Sentence Embedding for Neural Machine Translation Domain Adaptation

Rui Wang, Andrew Finch, Masao Utiyama and Eiichiro Sumita
National Institute of Information and Communications Technology (NICT)
3-5 Hikari-dai, Seika-cho, Soraku-gun, Kyoto, Japan
{wangrui,andrew.finch,mutiyama,eiichiro.sumita}@nict.go.jp

Abstract

Although new corpora are becoming increasingly available for machine translation, only those that belong to the same or similar domains are typically able to improve translation performance. Recently Neural Machine Translation (NMT) has become prominent in the field. However, most of the existing domain adaptation methods only focus on phrase-based machine translation. In this paper, we exploit the NMT's internal embedding of the source sentence and use the sentence embedding similarity to select the sentences which are close to in-domain data. The empirical adaptation results on the IWSLT English-French and NIST Chinese-English tasks show that the proposed methods can substantially improve NMT performance by 2.4-9.0 BLEU points, outperforming the existing state-of-the-art baseline by 2.3-4.5 BLEU points.

1 Introduction

Recently, Neural Machine Translation (NMT) has set new state-of-the-art benchmarks on many translation tasks (Cho et al., 2014; Bahdanau et al., 2015; Jean et al., 2015; Tu et al., 2016; Mi et al., 2016; Zhang et al., 2016). An ever increasing amount of data is becoming available for NMT training. However, only the in-domain or related-domain corpora tend to have a positive impact on NMT performance. Unrelated additional corpora, known as out-of-domain corpora, have been shown not to benefit some domains and tasks for NMT, such as TED-talks and IWSLT tasks (Luong and Manning, 2015).

To the best of our knowledge, there are only a few works concerning NMT adaptation (Luong and Manning, 2015; Freitag and Al-Onaizan, 2016). Most traditional adaptation methods focus on Phrase-Based Statistical Machine Translation (PBSMT) and they can be broken down broadly into two main categories namely *model adaptation* and *data selection* (Joty et al., 2015) as follows.

For model adaptation, several PBSMT models, such as language models, translation models and reordering models, individually corresponding to each corpus, are trained. These models are then combined to achieve the best performance (Sennrich, 2012; Sennrich et al., 2013; Durrani et al., 2015). Since these methods focus on the internal models within a PBSMT system, they are not applicable to NMT adaptation. Recently, an NMT adaptation method (Luong and Manning, 2015) was proposed. The training is performed in two steps: first the NMT system is trained using out-of-domain data, and then further trained using in-domain data. Empirical results show their method can improve NMT performance, and this approach provides a natural baseline.

For adaptation through data selection, the main idea is to score the out-domain data using models trained from the in-domain and out-of-domain data and select training data from the out-of-domain data using a cut-off threshold on the resulting scores. A language model can be used to score sentences (Moore and Lewis, 2010; Axelrod et al., 2011; Duh et al., 2013; Wang et al., 2015), as well as joint models (Hoang and Sima'an, 2014a,b; Durrani et al., 2015), and more recently Convolutional Neural Network (CNN) models (Chen et al., 2016). These methods select useful sentences from the whole corpus, so they can be directly applied to NMT. However, these methods are specifically designed for PBSMT and nearly all of them use the models or criteria which do not have a direct relationship with the neural

Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Short Papers), pages 560–566
Vancouver, Canada, July 30 - August 4, 2017.
https://doi.org/10.18653/v1/P17-2089

translation process.

For NMT sentences selection, our hypothesis is that the NMT system itself can be used to score each sentence in the training data. Specifically, an NMT system embeds the source sentence into a vector representation[1] and we can use these vectors to measure a sentence pair's similarity to the in-domain corpus. In comparison with the CNN or other sentence embedding methods, this method can directly make use of information induced by the NMT system information itself. In addition, the proposed sentence selection method can be used in conjunction with the NMT further training method (Luong and Manning, 2015).

2 NMT Background

An attention-based NMT system uses a Bidirectional RNN (BiRNN) as an encoder and a decoder that emulates searching through a source sentence during decoding (Bahdanau et al., 2015). The encoder's BiRNN consists of forward and backward RNNs. Each word x_i is represented by concatenating the forward hidden state $\overrightarrow{h_i}$ and the backward one $\overleftarrow{h_i}$ as $h_i = [\overrightarrow{h_i}; \overleftarrow{h_i}]^\top$. In this way, the source sentence $\mathbf{X} = \{x_1, ..., x_{T_x}\}$ can be represented as annotations $\mathbf{H} = \{h_1, ..., h_{T_x}\}$. In the decoder, an RNN hidden state s_j for time j is computed by:

$$s_j = f(s_{j-1}, y_{j-1}, c_j). \tag{1}$$

The context vector c_j is then, computed as a weighted sum of these annotations $\mathbf{H} = \{h_1, ..., h_{T_x}\}$, by using alignment weight α_{ji}:

$$c_j = \sum_{j=1}^{T_x} \alpha_{ji} h_i. \tag{2}$$

3 Sentence Embedding and Selection

3.1 Sentence Embedding

A source sentence can be represented as the annotations $\mathbf{H}$. However the length of $\mathbf{H}$ depends on the sentence length T_x. To represent a sentence as a fixed-length vector, we adopt the initial hidden layer state s_{init} for the decoder as this vector:

$$s_{init}(\mathbf{X}) = tanh(\mathbf{W}\frac{\sum_{i=1}^{T_x} h_i}{T_x} + \mathbf{b}), h_i \in \mathbf{H}, \tag{3}$$

where an average pooling layer averages the annotation h_i for each source word into a fixed-length source sentence vector, and a nonlinear transition layer (weights $\mathbf{W}$ and bias $\mathbf{b}$ are jointly trained with all the other components of NMT system) transforms this embedded source sentence vector into the initial hidden state s_{init} for the decoder (Bahdanau et al., 2015).

3.2 Sentence Selection

We employ the data selection method, which is inspired by (Moore and Lewis, 2010; Axelrod et al., 2011; Duh et al., 2013). As Axelrod et al. (2011) mentioned, there are some pseudo in-domain data in out-of-domain data, which are close to in-domain data. Our intuition is to select the sentences whose embeddings are similar to the average in-domain ones, while being dis-similar to the average out-of-domain ones:

- 1) We train a French-to-English NMT system $\mathbf{N_{FE}}$ using the in-domain and out-of-domain data together as training data.[2]
- 2) Each sentence f in the training data F (both in-domain F_{in} and out-of-domain F_{out}) is embedded as a vector $v_f = s_{init}(f)$ by using $\mathbf{N_{FE}}$.
- 3) The sentence pairs (f, e) in the out-of-domain corpus F_{out} are classified into two sets: the sentences close to in-domain sentences, and those that are distant.

That is, we firstly calculate the vector centers of in-domain $C_{F_{in}}$ and out-of-domain $C_{F_{out}}$ corpora, respectively.

$$\begin{aligned} C_{F_{in}} &= \frac{\sum_{f \in F_{in}} v_f}{|F_{in}|}, \\ C_{F_{out}} &= \frac{\sum_{f \in F_{out}} v_f}{|F_{out}|}. \end{aligned} \tag{4}$$

Then we measure the Euclidean distance d between each sentence vector v_f and in-domain

[1] Li et al. (2016)'s fine-tuned NMT systems apply a similar sentence representation. In comparison, we adopt a transition layer between the source and target layers and don't use test data.

[2] It is possible to use a sample of the out-of-domain data. In this paper, we use all of them.

vector center $C_{F_{in}}$ as $d(v_f, C_{F_{in}})$ and out-of-domain vector center $C_{F_{out}}$ as $d(v_f, C_{F_{out}})$, respectively. We use the difference δ of these two distances to classify each sentence:

$$\delta_f = d(v_f, C_{F_{in}}) - d(v_f, C_{F_{out}}). \quad (5)$$

By using an English-to-French NMT system $\mathbf{N_{EF}}$, we can obtain a target sentence embedding v_e, in-domain target vector center $C_{E_{in}}$ and out-of-domain target vector center $C_{E_{out}}$. Corresponding distance difference δ_e is,

$$\delta_e = d(v_e, C_{E_{in}}) - d(v_e, C_{E_{out}}). \quad (6)$$

δ_f, δ_e and $\delta_{fe} = \delta_f + \delta_e$ can be used to select sentences. That is, the sentence pairs (f, e) with δ_f (or δ_e, δ_{fe}) less than a threshold are the new selected in-domain corpus. This threshold is tuned by using the development data.

4 Experiments

4.1 Data sets

The proposed methods were evaluated on two data sets as shown in Table 1.

- IWSLT 2014 English (EN) to French (FR) corpus[3] was used as in-domain training data and dev2010 and test2010/2011 (Cettolo et al., 2014), were selected as development (dev) and test data, respectively. Out-of-domain corpora contained Common Crawl, Europarl v7, News Commentary v10 and United Nation (UN) EN-FR parallel corpora.[4]

- NIST 2006 Chinese (ZH) to English corpus[5] was used as the in-domain training corpus, following the settings of (Wang et al., 2014). Chinese-to-English UN data set (LDC2013T06) and NTCIR-9 (Goto et al., 2011) patent data set were used as out-of-domain data. NIST MT 2002-2004 and NIST MT 2005/2006 were used as the development and test data, respectively. We are aware of that there are additional NIST corpora in a similar domain, but because this task was for domain adaptation, we only selected a small subset, which is mainly focused on news and blog texts. The statistics on data sets were shown in Table 1.

These adaptation corpora settings were nearly the same as that used in (Wang et al., 2016). The differences were:

- For IWSLT, they chose FR-EN translation task, which is popular in PBSMT. We chose EN-FR, which is more popular in NMT;

- For NIST, they chose 02-05 as dev set, and we chose 02-04. Because we would report results on two test sets (MT05 and MT06) in comparison with only one (MT06).

IWSLT EN-FR	Sentences	Tokens
TED training (in-domain)	178.1K	3.5M
WMT training (out-of-domain)	17.8M	450.0M
TED dev2010	0.9K	20.1K
TED test2010	1.6K	31.9K
TED test2011	0.8K	15.6K
NIST ZH-EN	**Sentences**	**Tokens**
NIST in-domain training	430.8K	12.6M
out-of-domain training	8.8M	249.4M
dev (MT02-04)	3.4K	106.4K
test (MT05)	1.0K	34.7K
test (MT06)	1.6K	46.7K

Table 1: Statistics on data sets.

4.2 NMT System

We implemented the proposed method in Groundhog[6] (Bahdanau et al., 2015), which is one of the state-of-the-art NMT frameworks. The default settings of Groundhog were applied for all NMT systems: the word embedding dimension was 620 and the size of a hidden layer was 1000, the batch size was 64, the source and target side vocabulary sizes were 30K, the maximum sequence length were 50, and the beam size for decoding was 10. Default dropout were applied. We used a mini-batch Stochastic Gradient Descent (SGD) algorithm together with ADADELTA optimizer (Zeiler, 2012). Training was conducted on a single Tesla K80 GPU. Each NMT model was trained for 500K batches, taking 7-10 days. For sentence embedding and selection, it only took several hours to process all of sentences in the training data, because decoding was not necessary.

[3] https://wit3.fbk.eu/mt.php?release=2014-01
[4] http://statmt.org/wmt15/translation-task.html
[5] http://www.itl.nist.gov/iad/mig/tests/mt/2006/

[6] https://github.com/lisa-groundhog/GroundHog

4.3 Baselines

Along with the standard NMT baseline system, we also compared the proposed methods to the recent state-of-the-art NMT adaptation method of Luong and Manning (2015)[7] as described in Section 1. Two typical sentence selection methods for PBSMT were also used as baselines: Axelrod et al. (2011) used language model-based cross-entropy difference as criterion; Chen et al. (2016) used a CNN to classify the sentences as either in-domain or out-of-domain. In addition, we randomly sampled out-of-domain data to create a corpus the same size as that used for the best performing proposed system. We tried our best to re-implement the baseline methods using the same basic NMT setting as the proposed method.

4.4 Results and Analyses

In Tables 2 and 3, the in, out and $in + out$ indicate that the in-domain, out-of-domain and their mixture were used as the NMT training corpora. δ_f, δ_e and δ_{fe} indicate that corresponding criterion was used to select sentences, and these selected sentences were added to in-domain corpus to construct the new training corpora. $+fur$ indicates that the selected sentences were used to train an initial NMT system, and then this initial system was further trained by in-domain data (Luong and Manning, 2015). The threshold for the sentence selection method was selected on development data. That is, we selected the top ranked 10%, 20%,...,90% out-of-domain data to be added into the in-domain data, and the best performing models on development data were used in the evaluation on test data.

The vocabulary was built by using the selected corpus and in-domain corpus.[8] Translation performance was measured by case-insensitive BLEU (Papineni et al., 2002). Since the proposed method is a sentence selection approach, we can also show the effect on standard PBSMT (Koehn et al., 2007).

In the IWSLT task, the observations were as follows:

- Adding out-of-domain to in-domain data, or directly using out-of-domain data, degraded PBSMT and NMT performance.
- Adding data selected by δ_f, δ_e and δ_{fe} substantially improved NMT performance (3.9 to 6.6 BLEU points), and gave rise to a modest improvement in PBSMT performance (0.4 to 3.1 BLEU points). This method also outperformed the best existing baselines by up to 1.1 BLEU points for NMT and 0.8 BLEU for PBSMT.
- The proposed method worked synergistically with Luong's further training method, and the combination was able to add up to an additional 2-3 BLEU points, indicating that the proposed method and Luong's method are essentially orthogonal.
- The performance by using both sides of sentence embeddings δ_{fe} was slightly better

Methods	Sent. No.	SMT tst10	SMT tst11	NMT tst10	NMT tst11
in	178.1K	31.06	32.50	29.23	30.00
out	17.7M	30.04	29.29	27.30	28.48
in+out	17.9 M	30.00	30.26	28.89	28.55
Random	5.5M	31.22	33.85	30.53	32.37
Luong	17.9 M	N/A	N/A	32.21	35.03
Axelrod	9.0M	32.06	34.81	32.26	35.54
Chen	7.3M	31.42	33.78	30.32	33.81
δ_f	7.3M	31.46	33.13	32.13	34.81
δ_e	3.7M	**32.08**	35.94	32.84	36.56
δ_{fe}	5.5M	31.79	**35.66**	32.67	36.64
δ_f+fur	7.3M	N/A	N/A	34.04	37.18
δ_e+fur	3.7M	N/A	N/A	33.88	38.04
δ_{fe}+fur	5.5M	N/A	N/A	**34.52**	**39.02**

Table 2: IWSLT EN-FR results. Luong and Manning (2015)'s **fur**ther (shorted as fur in Tables 2 and 3) training method can only be applied to NMT.

Methods	Sent. No.	SMT MT05	SMT MT06	NMT MT05	NMT MT06
in	430.8K	29.66	30.73	27.28	26.82
out	8.8M	29.91	30.13	28.67	27.79
in+out	9.3M	30.23	30.11	28.91	28.22
Random	5.7M	29.90	30.18	28.02	27.49
Luong	9.3M	N/A	N/A	29.91	29.61
Axelrod	2.2M	30.52	30.96	28.41	28.75
Chen	4.8M	30.64	31.05	28.39	28.06
δ_f	4.8M	30.90	**31.96**	29.21	30.14
δ_e	2.2M	**30.94**	31.33	30.00	30.63
δ_{fe}	5.7M	30.72	31.43	30.13	31.07
δ_f+fur	4.8M	N/A	N/A	30.80	31.54
δ_e+fur	2.2M	N/A	N/A	30.49	31.13
δ_{fe}+fur	5.7M	N/A	N/A	**31.35**	**31.80**

Table 3: NIST ZH-EN results.

[7] Freitag and Al-Onaizan (2016)'s method is quite similar to Luong and Manning (2015)'s, so we did not compare to them.

[8] According to our empirical comparison, the performance did not significantly change if we used $in + out$ to build the vocabulary for all of the systems.

than using monolingual sentence embedding δ_f and δ_e.

In the NIST task, the observations were similar to the IWSLT task, except:

- Adding out-of-domain slightly improved PBSMT and NMT performance.
- The proposed method improved both PBSMT and NMT performance, but not as substantially as in IWSLT.

These observations suggest that the out-of-domain data was closer to the in-domain than in IWSLT.

5 Discussions

5.1 Selected Size Effect

We show experimental results on varying the size of additional data selected from the out-of-domain dataset, in Figure 1. It shows that the proposed method δ_{fe} reached the highest performance on dev set, when top 30% out-of-domain sentences are selected as pseudo in-domain data. δ_{fe} outperforms the other methods in most of the cases on development data.

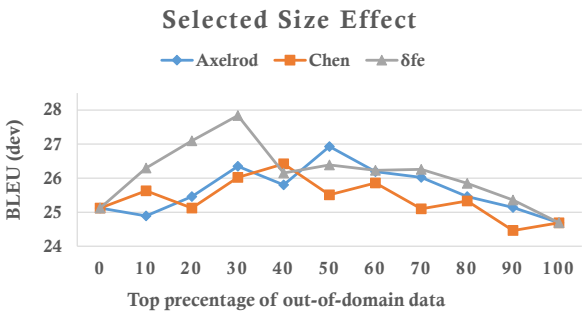

Figure 1: Selected size tuning on IWSLT.

5.2 Training Time Effect

We also show the relationship between BLEU and batches of training in Figure 2.

Most of the methods (without further training) converged after similar batches training. Specifically, in researched the highest BLEU performance on dev faster than other methods (without further training), then decreased and finally converged.

The further training methods, which firstly trained the models using out-of-domain data and then in-domain data, converged very soon after

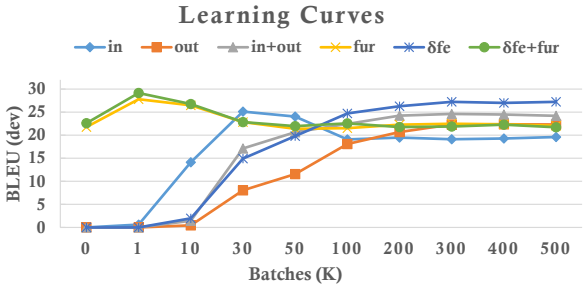

Figure 2: Training time on IWSLT.

in-domain data were introduced. In further training, the out-of-domain trained system could be considered as a pre-trained NMT system. Then the in-domain data training help NMT system overfit at in-domain data and gained around two BLEU improvement.

6 Conclusion and Future Work

In this paper, we proposed a straightforward sentence selection method for NMT domain adaptation. Instead of the existing external selection criteria, we applied the internal NMT sentence embedding similarity as the criterion. Empirical results on IWSLT and NIST tasks showed that the proposed method can substantially improve NMT performances and outperform state-of-the-art existing NMT adaptation methods on NMT (even PBSMT) performances.

In addition, we found that the combination of sentence selection and further training has an additional effect, with a fast convergence. In our further work, we will investigate the effect of training data order and batch data selection on NMT training.

Acknowledgments

Thanks a lot for the helpful discussions with Dr. Lemao Liu, Kehai Chen and Dr. Atsushi Fujita. We also appreciate the insightful comments from three anonymous reviewers.

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*. Edinburgh, Scotland, U.K., pages 355–362. http://www.aclweb.org/anthology/D11-1033.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly

learning to align and translate. In *International Conference on Learning Representations*. San Diego. http://arxiv.org/abs/1409.0473.

Mauro Cettolo, Jan Niehues, Sebastian Stüker, Luisa Bentivogli, and Marcello Federico. 2014. Report on the 11th IWSLT evaluation campaign. In *Proceedings of the International Workshop on Spoken Language Translation*. Lake Tahoe, CA, USA, pages 2–17. http://workshop2014.iwslt.org/64.php.

Boxing Chen, Roland Kuhn, George Foster, Colin Cherry, and Fei Huang. 2016. Bilingual methods for adaptive training data selection for machine translation. In *The Twelfth Conference of The Association for Machine Translation in the Americas*. Austin, Texas, pages 93–106. https://amtaweb.org/amta-2016-in-austin-tx/.

Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*. Doha, Qatar, pages 1724–1734. http://www.aclweb.org/anthology/D14-1179.

Kevin Duh, Graham Neubig, Katsuhito Sudoh, and Hajime Tsukada. 2013. Adaptation data selection using neural language models: Experiments in machine translation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Sofia, Bulgaria, pages 678–683. http://www.aclweb.org/anthology/P13-2119.

Nadir Durrani, Hassan Sajjad, Shafiq Joty, Ahmed Abdelali, and Stephan Vogel. 2015. Using joint models for domain adaptation in statistical machine translation. In *Proceedings of MT Summit XV*. Miami, FL, USA, pages 117–130. https://amtaweb.org/mt-summit-xv-proceedings/.

Markus Freitag and Yaser Al-Onaizan. 2016. Fast domain adaptation for neural machine translation. *arXiv preprint arXiv:1612.06897* http://arxiv.org/abs/1612.06897.

Isao Goto, Bin Lu, Ka Po Chow, Eiichiro Sumita, and Benjamin K. Tsou. 2011. Overview of the patent machine translation task at the NTCIR-9 workshop. In *Proceedings of NTCIR-9 Workshop Meeting*. Tokyo, Japan, pages 559–578. http://research.nii.ac.jp/ntcir/.

Cuong Hoang and Khalil Sima'an. 2014a. Latent domain phrase-based models for adaptation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*. Doha, Qatar, pages 566–576. http://www.aclweb.org/anthology/D14-1062.

Cuong Hoang and Khalil Sima'an. 2014b. Latent domain translation models in mix-of-domains haystack. In *Proceedings of the 25th International Conference on Computational Linguistics: Technical Papers*. Dublin, Ireland, pages 1928–1939. http://www.aclweb.org/anthology/C14-1182.

Sébastien Jean, Kyunghyun Cho, Roland Memisevic, and Yoshua Bengio. 2015. On using very large target vocabulary for neural machine translation. 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)*. Beijing, China, pages 1–10. http://www.aclweb.org/anthology/P15-1001.

Shafiq Joty, Hassan Sajjad, Nadir Durrani, Kamla Al-Mannai, Ahmed Abdelali, and Stephan Vogel. 2015. How to avoid unwanted pregnancies: Domain adaptation using neural network models. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Lisbon, Portugal, pages 1259–1270. http://aclweb.org/anthology/D15-1147.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*. Prague, Czech Republic, pages 177–180. http://www.aclweb.org/anthology/P07-2045.

Xiaoqing Li, Jiajun Zhang, and Chengqing Zong. 2016. One sentence one model for neural machine translation. *arXiv preprint* http://arxiv.org/abs/1609.06490.

Minh-Thang Luong and Christopher D Manning. 2015. Stanford neural machine translation systems for spoken language domains. In *Proceedings of the International Workshop on Spoken Language Translation*. Da Nang, Vietnam, pages 76–79. https://nlp.stanford.edu/pubs/luong-manning-iwslt15.pdf.

Haitao Mi, Zhiguo Wang, and Abe Ittycheriah. 2016. Vocabulary manipulation for neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Berlin, Germany, pages 124–129. http://anthology.aclweb.org/P16-2021.

Robert C Moore and William Lewis. 2010. Intelligent selection of language model training data. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Uppsala, Sweden, pages 220–224. http://www.aclweb.org/anthology/P10-2041.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*. Philadelphia, Pennsylvania, pages 311–318. http://www.aclweb.org/anthology/P02-1040.

Rico Sennrich. 2012. Perplexity minimization for translation model domain adaptation in statistical machine translation. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*. Avignon, France, pages 539–549. http://www.aclweb.org/anthology/E12-1055.

Rico Sennrich, Holger Schwenk, and Walid Aransa. 2013. A multi-domain translation model framework for statistical machine translation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Sofia, Bulgaria, pages 832–840. http://www.aclweb.org/anthology/P13-1082.

Zhaopeng Tu, Zhengdong Lu, Yang Liu, Xiaohua Liu, and Hang Li. 2016. Modeling coverage for neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Berlin, Germany, pages 76–85. http://www.aclweb.org/anthology/P16-1008.

Rui Wang, Hai Zhao, Bao-Liang Lu, Masao Utiyama, and Eiichiro. Sumita. 2015. Bilingual continuous-space language model growing for statistical machine translation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 23(7):1209–1220. https://doi.org/10.1109/TASLP.2015.2425220.

Rui Wang, Hai Zhao, Bao-Liang Lu, Masao Utiyama, and Eiichiro Sumita. 2016. Connecting phrase based statistical machine translation adaptation. In *Proceedings of the 26th International Conference on Computational Linguistics: Technical Papers*. Osaka, Japan, pages 3135–3145. http://aclweb.org/anthology/C16-1295.

Xiaolin Wang, Masao Utiyama, Andrew Finch, and Eiichiro Sumita. 2014. Empirical study of unsupervised Chinese word segmentation methods for SMT on large-scale corpora. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Baltimore, Maryland, pages 752–758. http://www.aclweb.org/anthology/P14-2122.

Matthew D Zeiler. 2012. ADADELTA: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701* http://arxiv.org/abs/1212.5701.

Biao Zhang, Deyi Xiong, jinsong su, Hong Duan, and Min Zhang. 2016. Variational neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Austin, Texas, pages 521–530. https://aclweb.org/anthology/D16-1050.