Soft Contextual Data Augmentation for Neural Machine Translation

Fei Gao[1,*], **Jinhua Zhu**[2,*], **Lijun Wu**[3], **Yingce Xia**[4], **Tao Qin**[4],
Xueqi Cheng[1], **Wengang Zhou**[2], **Tie-Yan Liu**[4]
[1]Institute of Computing Technology, Chinese Academy of Sciences;
[2]University of Science and Technology of China,
[3]Sun Yat-sen University, [4]Microsoft Reserach Asia;
[1]{gaofei17b, cxq}@ict.ac.cn,
[2]{teslazhu@mail., zhwg@}ustc.edu.cn,
[3]wulijun3@mail2.sysu.edu.cn,
[4]{Yingce.Xia, taoqin, tyliu}@microsoft.com

Abstract

While data augmentation is an important trick to boost the accuracy of deep learning methods in computer vision tasks, its study in natural language tasks is still very limited. In this paper, we present a novel data augmentation method for neural machine translation. Different from previous augmentation methods that randomly drop, swap or replace words with other words in a sentence, we softly augment a randomly chosen word in a sentence by its contextual mixture of multiple related words. More accurately, we replace the one-hot representation of a word by a distribution (provided by a language model) over the vocabulary, i.e., replacing the embedding of this word by a weighted combination of multiple semantically similar words. Since the weights of those words depend on the contextual information of the word to be replaced, the newly generated sentences capture much richer information than previous augmentation methods. Experimental results on both small scale and large scale machine translation datasets demonstrate the superiority of our method over strong baselines[1].

1 Introduction

Data augmentation is an important trick to boost the accuracy of deep learning methods by generating additional training samples. These methods have been widely used in many areas. For example, in computer vision, the training data are augmented by transformations like random rotation, resizing, mirroring and cropping (Krizhevsky et al., 2012; Cubuk et al., 2018).

While similar random transformations have also been explored in natural language processing (NLP) tasks (Xie et al., 2017), data augmentation is still not a common practice in neural machine translation (NMT). For a sentence, existing methods include randomly swapping two words, dropping word, replacing word with another one and so on. However, due to text characteristics, these random transformations often result in significant changes in semantics.

A recent new method is contextual augmentation (Kobayashi, 2018; Wu et al., 2018), which replaces words with other words that are predicted using language model at the corresponding word position. While such method can keep semantics based on contextual information, this kind of augmentation still has one limitation: to generate new samples with adequate variation, it needs to sample multiple times. For example, given a sentence in which N words are going to be replaced with other words predicted by one language model, there could be as many as exponential candidates. Given that the vocabulary size is usually large in languages, it is almost impossible to leverage all the possible candidates for achieving good performance.

In this work, we propose soft contextual data augmentation, a simple yet effective data augmentation approach for NMT. Different from the previous methods that randomly replace one word to another, we propose to augment NMT training data by replacing a randomly chosen word in a sentence with a *soft word*, which is a probabilistic distribution over the vocabulary. Such a distributional representation can capture a mixture of multiple candidate words with adequate variations in augmented data. To ensure the distribution reserving similar semantics with original word, we calculate it based on the contextual information by using a language model, which is pretrained on the training corpus.

To verify the effectiveness of our method, we conduct experiments on four machine transla-

[*]The first two authors contributed equally to this work.
[1]Our code can be found at https://github.com/teslacool/SCA

Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 5539–5544
Florence, Italy, July 28 - August 2, 2019.

tion tasks, including IWSLT2014 German to English, Spanish to English, Hebrew to English and WMT2014 English to German translation tasks. In all tasks, the experimental results show that our method can obtain remarkable BLEU score improvement over the strong baselines.

2 Related Work

We introduce several related works about data augmentation for NMT.

Artetxe et al. (2017) and Lample et al. (2017) randomly shuffle (swap) the words in a sentence, with constraint that the words will not be shuffled further than a fixed small window size. Iyyer et al. (2015) and Lample et al. (2017) randomly drop some words in the source sentence for learning an autoencoder to help train the unsupervised NMT model. In Xie et al. (2017), they replace the word with a placeholder token or a word sampled from the frequency distribution of vocabulary, showing that data noising is an effective regularizer for NMT. Fadaee et al. (2017) propose to replace a common word by low-frequency word in the target sentence, and change its corresponding word in the source sentence to improve translation quality of rare words. Most recently, Kobayashi (2018) propose an approach to use the prior knowledge from a bi-directional language model to replace a word token in the sentence. Our work differs from their work that we use a soft distribution to replace the word representation instead of a word token.

3 Method

In this section, we present our method in details.

3.1 Background and Motivations

Given a source and target sentence pair (s, t) where $s = (s_1, s_2, ..., s_T)$ and $t = (t_1, t_2, ..., t_{T'})$, a neural machine translation system models the conditional probability $p(t_1, ..., t_{T'}|s_1, ..., s_T)$. NMT systems are usually based on an encoder-decoder framework with an attention mechanism (Sutskever et al., 2014; Bahdanau et al., 2014). In general, the encoder first transforms the input sentence with words/tokens $s_1, s_2, ..., s_T$ into a sequence of hidden states $\{h_t\}_{t=1}^{T}$, and then the decoder takes the hidden states from the encoder as input to predict the conditional distribution of each target word/token $p(t_\tau|h_t, t_{<\tau})$ given the previous ground truth target word/tokens. Similar to the NMT decoder, a language model is intended to predict the next word distribution given preceding words, but without another sentence as a conditional input. In NMT, as well as other NLP tasks, each word is assigned with a unique ID, and thus represented as an one-hot vector. For example, the i-th word in the vocabulary (with size $|V|$) is represented as a $|V|$-dimensional vector $(0, 0, ..., 1, ..., 0)$, whose i-th dimension is 1 and all the other dimensions are 0.

Existing augmentation methods generate new training samples by replacing one word in the original sentences with another word (Wang et al., 2018; Kobayashi, 2018; Xie et al., 2017; Fadaee et al., 2017). However, due to the sparse nature of words, it is almost impossible for those methods to leverage all possible augmented data. First, given that the vocabulary is usually large, one word usually has multiple semantically related words as replacement candidates. Second, for a sentence, one needs to replace multiple words instead of a single word, making the number of possible sentences after augmentation increases exponentially. Therefore, these methods often need to augment one sentence multiple times and each time replace a different subset of words in the original sentence with different candidate words in the vocabulary; even doing so they still cannot guarantee adequate variations of augmented sentences. This motivates us to augment training data in a *soft* way.

3.2 Soft Contextual Data Augmentation

Inspired by the above intuition, we propose to augment NMT training data by replacing a randomly chosen word in a sentence with a *soft word*. Different from the discrete nature of words and their one-hot representations in NLP tasks, we define a soft word as a distribution over the vocabulary of $|V|$ words. That is, for any word $w \in V$, its soft version is $P(w) = (p_1(w), p_2(w), ..., p_{|V|}(w))$, where $p_j(w) \geq 0$ and $\sum_{j=1}^{|V|} p_j(w) = 1$.

Since $P(w)$ is a distribution over the vocabulary, one can sample a word with respect to this distribution to replace the original word w, as done in Kobayashi (2018). Different from this method, we directly use this distribution vector to replace a randomly chosen word from the original sentence. Suppose E is the embedding matrix of all the $|V|$ words. The embedding of the soft word w is

$$e_w = P(w)E = \sum_{j=0}^{|V|} p_j(w)E_j, \qquad (1)$$

which is the expectation of word embeddings over the distribution defined by the soft word.

The distribution vector $P(w)$ of a word w can be calculated in multiple ways. In this work, we leverage a pretrained language model to compute $P(w)$ and condition on all the words preceding w. That is, for the t-th word x_t in a sentence, we have

$$p_j(x_t) = LM(w_j|x_{<t}),$$

where $LM(w_j|x_{<t})$ denotes the probability of the j-th word in the vocabulary appearing after the sequence $x_1, x_2, \cdots, x_{t-1}$. Note that the language model is pretrained using the same training corpus of the NMT model. Thus the distribution $P(w)$ calculated by the language model can be regarded as a smooth approximation of the original one-hot representation, which is very different from previous augmentation methods such as random swapping or replacement. Although this distributional vector is noisy, the noise is aligned with the training corpus.

Figure 1 shows the architecture of the combination of the encoder of the NMT model and the language model. The decoder of the NMT model is similarly combined with the language model. In experiments, we randomly choose a word in the training data with probability γ and replace it by its soft version (probability distribution).

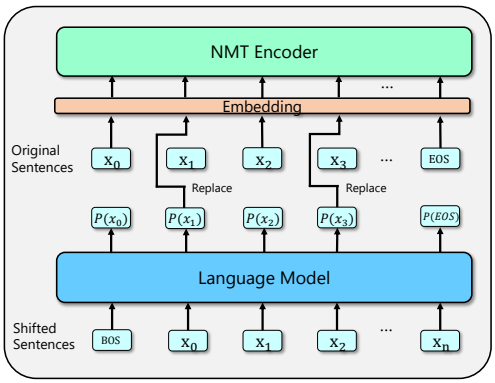

Figure 1: The overall architecture of our soft contextual data augmentation approach in encoder side for source sentences. The decoder side for target sentences is similar.

At last, it is worth pointing out that no additional monolingual data is used in our method. This is different from previous techniques, such as back translation, that rely on monolingual data (Sennrich et al., 2015a; Gulcehre et al., 2015; Cheng et al., 2016; He et al., 2016; Hoang et al., 2018). We leave the exploration of leveraging monolingual data to future work.

4 Experiment

In this section, we demonstrate the effectiveness of our method on four translation datasets with different scale. The translation quality is evaluated by case-sensitive BLEU score. We compare our approach with following baselines:

- *Base*: The original training strategy without any data augmentation;
- *Swap*: Randomly swap words in nearby positions within a window size k (Artetxe et al., 2017; Lample et al., 2017);
- *Dropout:* Randomly drop word tokens (Iyyer et al., 2015; Lample et al., 2017);
- *Blank*: Randomly replace word tokens with a placeholder token (Xie et al., 2017);
- *Smooth*: Randomly replace word tokens with a sample from the unigram frequency distribution over the vocabulary (Xie et al., 2017);
- LM_{sample}: Randomly replace word tokens sampled from the output distribution of one language model (Kobayashi, 2018).

All above introduced methods except *Swap* incorporate a hyper-parameter, the probability γ of each word token to be replaced in training phase. We set γ with different values in $\{0, 0.05, 0.1, 0.15, 0.2\}$, and report the best result for each method. As for *swap*, we use 3 as window size following Lample et al. (2017).

For our proposed method, we train two language models for each translation task. One for source language, and the other one for target language. The training data for the language models is the corresponding source/target data from the bilingual translation dataset.

4.1 Datasets

We conduct experiments on IWSLT2014 {German, Spanish, Hebrew} to English ({De, Es, He}→En) and WMT2014 English to German (En→De) translation tasks to verify our approach. We follow the same setup in Gehring et al. (2017) for IWSLT2014 De→En task. The training data and validation data consist of $160k$ and $7k$

	IWSLT			WMT
	De → En	**Es → En**	**He → En**	**En → De**
Base	34.79	41.58	33.64	28.40
+*Swap*	34.70	41.60	34.25	28.13
+*Dropout*	35.13	41.62	34.29	28.29
+*Blank*	35.37	42.28	34.37	28.89
+*Smooth*	35.45	41.69	34.61	28.97
$+LM_{sample}$	35.40	42.09	34.31	28.73
Ours	**35.78**	**42.61**	**34.91**	**29.70**

Table 1: BLEU scores on four translation tasks.

sentence pairs. *tst2010*, *tst2011*, *tst2012*, *dev2010* and *dev2012* are concatenated as our test data. For Es→En and He→En tasks, there are $181k$ and $151k$ parallel sentence pairs in each training set, and we use *tst2013* as the validation set, *tst2014* as the test set. For all IWSLT translation tasks, we use a joint source and target vocabulary with $10K$ byte-pair-encoding (BPE) (Sennrich et al., 2015b) types. For WMT2014 En→De translation, again, we follow Gehring et al. (2017) to filter out $4.5M$ sentence pairs for training. We concatenate *newstest2012* and *newstest2013* as the validation set and use *newstest2014* as test set. The vocabulary is built upon the BPE with $40k$ sub-word types.

4.2 Model Architecture and Optimization

We adopt the sate-of-the-art Transformer architecture (Vaswani et al., 2017) for language models and NMT models in our experiments. For IWSLT tasks, we take the $transformer\_base$ configuration, except a) the dimension of the inner MLP layer is set as 1024 instead of 2048 and b) the number of attention heads is 4 rather than 8. As for the WMT En→De task, we use the default $transformer\_big$ configuration for the NMT model, but the language model is configured with $transformer\_base$ setting in order to speed up the training procedure. All models are trained by Adam (Kingma and Ba, 2014) optimizer with default learning rate schedule as Vaswani et al. (2017). Note that after training the language models, the parameters of the language models are fixed while we train the NMT models.

4.3 Main Results

The evaluation results on four translation tasks are presented in Table 1. As we can see, our method can consistently achieve more than 1.0 BLEU score improvement over the strong Transformer base system for all tasks. Compared with other augmentation methods, we can find that 1) our method achieves the best results on all the translation tasks and 2) unlike other methods that may not be powerful in all tasks, our method universally works well regardless of the dataset. Specially, on the large scale WMT 2014 En→De dataset, although this dataset already contains a large amount of parallel training sentence pairs, our method can still outperform the strong base system by $+1.3$ BLEU point and achieve 29.70 BLEU score. These results clearly demonstrate the effectiveness of our approach.

4.4 Study

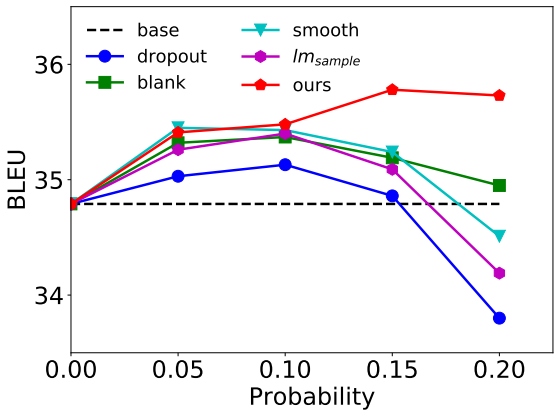

Figure 2: BLEU scores of each method on IWSLT De→En dataset with different replacing probability.

As mentioned in Section 4, we set different

probability value of γ to see the effect of our approach and other methods in this subsection. Figure 2 shows the BLEU scores on IWSLT De→En dataset of each method, from which we can see that our method can observe a consistent BLEU improvement within a large probability range and obtain a strongest performance when $\gamma = 0.15$. However, other methods are easy to lead to performance drop over the baseline if $\gamma > 0.15$, and the improvement is also limited for other settings of γ. This can again prove the superior performance of our method.

5 Conclusions and Future Work

In this work, we have presented soft contextual data augmentation for NMT, which replaces a randomly chosen word with a soft distributional representation. The representation is a probabilistic distribution over vocabulary and can be calculated based on the contextual information of the sentence. Results on four machine translation tasks have verified the effectiveness of our method.

In the future, besides focusing on the parallel bilingual corpus for the NMT training in this work, we are interested in exploring the application of our method on the monolingual data. In addition, we also plan to study our approach in other natural language tasks, such as text summarization.

References

Mikel Artetxe, Gorka Labaka, Eneko Agirre, and Kyunghyun Cho. 2017. Unsupervised neural machine translation. *arXiv preprint arXiv:1710.11041*.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Yong Cheng, Wei Xu, Zhongjun He, Wei He, Hua Wu, Maosong Sun, and Yang Liu. 2016. Semi-supervised learning for neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1965–1974.

Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. 2018. Autoaugment: Learning augmentation policies from data. *arXiv preprint arXiv:1805.09501*.

Marzieh Fadaee, Arianna Bisazza, and Christof Monz. 2017. Data augmentation for low-resource neural machine translation. *arXiv preprint arXiv:1705.00440*.

Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin. 2017. Convolutional Sequence to Sequence Learning. In *Proc. of ICML*.

Caglar Gulcehre, Orhan Firat, Kelvin Xu, Kyunghyun Cho, Loic Barrault, Huei-Chi Lin, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2015. On using monolingual corpora in neural machine translation. *arXiv preprint arXiv:1503.03535*.

Di He, Yingce Xia, Tao Qin, Liwei Wang, Nenghai Yu, Tie-Yan Liu, and Wei-Ying Ma. 2016. Dual learning for machine translation. In *Advances in Neural Information Processing Systems*, pages 820–828.

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

Mohit Iyyer, Varun Manjunatha, Jordan Boyd-Graber, and Hal Daumé III. 2015. Deep unordered composition rivals syntactic methods for text classification. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, volume 1, pages 1681–1691.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Sosuke Kobayashi. 2018. Contextual augmentation: Data augmentation by words with paradigmatic relations. *arXiv preprint arXiv:1805.06201*.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105.

Guillaume Lample, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2017. Unsupervised machine translation using monolingual corpora only. *arXiv preprint arXiv:1711.00043*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015a. Improving neural machine translation models with monolingual data. *arXiv preprint arXiv:1511.06709*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015b. Neural machine translation of rare words with subword units. *arXiv preprint arXiv:1508.07909*.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all

you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.

Xinyi Wang, Hieu Pham, Zihang Dai, and Graham Neubig. 2018. Switchout: an efficient data augmentation algorithm for neural machine translation. *arXiv preprint arXiv:1808.07512*.

Xing Wu, Shangwen Lv, Liangjun Zang, Jizhong Han, and Songlin Hu. 2018. Conditional bert contextual augmentation. *arXiv preprint arXiv:1812.06705*.

Ziang Xie, Sida I Wang, Jiwei Li, Daniel Lévy, Aiming Nie, Dan Jurafsky, and Andrew Y Ng. 2017. Data noising as smoothing in neural network language models. *arXiv preprint arXiv:1703.02573*.