Neural Machine Translation with Source Dependency Representation

Kehai Chen[1]*, **Rui Wang**[2]†, **Masao Utiyama**[2], **Lemao Liu**[3],
Akihiro Tamura[4], **Eiichiro Sumita**[2] and **Tiejun Zhao**[1]
[1]Machine Intelligence & Translation Laboratory, Harbin Institute of Technology
[2]ASTREC, National Institute of Information and Communications Technology (NICT)
[3]Tencent AI Lab
[4]Graduate School of Science and Engineering, Ehime University
{khchen and tjzhao}@hit.edu.cn
{wangrui, mutiyama and eiichiro.sumita}@nict.go.jp
lemaoliu@gmail.com and tamura@cs.ehime-u.ac.jp

Abstract

Source dependency information has been successfully introduced into statistical machine translation. However, there are only a few preliminary attempts for Neural Machine Translation (NMT), such as concatenating representations of source word and its dependency label together. In this paper, we propose a novel attentional NMT with source dependency representation to improve translation performance of NMT, especially on long sentences. Empirical results on NIST Chinese-to-English translation task show that our method achieves 1.6 BLEU improvements on average over a strong NMT system.

1 Introduction

Neural Machine Translation (NMT) (Kalchbrenner and Blunsom, 2013; Bahdanau et al., 2014; Sutskever et al., 2014) relies heavily on source representations, which encode implicitly semantic information of source words by neural networks (Mikolov et al., 2013a,b). Recently, several research works have been proposed to learn richer source representation, such as multi-source information (Zoph and Knight, 2016; Firat et al., 2016), and particularly source syntactic information (Eriguchi et al., 2016; Li et al., 2017; Huadong et al., 2017; Eriguchi et al., 2017), thus improving the performance of NMT.

In this paper, we enhance source representations by dependency information, which can capture source long-distance dependency constraints for word prediction. Actually, source dependency information has been shown greatly effective in Statistical Machine Translation (SMT) (Garmash and Monz, 2014; Kazemi et al., 2015; Hadiwinoto et al., 2016; Chen et al., 2017; Hadiwinoto and Ng, 2017). In NMT, there has been a quite recent preliminary exploration (Sennrich and Haddow, 2016), in which vector representations of source word and its dependency label are simply concatenated as source input, achieving state-of-the-art performance in NMT (Bojar et al., 2016).

In this paper, we propose a novel NMT with source dependency representation to improve translation performance. Compared with the simple approach of vector concatenation, we learn the Source Dependency Representation (**SDR**) to compute dependency context vectors and alignment matrices in a more sophisticated manner, which has the potential to make full use of source dependency information. To this end, we create a dependency unit for each source word to capture long-distance dependency constraints. Then we design an *Encoder* with convolutional architecture to jointly learn SDRs (Section 3) and source dependency annotations, thus computing dependency context vectors and hidden states by a novel double-context based *Decoder* for word prediction (Section 4). Empirical results on NIST Chinese-to-English translation task show that the proposed approach achieves significant gains over the method by Sennrich and Haddow (2016), and thus delivers substantial improvements over the standard attentional NMT (Section 5).

2 Background

An NMT model consists of an *Encoder* process and a *Decoder* process, and hence it is often called *Encoder-Decoder* model (Sutskever et al., 2014; Bahdanau et al., 2014). Typically, each unit of source input $x_j \in (x_1, \ldots, x_J)$ is firstly embedded as a vector V_{x_j}, and then represented as

*Kehai Chen was an internship research fellow at NICT when conducting this work.
†Corresponding author.

Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pages 2829–2835
Copenhagen, Denmark, September 7–11, 2017.

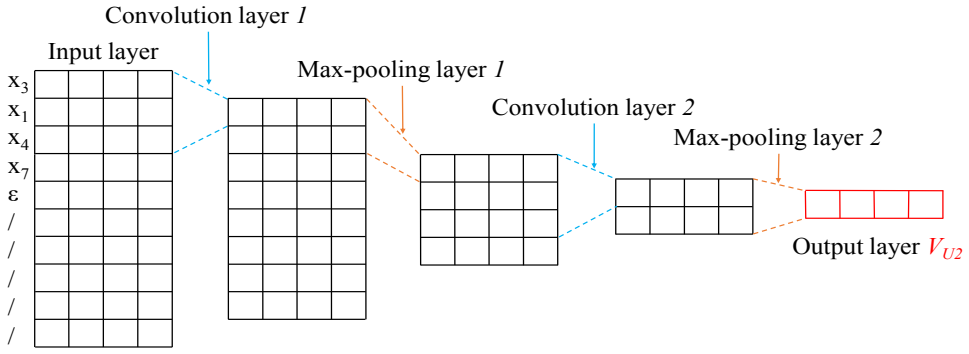

Figure 1: The CNN architecture for learning SRD.

an annotation vector h_j by

$$h_j = f_{enc}(V_{x_j}, h_{j-1}), \quad (1)$$

where f_{enc} is a bidirectional Recurrent Neural Network (**RNN**) (Bahdanau et al., 2014). These annotation vectors $H = (h_1, \ldots, h_J)$ are used to generate the target word in the *Decoder*.

An RNN *Decoder* is used to compute the target word y_i probability by a softmax layer g:

$$p(y_i|y_{<i}, x) = g(\hat{y}_{i-1}, s_i, c_i), \quad (2)$$

where $\hat{y}_{i-1}$ is the previously emitted word, and s_i is an RNN hidden state for the current time step:

$$s_i = \varphi(\hat{y}_{i-1}, s_{i-1}, c_i), \quad (3)$$

and the context vector c_i is computed as a weighted sum of these source annotations h_j:

$$c_i = \sum_{j=1}^{J} \alpha_{ij} h_j, \quad (4)$$

where the normalized alignment weight α_{ij} is computed by

$$\alpha_{ij} = \frac{exp(e_{ij})}{\sum_{k=1}^{J} exp(e_{ik})}, \quad (5)$$

where e_{ij} is an alignment which indicates how well the inputs around position j and the output at the position i match:

$$e_{ij} = f(s_{i-1}, h_j). \quad (6)$$

where f is a feedforward neural network.

3 Source Dependency Representation

In order to capture source long-distance dependency constraints, we extract a dependency unit U_j for each source word x_j from dependency tree, inspired by a dependency-based bilingual composition sequence for SMT (Chen et al., 2017). The extracted U_j is defined as the following:

$$U_j = \langle PA_{x_j}, SI_{x_j}, CH_{x_j} \rangle, \quad (7)$$

where PA_{x_j}, SI_{x_j}, CH_{x_j} denote the parent, siblings and children words of source word x_j in a dependency tree. Take x_2 in Figure 2 as an example, the blue solid box U_2 denotes its dependency unit: $PA_{x_2} = \langle x_3 \rangle$, $SI_{x_2} = \langle x_1, x_4, x_7 \rangle$ and $CH_{x_2} = \langle \varepsilon \rangle$ (no child), that is, $U_2 = \langle x_3, x_1, x_4, x_7, \varepsilon \rangle$.

We design a simplified neural network following Chen et al. (2017)'s Convolutional Neural Network (**CNN**) method, to learn the SDR for each source dependency unit U_j, as shown in Figure 1. Our neural network consists of an input layer, two convolutional layers, two pooling layers and an output layer:

- **Input layer**: the input layer takes words of a dependency unit U_j in the form of embedding vectors $n \times d$, where n is the number of words in a dependency unit and d is vector dimension of each word. In our experiments, we set n to 10,[1] and d is 620. For dependency units shorter than 10, we perform "/" padding at the ending of U_j. For example, the padded U_2 is $\langle x_3, x_1, x_4, x_7, \varepsilon, /, /, /, /, / \rangle$.

[1]We find that 99% of all the source dependency units contain no more than 10 words. So if the length is more than 10, the extra words are abandoned; if the length is less than 10, the rest positions are padded with "/".

- **Convolutional layer**: the first convolution consists of one $3\times d$ convolution kernels (the stride is 1) to output an $(n\text{-}2)\times d$ matrix; the second convolution consists of one $3\times d$ convolution kernels to output a $\frac{n-2}{2}\times d$ matrix.

- **Max-Pooling layer**: the first pooling layer performs row-wise max over the two consecutive rows to output a $\frac{n-2}{4}\times d$ matrix; the second pooling layer performs row-wise max over the two consecutive rows to output a $\frac{n-2}{8}\times d$ matrix.

- **Output layer**: the output layer performs row-wise average based on the output of the second pooling layer to learn a compact d-dimension vector V_{U_j} for U_j. In our experiment, the output of the output layer is $1 \times d$-dimension vector.

It should be noted that the dependency unit is similar to the source dependency feature of Sennrich and Haddow (2016) and the SDR is the same to the source-side representation of Chen et al. (2017). In comparison with Sennrich and Haddow (2016), who concatenate the source dependency labels and word together to enhance the *Encoder* of NMT, we adapt a separate attention mechanism together with a CNN dependency *Encoder*. Compared with Chen et al. (2017), which expands the famous neural network joint model (Devlin et al., 2014) with source dependency information to improve the phrase pair translation probability estimation for SMT, we focus on source dependency information to enhance attention probability estimation and to learn corresponding dependency context and RNN hidden state for improving translation.

4 NMT with SDR

In this section, we propose two novel NMT models **SDRNMT**-*1* and **SDRNMT**-*2*, both of which can make use of source dependency information SDR to enhance *Encoder* and *Decoder* of NMT.

4.1 SDRNMT-*1*

Compared with standard attentional NMT, the *Encoder* of SDRNMT-*1* model consists of a convolutional architecture and an bidirectional RNN, as shown in Figure 2. Therefore, the proposed *Encoder* can not only learn compositional representations for dependency units but also greatly tackle the sparsity issues associated with large dependency units.

Motivated by (Sennrich and Haddow, 2016), we concatenate the V_{x_j} and V_{U_j} as input of the *Encoder*, as shown in the black dotted box in Figure 2. Source annotation vectors are learned based on the concatenated representation with dependency information:

$$h_j = f_{enc}(V_{x_j} : V_{U_j}, h_{j-1}), \tag{8}$$

where ":" denotes the operation of vectors concatenation. Finally, these learned annotation vectors are as the input of the standard NMT *Decoder* to jointly learn alignment and translation. The only difference between our method and (Sennrich and Haddow, 2016)'s method is that they only use dependency label representation instead of V_{U_j}.

4.2 SDRNMT-*2*

In SDRNMT-*1*, a single annotation, learned over concatenating word representation and SDR, is used to compute the context vector and the RNN hidden state for the current time step. To relieve more translation performance for NMT from the SDR, we propose a *double-context* mechanism, as shown in Figure 3.

First, the *Encoder* of SDRNMT-*2* consists of two independent annotations h_j and d_j:

$$\begin{aligned} h_j &= f_{enc}(V_{x_j}, h_{j-1}), \\ d_j &= f_{enc}(V_{U_j}, d_{j-1}), \end{aligned} \tag{9}$$

where $H = [h_1, \cdots, h_J]$ and $D = [d_1, \cdots, d_J]$ encode source sequential and long-distance dependency information, respectively.

The *Decoder* learns the corresponding alignment matrices and context vectors over the H and D, respectively. That is, according to eq.(6), given the previous hidden state s^s_{i-1} and s^d_{i-1}, the current alignments $e^s_{i,j}$ and $e^d_{i,j}$ are computed over source annotation vectors h_j and d_j, respectively:

$$\begin{aligned} e^s_{i,j} &= f(s^s_{i-1} + h_j), \\ e^d_{i,j} &= f(s^d_{i-1} + d_j). \end{aligned} \tag{10}$$

According to eq.(5), we further compute the current alignment $\tilde{\alpha}$:

$$\tilde{\alpha}_{i,j} = \frac{exp(\lambda e^s_{i,j} + (1-\lambda) e^d_{i,j})}{\sum_{j=1}^{J} exp(\lambda e^s_{i,j} + (1-\lambda) e^d_{i,j})}, \tag{11}$$

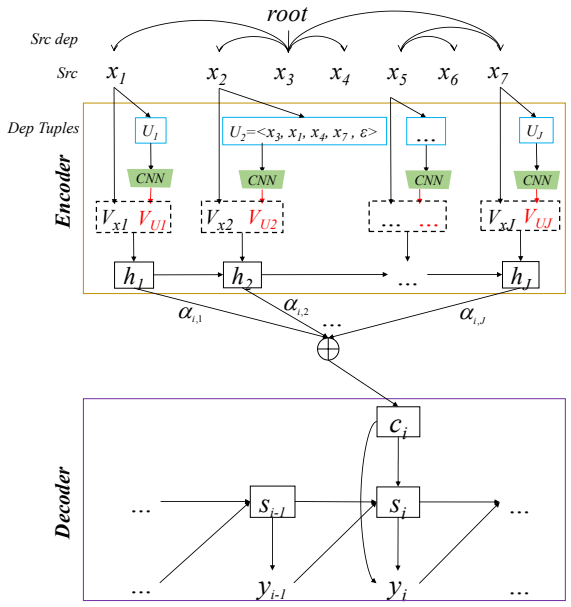

Figure 2: SDRNMT-*1* for the *i*-th time step.

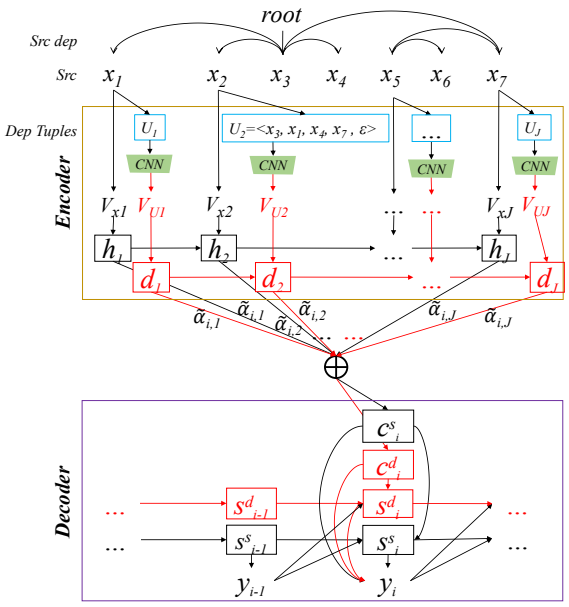

Figure 3: SDRNMT-*2* for the *i*-th time step.

where λ is a hyperparameter[2] to control the importance of H and D. Note that compared with the original alignment model only depending on the sequential annotation vectors H, the alignment weight $\tilde{\alpha}_{i,j}$ jointly compute statistic over source sequential annotation vectors H and dependency annotation vectors D.

The current context vector c_i^s and c_i^d are compute by eq.(4), respectively:

$$c_i^s = \sum_{j=1}^{J} \tilde{\alpha}_{i,j} h_j, \text{and } c_i^d = \sum_{j=1}^{J} \tilde{\alpha}_{i,j} d_j. \quad (12)$$

The current hidden state s_i^s and s_i^d are computed by eq.(3), respectively:

$$\begin{aligned} s_i^s &= \varphi(s_{i-1}^s, y_{i-1}, c_i^s), \\ s_i^d &= \varphi(s_{i-1}^d, y_{i-1}, c_i^d). \end{aligned} \quad (13)$$

Finally, according to eq.(2), the probabilities for the next target word are computed using two hidden states s_i^s and s_i^d, the previously emitted word $\hat{y}_{i-1}$, the current sequential context vector c_i^s and dependency context vector c_{d_i}:

$$p(y_i|y_{<i}, x, T) = g(\hat{y}_{i-1}, s_i^s, s_i^d, c_i^s, c_i^d). \quad (14)$$

5 Experiment

5.1 Setting up

We carry out experiments on Chinese-to-English translation. The training dataset consists of 1.42M sentence pairs extract from LDC corpora.[3] We use the Stanford dependency parser (Chang et al., 2009) to generate the dependency tree for Chinese. We choose the NIST 2002 (MT02) and the NIST 2003-2008 (MT03-08) datasets as the validation set and test sets, respectively. Case-insensitive *4*-gram NIST BLEU score (Papineni et al., 2002) is used as an evaluation metric, and *signtest* (Collins et al., 2005) is as statistical significance test.

The baseline systems include the standard Phrase-Based Statistical Machine Translation (**PBSMT**) implemented in Moses (Koehn et al., 2007) and the standard Attentional NMT (**AttNMT**) (Bahdanau et al., 2014), where only source word representation is utilized. We also compare with a state-of-the-art syntax enhanced NMT method (Sennrich and Haddow, 2016). For a fair comparison, we only utilize dependency information for (Sennrich and Haddow, 2016), called **Sennrich-deponly**. We try our best to re-implement the baseline methods on Nematus toolkit [4] (Sennrich et al., 2017).

For all NMT systems, we limit the source and target vocabularies to 30K, and the maximum sentence length is *80*. The word embedding dimension is *620*,[5] and the hidden layer dimension

[2] λ can be tuned according to a subset FBIS of training data and be set as *0.6* in the experiments.

[3] LDC2002E18, LDC2003E07, LDC2003E14, Hansards portion of LDC2004T07, LDC2004T08, and LDC2005T06.

[4] https://github.com/EdinburghNLP/nematus

[5] For SDRNMT-*1*, the *360* dimensions are from V_{x_j} and the *260* dimensions are from V_{U_j}.

System	Dev (MT02)	MT03	MT04	MT05	MT06	MT08	AVG
PBSMT	33.15	31.02	33.78	30.33	29.62	23.53	29.66
AttNMT	36.31	34.02	37.11	32.86	32.54	25.44	32.40
Sennrich-deponly	36.68	34.51	38.09	33.37	32.96	26.96	32.98
SDRNMT-*1*	36.88	34.98*	38.14	34.61	33.58	27.06	33.32
SDRNMT-*2*	**37.34**	**35.91****	**38.73***	**34.18****	**33.76****	**27.64***	**34.04**

Table 1: Results on NIST Chinese-to-English Translation Task. "*" indicates statistically significant better than "Sennrich-deponly" at p-value < 0.05 and "**" at p-value < 0.01. **AVG** = average BLEU scores for test sets.

is *1000*, and all the layers use the dropout training technique (Hinton et al., 2012). We shuffle training set before training and the mini-batch size is *80*. Training is conducted on a single Tesla P100 GPU. All NMT models train for *15* epochs using ADADELTA (Zeiler, 2012), and the train time is *6* days, which is *25%* slower than the standard NMT.

5.2 Results and Analyses

Table 1 shows the translation performances on test sets measured in BLEU score. The AttNMT significantly outperforms PBSMT by *2.74* BLEU points on average, indicating that it is a strong baseline NMT system. The baseline Sennrich-deponly improves the performance over the AttNMT by *0.58* BLEU points on average. This indicates that the proposed source dependency constraint is beneficial for improving the performance of NMT.

Moreover, SDRNMT-*1* gains improvements of *0.92* and *0.34* BLEU points on average than the AttNMT and Sennrich-deponly. These show that the proposed SDR can more effectively capture source dependency information than vector concatenation. Especially, the proposed SDRNMT-*2* outperforms the AttNMT and Sennrich-deponly on average by *1.64* and *1.03* BLEU points. These verify that the proposed double-context method is effective for word prediction.

5.3 Effect of Translating Long Sentences

We follow (Bahdanau et al., 2014) to group sentences of similar lengths all the test sets (MT03-08), for example, "*40*" indicates that the length of sentences is between 30 and 40, and compute a BLEU score per group. As demonstrated in Figure 4, the proposed models outperform other baseline systems, especially in translating long sentences. These results show that the proposed models can effective encode long-distance dependencies to improve translation.

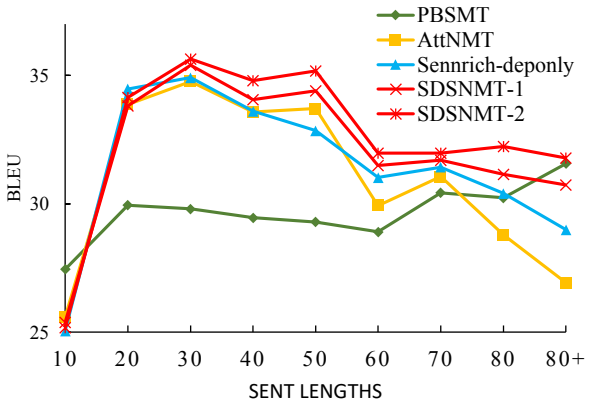

Figure 4: Translation qualities for different sentence lengths.

6 Conclusion and Future Work

In this paper, we explored the source dependency information to improve the performance of NMT. We proposed a novel attentional NMT with source dependency representation to capture source long-distance dependencies. In the future, we will try to exploit a general framework for utilizing richer syntax knowledge.

Acknowledgments

We are grateful to the anonymous reviewers for their insightful comments and suggestions. This work is partially supported by the program "Promotion of Global Communications Plan: Research, Development, and Social Demonstration of Multilingual Speech Translation Technology" of MIC, Japan. Tiejun Zhao is supported by the National Natural Science Foundation of China (NSFC) via grant 91520204 and National High Technology Research & Development Program of China (863 program) via grant 2015AA015405.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473.

Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, Varvara Logacheva, Christof Monz, Matteo Negri, Aurelie Neveol, Mariana Neves, Martin Popel, Matt Post, Raphael Rubino, Carolina Scarton, Lucia Specia, Marco Turchi, Karin Verspoor, and Marcos Zampieri. 2016. Findings of the 2016 conference on machine translation. In *Proceedings of the First Conference on Machine Translation*, pages 131–198, Berlin, Germany. Association for Computational Linguistics.

Pi-Chuan Chang, Huihsin Tseng, Dan Jurafsky, and Christopher D. Manning. 2009. Discriminative reordering with Chinese grammatical relations features. In *Proceedings of the Third Workshop on Syntax and Structure in Statistical Translation at NAACL HLT 2009*, pages 51–59, Boulder, Colorado. Association for Computational Linguistics.

Kehai Chen, Tiejun Zhao, Muyun Yang, and Lemao Liu. 2017. Translation prediction with source dependency-based context representation. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, pages 3166–3172, California, USA. AAAI Press.

Michael Collins, Philipp Koehn, and Ivona Kucerova. 2005. Clause restructuring for statistical machine translation. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*, pages 531–540, Ann Arbor, Michigan. Association for Computational Linguistics.

Jacob Devlin, Rabih Zbib, Zhongqiang Huang, Thomas Lamar, Richard Schwartz, and John Makhoul. 2014. Fast and robust neural network joint models for statistical machine translation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1370–1380, Baltimore, Maryland. Association for Computational Linguistics.

Akiko Eriguchi, Kazuma Hashimoto, and Yoshimasa Tsuruoka. 2016. Tree-to-sequence attentional neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 823–833, Berlin, Germany. Association for Computational Linguistics.

Akiko Eriguchi, Yoshimasa Tsuruoka, and Kyunghyun Cho. 2017. Learning to parse and translate improves neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, Vancouver, Canada. Association for Computational Linguistics.

Orhan Firat, Kyunghyun Cho, and Yoshua Bengio. 2016. Multi-way, multilingual neural machine translation with a shared attention mechanism. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 866–875, San Diego, California. Association for Computational Linguistics.

Ekaterina Garmash and Christof Monz. 2014. Dependency-based bilingual language models for reordering in statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1689–1700, Doha, Qatar. Association for Computational Linguistics.

Christian Hadiwinoto, Yang Liu, and Hwee Tou Ng. 2016. To swap or not to swap? exploiting dependency word pairs for reordering in statistical machine translation. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, pages 2943–2949, Arizona, USA. AAAI Press.

Christian Hadiwinoto and Hwee Tou Ng. 2017. A dependency-based neural reordering model for statistical machine translation. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, California, USA. AAAI Press.

Geoffrey E. Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2012. Improving neural networks by preventing co-adaptation of feature detectors. *CoRR*.

Chen Huadong, Huang Shujian, Chiang David, and Chen Jiajun. 2017. Improved neural machine translation with a syntax-aware encoder and decoder. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, Vancouver, Canada. Association for Computational Linguistics.

Nal Kalchbrenner and Phil Blunsom. 2013. Recurrent continuous translation models. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1700–1709, Seattle, Washington, USA. Association for Computational Linguistics.

Arefeh Kazemi, Antonio Toral, Andy Way, Amirhassan Monadjemi, and Mohammadali Nematbakhsh. 2015. Dependency-based reordering model for constituent pairs in hierarchical smt. In *Proceedings of the 18th Annual Conference of the European Association for Machine Translation*, pages 43–50, Antalya, Turkey. Association for Computational Linguistics.

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*, pages 177–180, Prague, Czech Republic. Association for Computational Linguistics.

Junhui Li, Xiong Deyi, Tu Zhaopeng, Zhu Muhua, and Zhou Guodong. 2017. Modeling source syntax for neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, Vancouver, Canada. Association for Computational Linguistics.

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

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems*, pages 3111–3119, USA. Curran Associates Inc.

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.

Rico Sennrich, Orhan Firat, Kyunghyun Cho, Alexandra Birch, Barry Haddow, Julian Hitschler, Marcin Junczys-Dowmunt, Samuel Läubli, Antonio Valerio Miceli Barone, Jozef Mokry, and Maria Nadejde. 2017. Nematus: a toolkit for neural machine translation. In *Proceedings of the Software Demonstrations of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pages 65–68, Valencia, Spain. Association for Computational Linguistics.

Rico Sennrich and Barry Haddow. 2016. Linguistic input features improve neural machine translation. In *Proceedings of the First Conference on Machine Translation*, pages 83–91, Berlin, Germany. Association for Computational Linguistics.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems*, pages 3104–3112, Cambridge, MA, USA. MIT Press.

Matthew D. Zeiler. 2012. ADADELTA: an adaptive learning rate method. *CoRR*, abs/1212.5701.

Barret Zoph and Kevin Knight. 2016. Multi-source neural translation. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 30–34, San Diego, California. Association for Computational Linguistics.