PhraseOut: A Code Mixed Data Augmentation Method for Multilingual Neural Machine Translation

Binu Jasim
IIIT Hyderabad
binujasim.t
@research.iiit.ac.in

Vinay P Namboodiri
IIT Kanpur
vinaypn
@cse.iitk.ac.in

C.V. Jawahar
IIIT Hyderabad
jawahar
@iiit.ac.in

Abstract

Data Augmentation methods for Neural Machine Translation (NMT) such as back-translation (BT) and self-training (ST) are quite popular. In a multilingual NMT system, simply copying monolingual source sentences to the target (Copying) is an effective data augmentation method. Back-translation augments parallel data by translating monolingual sentences in the target side to source language. In this work we propose to use a partial back-translation method in a multilingual setting. Instead of translating the entire monolingual target sentence back into the source language, we replace selected high confidence phrases only and keep the rest of the words in the target language itself. (We call this method PhraseOut). Our experiments on low resource multilingual translation models show that PhraseOut gives reasonable improvements over the existing data augmentation methods.

1 Introduction

Data augmentation methods are popular in the field of computer vision. For example, it is a common practice to obtain extra training data by flipping and cropping images. Consistency regularization refers to the idea that a model should output the same label to an augmented example as the original example which in turn encourages distributional smoothness in the model (Berthelot et al., 2019). Consistency regularization has been used to obtain state of the art results in automatic speech recognition (S. Park et al., 2019) by randomly striking out horizontal and vertical portions of speech spectrogram. But for textual data, which is discrete, consistency regularization techniques are not easily applicable since changing a single word could change the entire meaning of a sentence. This work proposes a data augmentation method for textual data which doesn't change its original meaning and encourage consistency regularization.

Some of the existing approaches to data augmentation in Neural Machine Translation (NMT) are based on word replacements such as word dropout (Sennrich et al., 2016a; Gal and Ghahramani, 2016). A recent approach, termed SwitchOut (Wang et al., 2018) claims that randomly replacing words in the source and target sentences by words uniformly sampled from the respective vocabularies can improve neural machine translation. There are also attempts to inject artificial noise in the clean data according to the distribution of types of actual noise to make NMT systems more robust (Vaibhav et al., 2019).

Another approach to data augmentation is to make use of monolingual data. The most popular example is back-translation (Sennrich et al., 2016b) where a reverse translation model is employed to translate large amounts of target monolingual data back into source language. This (noisy) source and original target pair is added as additional parallel data and is shown to be useful while attempting to obtain state of the art results in machine translation (Edunov et al., 2018).

Multilingual NMT whereby sharing a single model between several languages has become a standard paradigm in NMT including industrial applications such as Google Translate (Johnson et al., 2017). Multilingual NMT has several advantages over training individual translation models for each language pair including faster inference, enabling zero shot machine translation and superior performance (Aharoni et al., 2019). A multilingual NMT model is particularly beneficial when training to translate low resource languages (Neubig and Hu, 2018).

The main contribution of this work is to show that a phrase based data augmentation strategy consistently provides improvement especially in low-resource language settings. We show that this is also useful in code-mixed translation settings.

2 Existing Data Augmentation Methods for NMT

Let $(\mathcal{X}, \mathcal{Y})$ denote the available parallel corpus. Our task is to find augmented parallel sentences $(\hat{x}, \hat{y})$ from the same distribution where the parallel data is sampled from.

After acquiring synthetic parallel data, it is appended to the available parallel data. An encoder-decoder based NMT model is trained to maximize the usual MLE objective as follows:

$$\sum_{n=1}^{N} \sum_{t=1}^{T^{(n)}} \log p_\theta(y_t^{(n)} | \mathbf{x}^{(n)}, y_{<t}^{(n)}) \quad (1)$$

Below we discuss several existing data augmentation techniques to generate synthetic parallel data.

Copying Monolingual Data (Copying): The monolingual target sentences are copied as the source sentence as well to create a synthetic parallel corpus to train (Currey et al., 2017; Burlot and Yvon, 2018). This is shown to improve the target side language fluency as it could learn from large amounts of monolingual sentences. This method is suitable for a multi-lingual setting as the encoder has to deal with sentences in more than one language. Note that we can combine word dropout with Copying.

Word Dropout: Dropping words from the source as well as target sentences (with a probability, say 0.1) has shown to improve the performance of NMT systems (Sennrich et al., 2016a). This could help learn the sentence representation better as in a denoising autoencoder (Vincent et al., 2008).

SwitchOut: This method proposes to replace a word (with a fixed probability, similar to the dropout probability) with another word from the vocabulary, chosen uniformly (Wang et al., 2018). SwitchOut is applied to both the source and the target sentences. Note that SwitchOut doesn't make use of any additional monolingual data. It performs augmentations only on the parallel data.

The paper report high variance in performance, because of which they run experiments 7 times and report the median performance. The performance was shown to be better than word dropout. Yet we find that data augmentation by Copying is much better than SwitchOut. Hence if extra monolingual data is available, then SwitchOut is of no practical significance.

Back Translation (BT): Along with the original NMT system, maintain a translation system in the reverse direction. Then each monolingual sentence is translated into the source sentence. This noisy source and good target pair can be used as a synthetic training data (Sennrich et al., 2016b). This has been shown to improve machine translation significantly to achieve new state-of-the-art results (Edunov et al., 2018). Yet Back Translation is of limited benefit if the initial reverse model is of poor quality (Wang et al., 2018).

Self Training (ST) Forward translate monolingual data using the same model to obtain target sentence. This source and noisy target pair can be used as synthetic augmented data along with parallel data (He et al., 2020). Self training gives similar performance improvements as BT.

In this work we don't compare against ST since it uses monolingual data in the source side while BT as well as other methods use monolingual data in the target side.

3 PhraseOut

We propose to use a partial back-translation technique in a multi-lingual setting. We do the augmentation only in the source side and keep the target sentence as it is similar to back-translation. The augmented sentence $\hat{y}$ is obtained by replacing a randomly chosen phrase from the target sentence y with a source language phrase. Hence we hope that both y and $\hat{y}$ are close in the semantic space and hence the distributional smoothness is maintained. This augmented sentence is copied as the source sentence.

PhraseOut is described in Algorithm 1. Using the available parallel data, we learn a phrase mapping table. A phrase alignment tool like `mgiza`[1] can be used for this. Next we augment target monolingual sentences if phrases in that sentence is present in the phrase table and replace that phrase with the corresponding source phrase. This essentially creates a code mixed source sentence. We hope that this could help bring the word embedding of similar words in different languages to be aligned. The augmented data generated by PhraseOut is concatenated with the original parallel data for further training or finetuning.

[1] https://github.com/moses-smt/mgiza

initialization;
$(\mathcal{X}, \mathcal{Y})$: Available parallel data ;
$\tilde{\mathcal{Y}}$: Monolingual data in the target domain ;
Learn a phrase table $\mathcal{P}$ using $(\mathcal{X}, \mathcal{Y})$;
foreach *sentence* $\tilde{\mathbf{y}}$ *in* $\tilde{\mathcal{Y}}$ **do**
  Let $\mathcal{N}$ denote all $ngrams$ in $\tilde{\mathbf{y}}$ (n<=4) ;
  foreach $\tilde{n}_y \in \mathcal{N}$ *chosen randomly* **do**
    if $\tilde{n}_y \in \mathcal{P}$ **then**
      Find corresponding source phrase $\tilde{n}_x$ from $\mathcal{P}$;
      $\tilde{\mathbf{x}} \leftarrow$ Replace $\tilde{n}_y$ with $\tilde{n}_x$ in $\tilde{\mathbf{y}}$
      Append $(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})$ to $(\mathcal{X}, \mathcal{Y})$;
      break
    end
  end
end

Algorithm 1: PhraseOut is a data augmentation method suited for a multilingual neural machine translation

4 Experimental Setup

4.1 Datasets

We experiment on *many to one* multilingual translation from Indian languages to English. We use the WAT 2018 dataset (Nakazawa et al., 2018) for our experiments. It has English translations of several Indian languages. We chose Hindi (hi), Bengali (bn), Malayalam (ml), Tamil (ta) and Telugu (te). The training data size is shown in 1

We use English sentences from the book corpus, a subset of the IIT-B dataset (Kunchukuttan et al., 2018) as our monolingual corpus.

We use moses tokenizer [2] to tokenize English and Indic nlp library [3] for tokenizing Indian languages.

	Train Size	Dev Size	Test Size
bn-en	362,240	1250	1750
hi-en	125,953	1500	2000
ml-en	395,047	1500	2000
ta-en	66,537	1500	2000
te-en	68,573	1500	2000
iitb	-	-	2507

Table 1: Dataset Description

[2] https://github.com/moses-smt/mosesdecoder/blob/master/scripts/tokenizer/tokenizer.perl

[3] https://github.com/anoopkunchukuttan/indic_nlp_library

4.2 Phrase Table Generation

We use Moses (Koehn et al., 2007) to learn phrase tables. The phrase alignment is learned by mgiza and the phrase tables are generated by moses [4]. Bad phrase table entries are removed by filtering by a probability threshold to ensures that good quality phrase translations are extracted out. In order to filter, we multiply the phrase translation probability in both directions and lexical weighting probability (Koehn et al., 2003) in both the directions together and keep the entry only if the multiplied probability is above $1e-12$. A snippet of such a phrase table is shown in 2.

Peter speaking softly	पीटर धीर बोले
Peter taught me	पीटर ने मुझे सिखाया
Peter	पीटर
Petersburg , Russia	पीटर्सबर्ग , रूस
Petersburg ,	पीटर्सबर्ग
Peth .	पेठ ।
Peth	पेठ
Petition ?	याचीका ?
Petition	याचीका

Table 2: A snippet from the phrase table learned using the parallel corpus

4.3 Models and Experimental Procedures

We use the transformer architecture from the fairseq framework [5]. and take an ensemble of last 10 checkpoints for testing.

We use SentencePiece (Kudo, 2018) subword tokenization. A joint subword vocabulary of 16,000 is used for the source side Indian languages and 8,000 is used for the target side English.

5 Results

We provide evaluations that compare several mono lingual augmentation methods, the effect of monolingual data size and the comparison with back-translation.

5.1 Copying vs PhraseOut

We use 200K monolingual sentences and compare mono-lingual augmentation (Copying) against PhraseOut. Note that PhraseOut is applied to augment all 5 language pairs. As shown in the table, PhraseOut gives around 1 BLEU point improvement Copying.

[4] http://www.statmt.org/moses/

[5] https://github.com/pytorch/fairseq

Lang	Baseline	Copying	Switch Out	PhraseOut
wat-hi	30.30	31.14	30.51	**32.06**
wat-bn	20.39	20.86	20.44	**21.07**
wat-ml	17.68	19.04	17.66	**20.62**
wat-ta	20.99	21.60	21.59	**21.64**
wat-te	27.74	28.44	28.36	**29.76**
iitb-hi	8.70	9.52	9.32	**9.90**

Table 3: Results on the WAT 2018 Test Set (Tokenized BLEU score)

5.2 Effect of Monolingual Data Size

We vary monolingual data size from 50K, 100K, 200K to 500K used for PhraseOut. The results on the IITB test set for the Hindi to English translation is shown below.

Size	Baseline	50K	100K	200K	500K
iitb-hi	8.70	9.25	9.68	9.90	10.09

Table 4: Results on IITB Test Set: Monolingual data size vs BLEU score

As shown in the table, the performance of PhraseOut increases with more monolingual data, but the improvement becomes lesser as monolingual data size is further increased.

5.3 Back Translation for Data Augmentation

Back-translation requires a reasonably good model to begin with, since generating too poor synthetic sentences could even deteriorate performance. With the amount of training data we use, BT with NMT doesn't produce good synthetic source sentences. Hence we train SMT (moses) in the reverse direction (i.e. English to Hindi) using the WAT2018 hi-en parallel data and agument 50K back-translated monolingual sentences to the parallel data. We obtain a BLEU score of 8.78 which is only slightly better than the baseline.

5.4 Qualitative Analysis: Translation of Code Mixed Text

Code mixed text is abundant in social media, especially in non-native English speaking countries such as India. A qualitative comparison of translation of a code mixed text is shown in Table 5.

The baseline multi-lingual NMT system is brittle and breaks when it has to translate a code mixed text. On the other hand, a multi-lingual system trained with PhraseOut augmentation is more robust to code mixed input and outputs a reasonable translation.

Source	I am sure minister ने अपने hired writer को बोला होगा "कुछ अच्छा लिखो"
Reference	I am sure the minister would have told his hired writer to "Write Something Good"
Multilingual NMT	I am sure minister hired writer " "
PhraseOut	I am sure minister has told his hired speech to write the good note

Table 5: Translating social media text

6 Related Works

Recently a few works have explored the utility of code mixed (also called code switched) augmentations. (Song et al., 2019) proposes word replacements as in PhraseOut for the purpose of lexicon induction. They perform data augmentation on the parallel data and don't use any monolingual data. Recently code switched pretraining (Yang et al., 2020) (Lin et al.) has shown to work favorably against popular cross lingual pretraining methods.

But all these methods use unsupervised dictionary induction (Artetxe et al., 2018) to obtain parallel word translations. But unsupervised dictionary induction performs quite poorly for distant language pairs such as English to Indian languages (Khatri et al., 2020).

7 Conclusion and Future Work

We propose a simple yet useful data augmentation technique suitable for a multilingual NMT setting, called PhraseOut. Our experiments confirm that PhraseOut is effective in improving the performance of multilingual NMT systems. The improvement in performance could be attributed to better regularization from code mixing.

Training using code mixed data could be useful for improving the robustness of NMT systems. Social media text usually contains code mixed data. One future direction of research is to see how PhraseOut can improve robustness in this kind of a setting.

References

Roee Aharoni, Melvin Johnson, and Orhan Firat. 2019. Massively multilingual neural machine translation. In *NAACL*.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2018. A robust self-learning method for fully unsupervised cross-lingual mappings of word embeddings. In *ACL*.

David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin Raffel. 2019. Mixmatch: A holistic approach to semi-supervised learning. *arXiv preprint arXiv:1905.02249*.

Franck Burlot and Francois Yvon. 2018. Using monolingual data in neural machine translation: a systematic study. In *WMT*.

Anna Currey, Antonio Valerio Miceli Barone, and Kenneth Heafield. 2017. Copied monolingual data improves low-resource neural machine translation. In *WMT*.

Sergey Edunov, Myle Ott, Michael Auli, and David Grangier. 2018. Understanding back-translation at scale. In *EMNLP*.

Yarin Gal and Zoubin Ghahramani. 2016. A theoretically grounded application of dropout in recurrent neural networks. In *NIPS*.

Junxian He, Jiatao Gu, Jiajun Shen, and Marc'Aurelio Ranzato. 2020. Revisiting self-training for neural sequence generation. In *ICLR*.

Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2017. Google's multilingual neural machine translation system: Enabling zero-shot translation. In *ACL*.

Jyotsana Khatri, Rudra Murthy, and Pushpak Bhattacharyya. 2020. A study of efficacy of cross-lingual word embeddings for indian languages. In *CoDS COMAD*.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. 2007. Moses: Open source toolkit for statistical machine translation. In *ACL*.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *NAACL*.

Taku Kudo. 2018. Subword regularization: Improving neural network translation models with multiple subword candidatesn. In *ACL*.

Anoop Kunchukuttan, Pratik Mehta, and Pushpak Bhattacharyya. 2018. The IIT Bombay English-Hindi Parallel Corpus. In *LREC*.

Zehui Lin, Xiao Pan, Mingxuan Wang, Xipeng Qiu, Jiangtao Feng, Hao Zhou, and Lei Li. Pre-training multilingual neural machine translation by leveraging alignment information.

Toshiaki Nakazawa, Katsuhito Sudoh, Shohei Higashiyama, Chenchen Ding, Raj Dabre, Hideya Mino, Isao Goto, Win Pa Pa, Anoop Kunchukuttan, and Sadao Kurohashi. 2018. Overview of the 5th workshop on asian translation. In *WAT*.

Graham Neubig and Junjie Hu. 2018. Rapid adaptation of neural machine translation to new languages. In *EMNLP*.

Daniel S. Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D. Cubuk, and Quoc V. Le. 2019. Specaugment: A simple data augmentation method for automatic speech recognition. In *arXiv:1904.08779v1*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016a. Edinburgh neural machine translation systems for wmt 16. In *WMT*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016b. Improving neural machine translation models with monolingual data. In *ACL*.

Kai Song, Yue Zhang, Heng Yu, Weihua Luo, Kun Wang, and Min Zhang. 2019. Code-switching for enhancing nmt with pre-specified translation. In *NAACL*.

Vaibhav, Sumeet Singh, Craig Stewart, and Graham Neubig. 2019. Improving robustness of machine translation with synthetic noise. In *NAACL*.

P Vincent, Hugo Larochelle, Yoshua Bengio, , and P.A. Manzago. 2008. Extracting and composing robust features with denoising autoencoders. In *ICML*.

Xinyi Wang, Hieu Pham, Zihang Dai1, and Graham Neubig. 2018. Switchout: an efficient data augmentation algorithm for neural machine translation. In *EMNLP*.

Zhen Yang, Bojie Hu, Ambyera Han, Shen Huang, and Qi Ju. 2020. Csp: Code-switching pre-training for neural machine translation. In *EMNLP*.