Hybrid Neural Network Alignment and Lexicon Model in Direct HMM for Statistical Machine Translation

Weiyue Wang, Tamer Alkhouli, Derui Zhu, Hermann Ney
Human Language Technology and Pattern Recognition, Computer Science Department
RWTH Aachen University, 52056 Aachen, Germany
<surname>@i6.informatik.rwth-aachen.de

Abstract

Recently, the neural machine translation systems showed their promising performance and surpassed the phrase-based systems for most translation tasks. Retreating into conventional concepts machine translation while utilizing effective neural models is vital for comprehending the leap accomplished by neural machine translation over phrase-based methods. This work proposes a direct hidden Markov model (HMM) with neural network-based lexicon and alignment models, which are trained jointly using the Baum-Welch algorithm. The direct HMM is applied to rerank the n-best list created by a state-of-the-art phrase-based translation system and it provides improvements by up to 1.0% BLEU scores on two different translation tasks.

1 Introduction

The hidden Markov model (HMM) was first introduced to statistical machine translation for addressing the word alignment problem (Vogel et al., 1996). Then the HMM-based approach was widely used along with the IBM models (Brown et al., 1993) for aligning the source and target words. In the conventional approach, the Bayes' theorem is used and the HMM is applied to the inverse translation model

$$\begin{aligned}\Pr(e_1^I|f_1^J) &= \Pr(e_1^I)\cdot\Pr(f_1^J|e_1^I)\\ &= \sum_{a_1^J}\Pr(f_1^J,a_1^J|e_1^I)\end{aligned} \quad (1)$$

In this case, as a part of a noisy channel model, the marginalisation becomes intractable for every e.

This work proposes a novel concept focusing on direct HMM for $\Pr(e_1^I|f_1^J)$, in which the alignment direction is from target to source positions. This specific property allows us to introduce dependencies into the translation model that take the full source sentence into account. This aspect will be important for the future decoder to be developed. The lexicon and alignment probabilities in the HMM are modeled using feed-forward neural networks (FFNN) and they are trained jointly. The trained HMM is then applied for reranking the n-best lists created by a state-of-the-art open source phrase-based translation system. The experiments are conducted on the IWSLT 2016 German→English and BOLT Chinese→English translation tasks. The FFNN-based hybrid HMM provides improvements by up to 1.0% BLEU scores.

2 Related Work

In order to discuss related work, we will consider the following two key concepts that are essential for the work to be presented:

- **Neural lexicon and alignment models**

  The idea of using neural networks for lexicon modeling is not new (Schwenk, 2012; Sundermeyer et al., 2014; Devlin et al., 2014). Apart from differences in the neural network architecture, the important difference to this work is that those approaches did not include the concepts of HMM models and end-to-end training. In addition to neural lexicon modeling, (Alkhouli et al., 2016) also applied a neural network for alignment modeling like this work, but their training procedure was based on the maximum approximation and on predefined GIZA++ (Och and Ney, 2003) alignments.

Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Short Papers), pages 125–131
Vancouver, Canada, July 30 - August 4, 2017. ©2017 Association for Computational Linguistics
https://doi.org/10.18653/v1/P17-2020

There were other studies that focused on feature-rich alignment models (Blunsom and Cohn, 2006; Berg-Kirkpatrick et al., 2010; Dyer et al., 2011), but those studies did not use a neural network to automatically learn features (as we do in this work). (Yang et al., 2013) used neural network-based lexicon and alignment models inside the HMM alignment model, but they model alignments using a simple distortion model that has no dependence on lexical context. Their goal was to improve the alignment quality in the context of a phrase-based translation system. However, apart from (Dyer et al., 2011), no results on translation were reported.

The idea of using neural networks is the basis of the state-of-the-art attention-based approach to machine translation (Bahdanau et al., 2015; Luong et al., 2015). However, that approach is not based on the principle of an explicit and separate lexicon model.

- **End-to-end training**

  The HMM in combination with the neural translation model lends itself to what is usually called end-to-end training. The training criterion is the logarithm of the target sentence posterior probability. This criterion results in a specific training algorithm that can be interpreted as a combination of forward-backward algorithm (as in EM style training of HHMs) and backpropagation. To the best of our knowledge, this end-to-end training has not been considered before for machine translation. In the context of signal processing and recognition, the connectionist temporal classification (CTC) approach (Graves et al., 2006) leads to a similar training procedure. (Tran et al., 2016) studied neural networks for unsupervised training for a part-of-speech tagging task. In their approach, the training criterion for this problem results in a combination of EM framework and backpropagation, which has a certain similarity to the training algorithm for translation as presented in this work.

3 Definition of neural network-based HMM

Similar to hidden alignments $a_j = j \to i$ between the source string $f_1^J = f_1 ... f_j ... f_J$ and the target string $e_1^I = e_1 ... e_i ... e_I$ in the conventional HMM, we define the alignments in direct HMM as $b_i = i \to j$. Then the model can be defined as:

$$\Pr(e_1^I | f_1^J) = \sum_{b_1^I} \Pr(e_1^I, b_1^I | f_1^J) \tag{2}$$

$$\begin{aligned} &\Pr(e_1^I, b_1^I | f_1^J) \\ &= \prod_{i=1}^{I} p(e_i, b_i | b_1^{i-1}, e_1^{i-1}, f_1^J) \\ &= \prod_{i=1}^{I} \underbrace{p(e_i | b_1^i, e_1^{i-1}, f_1^J)}_{\text{lexicon model}} \cdot \underbrace{p(b_i | b_1^{i-1}, e_1^{i-1}, f_1^J)}_{\text{alignment model}} \end{aligned} \tag{3}$$

Our feed-forward alignment model has the same architecture (Figure 1) as the one proposed in (Alkhouli et al., 2016). Thus the alignment probability can be modeled by:

$$p(b_i | b_1^{i-1}, e_1^{i-1}, f_1^J) = p(\Delta_i | f_{b_{i-1}-\gamma_m}^{b_{i-1}+\gamma_m}, e_{i-n}^{i-1}) \tag{4}$$

where $\gamma_m = \frac{m-1}{2}$ and m indicates the window size. $\Delta_i = b_i - b_{i-1}$ denotes the jump from the predecessor position to the current position. Thus, the jump over the source is estimated based on a m-words source context window and n predecessor target words.

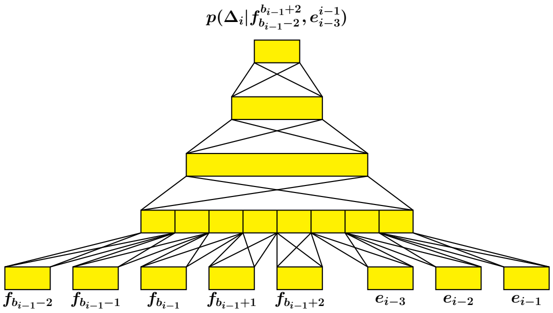

Figure 1: A feed-forward alignment neural network with 3 target history words, 5-gram source window, a projection layer, 2 non-linear hidden layers and a small output layer to predict jumps.

For the lexicon model, we assume a similar dependence as in the alignment model with a shift, namely on the source words within a window centred on the aligned source word and n predecessor target words. To overcome the high costs of the softmax function for large vocabularies, we adopt the class-factored output layer consisting of a class layer and a word layer (Goodman, 2001; Morin and Bengio, 2005). The model in this case is defined as

$$\begin{aligned}&p(e_i|b_1^i, e_1^{i-1}, f_1^J)\\&= p(e_i|f_{b_i-\gamma_m}^{b_i+\gamma_m}, e_{i-n}^{i-1})\\&= p(e_i|c(e_i), f_{b_i-\gamma_m}^{b_i+\gamma_m}, e_{i-n}^{i-1}) \cdot p(c(e_i)|f_{b_i-\gamma_m}^{b_i+\gamma_m}, e_{i-n}^{i-1})\end{aligned} \tag{5}$$

where c denotes a word mapping that assigns each target word to a single class, where the number of classes is chosen to be much smaller than the vocabulary size. The lexicon model architecture is shown in Figure 2.

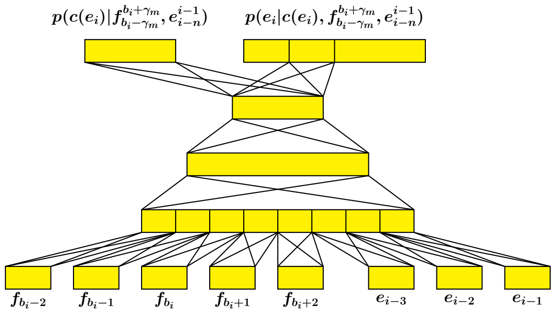
$p(c(e_i)|f_{b_i-\gamma_m}^{b_i+\gamma_m}, e_{i-n}^{i-1})$
$p(e_i|c(e_i), f_{b_i-\gamma_m}^{b_i+\gamma_m}, e_{i-n}^{i-1})$
f_{b_i-2}
f_{b_i-1}
f_{b_i}
f_{b_i+1}
f_{b_i+2}
e_{i-3}
e_{i-2}
e_{i-1}

Figure 2: A feed-forward lexicon neural network with the same structure as the alignment model, except a class-factored output layer.

4 Training

The training data of the direct HMM are the source and target sequences, without any alignment information. In the training of direct HMM including neural network-based models, the weights have to be updated along with the posterior probabilities calculated by the Baum-Welch algorithm. Similar to the training procedure used in (Berg-Kirkpatrick et al., 2010), we apply the EM algorithm and define the auxiliary function as

$$\begin{aligned}&Q(\theta;\hat{\theta})\\&= \sum_{b_1^I} p(b_1^I|f_1^J, e_1^I, \theta) \log p(e_1^I, b_1^I|f_1^J, \hat{\theta})\\&= \sum_{b_1^I} p(b_1^I|f_1^J, e_1^I, \theta) \sum_{i=1}^{I} [\log p(e_i|f_{b_i-\gamma_m}^{b_i+\gamma_m}, e_{i-n}^{i-1}, \hat{\alpha})\\&\qquad\qquad + \log p(\Delta_i|f_{b_{i-1}-\gamma_m}^{b_{i-1}+\gamma_m}, e_{i-n}^{i-1}, \hat{\beta})]\\&= \sum_i \sum_j p_i(j|e_1^I, f_1^J, \theta) \log p(e_i|f_{j-\gamma_m}^{j+\gamma_m}, e_{i-n}^{i-1}, \hat{\alpha})\\&\quad + \sum_i \sum_{j'} \sum_j p_i(j', j|e_1^I, f_1^J, \theta) \log p(\Delta_i|f_{j'-\gamma_m}^{j'+\gamma_m}, e_{i-n}^{i-1}, \hat{\beta})\end{aligned} \tag{6}$$

where $\hat{\theta} = \{\hat{\alpha}, \hat{\beta}\}$, $j' = b_{i-1}$ and

$$p_i(j|e_1^I, f_1^J, \theta) = \sum_{b_1^I : b_i = j} p(b_1^I|e_1^I, f_1^J, \theta) \tag{7}$$

Then the parameters can be separated for lexicon model and alignment model:

$$Q(\theta;\hat{\theta}) = Q_{\text{lex}}(\theta;\hat{\alpha}) + Q_{\text{align}}(\theta;\hat{\beta}) \tag{8}$$

where

$$\begin{aligned}\frac{\partial Q_{\text{lex}}(\theta,\hat{\alpha})}{\partial\hat{\alpha}} = \sum_i \sum_j &\overbrace{p_i(j|e_1^I, f_1^J, \theta)}^{\text{forward-backward algorithm}}\\&\underbrace{\cdot \frac{\partial}{\partial\hat{\alpha}} \log p(e_i|f_{j-\gamma_m}^{j+\gamma_m}, e_{i-n}^{i-1}, \hat{\alpha})}_{\text{backpropagation}}\end{aligned} \tag{9}$$

$$\begin{aligned}\frac{\partial Q_{\text{align}}(\theta,\hat{\beta})}{\partial\hat{\beta}} = \sum_i \sum_{j'} \sum_j &\overbrace{p_i(j', j|e_1^I, f_1^J, \theta)}^{\text{forward-backward algorithm}}\\&\underbrace{\cdot \frac{\partial}{\partial\hat{\beta}} \log p(\Delta_i|f_{j'-\gamma_m}^{j'+\gamma_m}, e_{i-n}^{i-1}, \hat{\beta})}_{\text{backpropagation}}\end{aligned} \tag{10}$$

From Equations (9) and (10) we can observe that the marginalisation of hidden alignments ($\sum_j p_i(j|e_1^I, f_1^J, \theta)$) is the only difference compared to the derivative of neural network training based on word-aligned data. In this approach we iterate over all source positions and the word alignment toolkit such as GIZA++ is not required. Furthermore, the word-aligned data generated e.g. by GIZA++ might contain unaligned and multiply aligned words, which make the data difficult to use for training neural networks. Thus the heuristic-based approaches (Sundermeyer et al., 2014; Devlin et al., 2014) have to be used in order to guarantee the one-on-one alignments, which may negatively influence the quality of the alignments. By contrast, the neural network-based HMM is not constrained by these heuristics. In addition, even though the training process of the direct HMM takes more time than the neural network training on the word-aligned data, we should note that generating the word-aligned data using GIZA++ is also a time-consuming process.

In general, our training procedure can be summarized as follows:

1. One iteration IBM-1 model training to create lexicon table for initializing the forward-backward table.
2. In the first epoch, for each sentence pair calculate and save the entire table of posterior probabilities $p_i(b|e_1^I, f_1^J)$ (also $p_i(b', b|e_1^I, f_1^J)$ for alignment model) using forward-backward algorithm based on the results of IBM-1 model.
3. Training neural network lexicon and alignment models based on the posterior probabilities.
4. From the second epoch onwards:
   (a) For each sentence pair, calculating the posterior probabilities based on the lexicon and alignment probabilities estimated by neural network models.
   (b) Updating weights of neural networks based on the posterior probabilities.
   (c) Repeating step 4 until the perplexity converges.

In this work the IBM-1 initialization is required. We tried to train neural network models from scratch, but the perplexity converges towards a bad local minimum and gets stuck in it. We also attempted other heuristics for initialization, such as assigning probability 0.9 to diagonal alignments and spreading the left 0.1 evenly among other source positions. The resulted perplexity is much higher compared to initializing using IBM-1.

5 Experimental Results

The experiments are conducted on the IWSLT 2016 German→English and BOLT Chinese→English translation tasks, which consist of $20M$ and $4M$ parallel sentence pairs respectively. The feed-forward neural network alignment and lexicon models are jointly trained on the subset of about $200K$ sentence pairs. As an initial research of this topic, our new model is only applied for reranking n-best lists created by a phrase-based decoder. The maximum size of the n-best lists is 500. The translation quality is evaluated by case-insensitive BLEU (Papineni et al., 2002) and TER (Snover et al., 2006) metrics using MultEval (Clark et al., 2011). The scaling factors are tuned with MERT (Och, 2003) with BLEU as optimization criterion on the development sets. For the translation experiments, the averaged scores are presented on the development set from three optimization runs.

Our direct HMM consists of a feed-forward neural network lexicon model with following configuration:

- Five one-hot input vectors for source words and three for target words
- Projection layer size 100 for each word
- Two non-linear hidden layers with 1000 and 500 nodes respectively
- A class-factored output layer with 1000 singleton classes dedicated to the most frequent words, and 1000 classes shared among the rest of the words.

and a feed-forward neural network alignment model with the same configuration as the lexicon model, except a small output layer with 201 nodes, which reflects that the aligned position can jump within the scope from -100 to 100 (Alkhouli et al., 2016).

We conducted experiments on the source and target window size of both network models. Larger source and target windows could not provide significant improvements on BLEU scores, at least for rescoring experiments.

The model is applied for reranking the n-best lists created by the Jane toolkit (Vilar et al., 2010; Wuebker et al., 2012) with a log-linear framework containing phrasal and lexical smoothing models for both directions, word and phrase penalties, a distance-based reordering model, enhanced low frequency features (Chen et al., 2011), a hierarchical reordering model (Galley and Manning, 2008), a word class language mode (Wuebker et al., 2013) and an n-gram language model. The word alignments used for the training of phrase-tables are generated by GIZA++, which performs the alignment training sequentially for IBM-1, HMM and IBM-4. More details about our phrase-based baseline system can be found in (Peter et al., 2015).

The experimental results are demonstrated in Table 1. The rescoring experiments are conducted by adding HMM probability as feature and tuned with MERT. The applied attention-based neural network is a neural machine translation system similar to (Bahdanau et al., 2015). The decoder and encoder word embeddings are of size 620, the encoder uses a bidirectional layer with 1000 LSTMs (Hochreiter and Schmidhuber, 1997) to encode the source side. A layer with 1000 LSTMs

Table 1: Experimental results of rescoring using neural network-based direct HMM. The model with *sum* denotes the system proposed in this work, while the model with *Viterbi* denotes the model with the same neural network structure, which was trained based on the word-aligned data (alignments generated by GIZA++) (Alkhouli et al., 2016). Improvements by systems marked by $*$ have a 95% statistical significance from the *NN-based direct HMM (Viterbi)* system, whereas † denotes the 95% statistical significant improvements with respect to the *attention-based system* in rescoring. [1] was used in reranking the n-best lists, while [2] denotes the stand-alone attention-based decoder.

	IWSLT				BOLT			
	TEDX.tst.2014		MSLT.dev2016		DEV12		P1R6	
	BLEU[%]	TER[%]	BLEU[%]	TER[%]	BLEU[%]	TER[%]	BLEU[%]	TER[%]
Phrase-based translation system	25.9	55.7	39.7	39.8	17.9	68.3	17.1	67.4
+ NN-based direct HMM (Viterbi)	26.6	54.9	40.2	39.3	18.5	67.6	17.8	66.8
+ NN-based direct HMM (sum)	26.9	**54.7**	**40.6***	**38.9***†	18.9*	67.3*†	18.3*	66.4
+ attention-based system [1]	26.8	55.0	40.4	39.3	18.9*	67.6	18.0	66.4*
Stand-alone attention-based system [2]	**27.0**	54.8	40.4	39.2	**19.6***†	**67.0***†	**18.5***†	**66.1***

is used by the decoder. The data is converted into subword units using byte pair encoding with 20000 operations (Sennrich et al., 2016). During training a batch size of 50 is used. More details about our neural machine translation system can be found in (Peter et al., 2016).

With n-best rescoring, all neural network-based systems achieve significant improvements over the phrase-based system. The neural network-based HMMs provide promising performance, even with simple feed-forward neural networks. The direct HMM trained by the EM procedure with marginalizing the hidden alignments outperformed the same model trained on the word-aligned data. For the rescoring tasks, it provides comparable performance with the attention-based network. The neural network-based HMM also helps the phrase-based system achieve comparable results with the stand-alone attention-based system on the German→English task.

6 Conclusion and Future Work

This work aims to close the gap between the conventional word alignment models and the novel neural machine translation. The proposed direct HMM consists of neural network-based alignment and lexicon models, both models are trained jointly and without any alignment information. With the simple feed-forward neural network models, the HMM model already provides promising results and significantly improves the strong phrase-based translation system.

As future work, we are searching for alternatives to initialize the training instead of using IBM-1. We will investigate recurrent model structures, such as the LSTM representation for source and target word embeddings (Luong et al., 2015). In addition to the network structure, we will implement a stand-alone decoder based on this novel model. The first step would be to apply maximum approximation for the search problem as elucidated in (Yu et al., 2017). Then we plan to investigate heuristics for marginalizing the hidden alignment during search.

Acknowledgments

The work reported in this paper results from two projects, SEQCLAS and QT21. SEQCLAS has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme under grant agreement n^o 694537. QT21 has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement n^o 645452. The work reflects only the authors' views and neither the European Commission nor the European Research Council Executive Agency are responsible for any use that may be made of the information it contains.

Tamer Alkhouli was partly funded by the 2016 Google PhD Fellowship for North America, Europe and the Middle East.

The authors would like to thank Jan-Thorsten Peter for providing the attention-based neural network models.

References

Tamer Alkhouli, Gabriel Bretschner, Jan-Thorsten Peter, Mohammed Hethnawi, Andreas Guta, and Hermann Ney. 2016. Alignment-Based Neural Machine Translation. In *Proceedings of the ACL 2016 First Conference on Machine Translation (WMT 2016)*. Berlin, Germany, pages 54–65.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural Machine Translation by Jointly Learning to Align and Translate. In *Proceedings of the 3rd International Conference on Learning Representations*. San Diego, CA, USA.

Taylor Berg-Kirkpatrick, Alexandre Bouchard-Côté, John DeNero, and Dan Klein. 2010. Painless Unsupervised Learning with Features. In *Proceedings of the 2010 Annual Conference of the North American Chapter of the ACL*. Los Angeles, CA, USA, pages 582–590.

Phil Blunsom and Trevor Cohn. 2006. Discriminative Word Alignment with Conditional Random Fields. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the ACL*. Sydney, Australia, pages 65–72.

Peter F. Brown, Vincent J. Della Pietra, Stephen A. Della Pietra, and Robert L. Mercer. 1993. The Mathematics of Statistical Machine Translation: Parameter Estimation. *Computational Linguistics* 19(2):263–311.

Boxing Chen, Roland Kuhn, George Foster, and Howard Johnson. 2011. Unpacking and Transforming Feature Functions: New Ways to Smooth Phrase Tables. In *Proceedings of MT Summit XIII*. Xiamen, China, pages 269–275.

Jonathan H. Clark, Chris Dyer, Alon Lavie, and Noah A. Smith. 2011. Better Hypothesis Testing for Statistical Machine Translation: Controlling for Optimizer Instability. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*. Portland, OR, USA, pages 176–181.

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*. Baltimore, MD, USA, pages 1370–1380.

Chris Dyer, Jonathan Clark, Alon Lavie, and Noah A. Smith. 2011. Unsupervised Word Alignment with Arbitrary Features. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*. Portland, OR, USA, pages 409–419.

Michel Galley and Christopher D. Manning. 2008. A simple and effective hierarchical phrase reordering model. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*. Honolulu, HI, USA, pages 848–856.

Joshua Goodman. 2001. Classes for fast maximum entropy training. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*. Salt Lake City, UT, USA, pages 561–564.

Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. 2006. Connectionist Temporal Classification: Labelling Unsegmented Sequence Data with Recurrent Neural Networks. In *Proceedings of the 23rd International Conference on Machine Learning*. Pittsburgh, PA, USA, pages 369–376.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation* 9(8):1735–1780.

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

Frederic Morin and Yoshua Bengio. 2005. Hierarchical probabilistic neural network language model. In *Proceedings of the international workshop on artificial intelligence and statistics*. Barbados, pages 246–252.

Franz Josef Och. 2003. Minimum Error Rate Training in Statistical Machine Translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*. Sapporo, Japan, pages 160–167.

Franz Josef Och and Hermann Ney. 2003. A Systematic Comparison of Various Statistical Alignment Models. *Computational Linguistics* 29:19–51.

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, PA, USA, pages 311–318.

Jan-Thorsten Peter, Andreas Guta, Nick Rossenbach, Miguel Graça, and Hermann Ney. 2016. The RWTH Aachen Machine Translation System for IWSLT 2016. In *International Workshop on Spoken Language Translation*. Seattle, WA, USA.

Jan-Thorsten Peter, Farzad Toutounchi, Joern Wuebker, and Hermann Ney. 2015. The RWTH Aachen German-English Machine Translation System for WMT 2015. In *EMNLP 2015 Tenth Workshop on Statistical Machine Translation*. Lisbon, Portugal, pages 158–163.

Holger Schwenk. 2012. Continuous Space Translation Models for Phrase-Based Statistical Machine Translation. In *Proceedings of the 24th International Conference on Computational Linguistics*. Mumbai, India, pages 1071–1080.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural Machine Translation of Rare Words with Subword Units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*. Berlin, Germany, pages 1715–1725.

Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of the Conference of the Association for Machine Translation in the Americas*. Cambridge, MA, USA, pages 223–231.

Martin Sundermeyer, Tamer Alkhouli, Joern Wuebker, and Hermann Ney. 2014. Translation Modeling with Bidirectional Recurrent Neural Networks. In *Conference on Empirical Methods in Natural Language Processing*. Doha, Qatar, pages 14–25.

Ke Tran, Yonatan Bisk, Ashish Vaswani, Daniel Marcu, and Kevin Knight. 2016. Unsupervised Neural Hidden Markov Models. In *Proceedings of the Workshop on Structured Prediction for NLP*. Austin, TX, USA, pages 63–71.

David Vilar, Daniel Stein, Matthias Huck, and Hermann Ney. 2010. Jane: Open Source Hierarchical Translation, Extended with Reordering and Lexicon Models. In *ACL 2010 Joint Fifth Workshop on Statistical Machine Translation and Metrics MATR*. Uppsala, Sweden, pages 262–270.

Stephan Vogel, Hermann Ney, and Christoph Tillmann. 1996. HMM-based Word Alignment in Statistical Translation. In *Proceedings of the 16th Conference on Computational Linguistics - Volume 2*. Copenhagen, Denmark, COLING '96, pages 836–841.

Joern Wuebker, Matthias Huck, Stephan Peitz, Malte Nuhn, Markus Freitag, Jan-Thorsten Peter, Saab Mansour, and Hermann Ney. 2012. Jane 2: Open Source Phrase-based and Hierarchical Statistical Machine Translation. In *International Conference on Computational Linguistics*. Mumbai, India, pages 483–491.

Joern Wuebker, Stephan Peitz, Felix Rietig, and Hermann Ney. 2013. Improving Statistical Machine Translation with Word Class Models. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*. Seattle, WA, USA, pages 1377–1381.

Nan Yang, Shujie Liu, Mu Li, Ming Zhou, and Nenghai Yu. 2013. Word Alignment Modeling with Context Dependent Deep Neural Network. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*. Sofia, Bulgaria, pages 166–175.

Lei Yu, Phil Blunsom, Chris Dyer, Edward Grefenstette, and Tomáš Kočiský. 2017. The Neural Noisy Channel. In *Proceedings of the 5th International Conference on Learning Representations*. Toulon, France.