

$$context(S, i) = [S^{i-w}, \ldots, S^{i-1}, S^{i+1}, \ldots, S^{i+w}]$$

where S^j is the token at index j in sentence S.

Sentence from bitext containing difficult token:
He attended **Stan**|ford University, where he double maj|ored in Spanish and History.
Sampled sentences from monolingual data:
– The group is headed by Aar|on K|ush|ner, a **Stan**|ford University gradu|ate who formerly headed a gre|eting card company. – Ford just opened a new R&D center near **Stan**|ford University, a hot|bed of such technological research. – Joe Grund|fest, a professor and a colleague at **Stan**|ford Law School, outlines four reasons why the path to the IP|O has become so steep for asp|iring companies.

Table 6: Results of targeted sampling for the difficult subword unit *'Stan'*.

Neighboring subword units In our analysis of prediction loss during training, we observe that several tokens that are difficult to predict are indeed subword units. Current state-of-the-art NMT systems apply BPE to the training data to address large vocabulary challenges (Sennrich et al., 2016b).

By using BPE the model generalizes common subword units towards what is more frequent in the training data. This is inherently useful since it allows for better learning of less frequent words. However, a side effect of this approach is that at times the model generates subword units that are not linked to any words in the source sentence.

As an example, in Table 5 German source and English reference translation show this problem. The word *B|ahr* consisting of two subword units is incorrectly translated into *B|risk*.

We address the insufficiency of the context for subword units with high prediction losses by targeting these tokens in sentence sampling.

Algorithm 3 formalizes this approach in sampling sentences from the monolingual data. For a sentence S containing a difficult subword at index i, the context function is defined as:

$$context(S, i) = [\ldots, S^{i-1}, S^{i+1}, \ldots]$$

where every token S^j in the local context is a subword unit and part of the same word as S^i. Table 6 presents examples of sampled sentences for the difficult subword unit *'Stan'*.

Sentence from bitext containing difficult word:
Bud|dy Hol|ly was part of the first group induc|ted into the **Rock** and R|oll Hall of F|ame on its formation in 1986.
Sampled sentences from monolingual data:
– A 2008 **Rock** and R|oll Hall of F|ame induc|t|ee, Mad|onna is ran|ked by the Gu|inn|ess Book of World Rec|ords as the top-selling recording artist of all time. – The **Rock** and R|oll Hall of Fam|ers gave birth to the California rock sound. – The winners were chosen by 500 voters, mostly musicians and other music industry veter|ans, who belong to the **Rock** and R|oll Hall of F|ame Foundation.

Table 7: Results of context-aware targeted sampling for the difficult token *'Rock'*

5.2 Similarity of the Local Contexts

In context-aware targeted sampling, we compare the context of a sampled sentence and the difficult context in the parallel data and select the sentence if they are *similar*. In the following, we propose two approaches for quantifying the similarities.

Algorithm 3 Sampling with context

```
Input: Difficult tokens and the corresponding sentences in the bitext $\mathfrak{D} = \{y_t, Y_{y_t} = [y_1, \ldots, y_t, \ldots, y_m]\}$, monolingual corpus $\mathbb{M}$, context function $context$, number of required samples $N$, similarity threshold $s$
Output: Sampled sentences $S = \{S_i\}_{i=1}^{N}$ where each sentence $S_i$ is sampled from $\mathbb{M}$
procedure CNTXTSAMPLING($\mathfrak{D}, \mathbb{M}, context, N, s$):
    Initialize $S = \{\}$
    repeat
        Sample $S_c$ from $\mathbb{M}$
        for all tokens $y_t$ in $S_c$ do
            if $y_t \in \mathfrak{D}$ then
                $C_m \leftarrow context(S_c, \text{indxof}(S_c, y_t))$
                for all $Y_{y_t}$ do
                    $C_p \leftarrow context(Y_{y_t}, \text{indxof}(Y_{y_t}, y_t))$

                    if Sim$(C_m, C_p) > s$ then
                        Add $S_c$ to $S$
    until $|S| = N$
    return $S$
```

Matching the local context In this approach we aim to sample sentences containing the difficult token, matching the exact context to the problematic context. By sampling sentences that match in a local window with the problematic context and differ in the rest of the sentence, we have more instances of the difficult token for training.

Algorithm 3 formalizes this approach where the similarity function is defined as:

$$\text{Sim}(C_m, C_p) = \frac{1}{c}\sum_{i=1}^{c} \delta(C_m^i, C_p^i)$$

C_m and C_p are the contexts of the sentences from monolingual and parallel data respectively, and c is the number of tokens in the contexts.

Word representations Another approach to sampling sentences that are similar to the problematic context is to weaken the matching assumption. Acquiring sentences that are similar in subject and not match the exact context words allows for lexical diversity in the training data. We use embeddings obtained by training the Skipgram model (Mikolov et al., 2013) on monolingual data to calculate the similarity of the two contexts.

For this approach we define the similarity function in Algorithm 3 as:

$$\text{Sim}(C_m, C_p) = cos(\mathbf{v}(C_m), \mathbf{v}(C_p))$$

where $\mathbf{v}(C_m)$ and $\mathbf{v}(C_p)$ are the averaged embeddings of the tokens in the contexts.

			De-En				En-De			
System			test2014	test2015	test2016	test2017	test2014	test2015	test2016	test2017
BASELINE †			26.7	27.6	32.5	28.1	21.2	23.3	28.0	22.4
RANDOM †			28.7	29.7	36.3	30.8	24.0	26.0	30.7	24.8
Difficulty criterion	**Context**	**Similarity**								
FREQ	TOKENS	EMB	30.0	30.8	37.6	31.7	24.4	26.3	31.5	25.6
PREDLOSS	SWORDS	MATCH	29.1	30.1	36.9	31.0	23.8	26.2	28.8	23.2
PREDLOSS	TOKENS	MATCH	29.7	30.6	37.6	31.8	24.3	27.4	31.6	25.5
PREDLOSS	TOKENS	EMB	29.9	30.8	37.7	31.9	24.5	**27.5**	31.7	25.6
PREDLOSS	SENTENCE	EMB	24.9	25.5	30.1	26.2	22.0	24.6	27.9	22.5
MEANPREDLOSS	TOKENS	EMB	**30.2**	**31.4**	**37.9**	**32.2**	**24.4**	27.2	**31.8**	**25.6**

Table 8: German↔English translation quality (BLEU). Experiments marked † are averaged over 3 runs. PREDLOSS is the contextual prediction loss and MEANPREDLOSS is the average loss. TOKEN and SWORD are context selection definitions from neighboring tokens and subword units respectively. Note that token includes both subword units and full words. EMB is computing context similarities with token embeddings and MATCH is comparing the context tokens.

Table 7 presents examples of sampled sentences for the difficult word *rock*. In this example, the context where the word *'Rock'* has high prediction loss is about the *music genre* and not the most prominent sense of the word, *stone*. Sampling sentences that contain this word in this particular context provides an additional signal for the translation model to improve parameter estimation.

5.3 Results

The results of the translation experiments are given in Table 8. In these experiments, we set the hyperparameters s and w to 0.75 and 4, respectively. Comparing the experiments with different similarity measures, MATCH and EMB, we observe that in all test sets we achieve the best results when using word embeddings. This indicate that for targeted sampling, it is more beneficial to have diversity in the context of difficult words as opposed to having the exact ngrams.

When using embeddings as the similarity measure, it is worth noting that with a context of size 4 the model performs very well but fails when we increase the window size to include the whole sentence.

The experiments focusing on subword units (SWORD) achieve improvements over the baselines, however they perform slightly worse than the experiments targeting tokens (TOKEN).

The best BLEU scores are obtained with the mean of prediction loss as difficulty criterion (MEANPREDLOSS) and using word representations to identify the most similar contexts. We observe that summarizing the distribution of the prediction losses by its mean is more beneficial than using individual losses. Our results motivate further explorations of using context for targeted sampling sentences for back-translation.

6 Conclusion

In this paper we investigated the effective method of back-translation for NMT and explored alternatives to select the monolingual data in the target language that is to be back-translated into the source language to improve translation quality.

Our findings showed that words with high prediction losses in the target language benefit most from additional back-translated data.

As an alternative to random sampling, we proposed targeted sampling and specifically targeted words that are difficult to predict. Moreover, we used the contexts of the difficult words by incorporating context similarities as a feature to sample sentences for back-translation. We discovered that using the prediction loss to identify weaknesses of the translation model and providing additional synthetic data targeting these shortcomings improved the translation quality in German↔English by up to 1.7 BLEU points.

Acknowledgments

This research was funded in part by the Netherlands Organization for Scientific Research (NWO) under project numbers 639.022.213 and 612.001.218. We also thank NVIDIA for their hardware support and the anonymous reviewers for their helpful comments.

References

Amittai Axelrod, Xiaodong He, and Jianfeng Gao. 2011. Domain adaptation via pseudo in-domain data selection. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 355–362. Association for Computational Linguistics.

Amittai Axelrod, Yogarshi Vyas, Marianna Martindale, Marine Carpuat, and Johns Hopkins. 2015. Class-based n-gram language difference models for data selection. In *IWSLT (International Workshop on Spoken Language Translation)*, pages 180–187.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Shujian Huang, Matthias Huck, Philipp Koehn, Qun Liu, Varvara Logacheva, Christof Monz, Matteo Negri, Matt Post, Raphael Rubino, Lucia Specia, and Marco Turchi. 2017. Findings of the 2017 conference on machine translation (wmt17). In *Proceedings of the Second Conference on Machine Translation, Volume 2: Shared Task Papers*, pages 169–214, Copenhagen, Denmark. Association for Computational Linguistics.

Stanley F Chen and Joshua Goodman. 1996. An empirical study of smoothing techniques for language modeling. In *Proceedings of the 34th annual meeting on Association for Computational Linguistics*, pages 310–318. Association for Computational Linguistics.

Kyunghyun Cho, B van Merrienboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the properties of neural machine translation: Encoder-decoder approaches. In *Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation (SSST-8), 2014*.

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.

Mercedes García-Martínez, Ozan Caglayan, Walid Aransa, Adrien Bardet, Fethi Bougares, and Loïc Barrault. 2017. Lium machine translation systems for wmt17 news translation task. In *Proceedings of the Second Conference on Machine Translation*, pages 288–295, Copenhagen, Denmark. Association for Computational Linguistics.

Caglar Gulcehre, Orhan Firat, Kelvin Xu, Kyunghyun Cho, and Yoshua Bengio. 2017. On integrating a language model into neural machine translation. *Computer Speech & Language*, 45:137–148.

Thanh-Le. Ha, Jan Niehues, and Alexander Waibel. 2017. Effective Strategies in Zero-Shot Neural Machine Translation. *ArXiv e-prints*.

Huda Khayrallah and Philipp Koehn. 2018. On the impact of various types of noise on neural machine translation. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pages 74–83. Association for Computational Linguistics.

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

Patrik Lambert, Holger Schwenk, Christophe Servan, and Sadaf Abdul-Rauf. 2011. Investigations on translation model adaptation using monolingual data. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, WMT '11, pages 284–293, Stroudsburg, PA, USA. Association for Computational Linguistics.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421, Lisbon, Portugal. Association for Computational Linguistics.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

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

Ngoc-Quan Pham, Jan Niehues, Thanh-Le Ha, Eunah Cho, Matthias Sperber, and Alexander Waibel. 2017. The karlsruhe institute of technology systems for the news translation task in wmt 2017. In *Proceedings of the Second Conference on Machine Translation*, pages 366–373.

Alberto Poncelas, Dimitar Shterionov, Andy Way, Gideon Maillette de Buy Wenniger, and Peyman Passban. 2018. Investigating Backtranslation in Neural Machine Translation. *ArXiv e-prints*.

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. 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.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 3104–3112. Curran Associates, Inc.

Marlies van der Wees, Arianna Bisazza, and Christof Monz. 2017. Dynamic data selection for neural machine translation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1400–1410, Copenhagen, Denmark. Association for Computational Linguistics.