Neural Network Based Bilingual Language Model Growing for Statistical Machine Translation

Rui Wang[1,3,*], **Hai Zhao**[1,3], **Bao-Liang Lu**[1,3], **Masao Utiyama**[2] **and Eiichro Sumita**[2]
[1]Center for Brain-Like Computing and Machine Intelligence,
Department of Computer Science and Engineering,
Shanghai Jiao Tong University, Shanghai, 200240, China
[2]Multilingual Translation Laboratory, MASTAR Project,
National Institute of Information and Communications Technology,
3-5 Hikaridai, Keihanna Science City, Kyoto, 619-0289, Japan
[3]Key Laboratory of Shanghai Education Commission for Intelligent Interaction
and Cognitive Engineering, Shanghai Jiao Tong University, Shanghai, 200240, China
wangrui.nlp@gmail.com, {zhaohai, blu}@cs.sjtu.edu.cn,
{mutiyama, eiichiro.sumita}@nict.go.jp

Abstract

Since larger n-gram Language Model (LM) usually performs better in Statistical Machine Translation (SMT), how to construct efficient large LM is an important topic in SMT. However, most of the existing LM growing methods need an extra monolingual corpus, where additional LM adaption technology is necessary. In this paper, we propose a novel neural network based bilingual LM growing method, only using the bilingual parallel corpus in SMT. The results show that our method can improve both the perplexity score for LM evaluation and BLEU score for SMT, and significantly outperforms the existing LM growing methods without extra corpus.

1 Introduction

'*Language Model (LM) Growing*' refers to adding n-grams outside the corpus together with their probabilities into the original LM. This operation is useful as it can make LM perform better through letting it become larger and larger, by only using a small training corpus.

There are various methods for adding n-grams selected by different criteria from a monolingual corpus (Ristad and Thomas, 1995; Niesler and Woodland, 1996; Siu and Ostendorf, 2000; Siivola et al., 2007). However, all of these approaches need additional corpora. Meanwhile the extra corpora from different domains will not result in better LMs (Clarkson and Robinson, 1997; Iyer et al., 1997; Bellegarda, 2004; Koehn and Schroeder, 2007). In addition, it is very difficult or even impossible to collect an extra large corpus for some special domains such as the TED corpus (Cettolo et al., 2012) or for some rare languages. Therefore, to improve the performance of LMs, without assistance of extra corpus, is one of important research topics in SMT.

Recently, Continues Space Language Model (CSLM), especially Neural Network based Language Model (NNLM) (Bengio et al., 2003; Schwenk, 2007; Mikolov et al., 2010; Le et al., 2011), is being actively used in SMT (Schwenk et al., 2006; Son et al., 2010; Schwenk, 2010; Schwenk et al., 2012; Son et al., 2012; Niehues and Waibel, 2012). One of the main advantages of CSLM is that it can more accurately predict the probabilities of the n-grams, which are not in the training corpus. However, in practice, CSLMs have not been widely used in the current SMT systems, due to their too high computational cost.

Vaswani and colleagues (2013) propose a method for reducing the training cost of CSLM and apply it to SMT decoder. However, they do not show their improvement for decoding speed, and their method is still slower than the n-gram LM. There are several other methods for attempting to implement neural network based LM or translation model for SMT (Devlin et al., 2014; Liu et al., 2014; Auli et al., 2013). However, the decoding speed using n-gram LM is still state-of-the-art one. Some approaches calculate the probabilities of the n-grams n-grams before decoding, and store them in the n-gram format (Wang et al., 2013a; Arsoy et al., 2013; Arsoy et al., 2014). The '*converted CSLM*' can be directly used in SMT. Though more n-grams which are not in the train-

[*]Part of this work was done as Rui Wang visited in NICT.

Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 189–195,
October 25-29, 2014, Doha, Qatar. ©2014 Association for Computational Linguistics

ing corpus can be generated by using some of these '*converting*' methods, these methods only consider the monolingual information, and do not take the bilingual information into account.

We observe that the translation output of a phrase-based SMT system is concatenation of phrases from the phrase table, whose probabilities can be calculated by CSLM. Based on this observation, a novel neural network based bilingual LM growing method is proposed using the '*connecting phrases*'. The remainder of this paper is organized as follows: In Section 2, we will review the existing CSLM converting methods. The new neural network based bilingual LM growing method will be proposed in Section 3. In Section 4, the experiments will be conducted and the results will be analyzed. We will conclude our work in Section 5.

2 Existing CSLM Converting Methods

Traditional Backoff N-gram LMs (BNLMs) have been widely used in many NLP tasks (Zhang and Zhao, 2013; Jia and Zhao, 2014; Zhao et al., 2013; Zhang et al., 2012; Xu and Zhao, 2012; Wang et al., 2013b; Jia and Zhao, 2013; Wang et al., 2014). Recently, CSLMs become popular because they can obtain more accurate probability estimation.

2.1 Continues Space Language Model

A CSLM implemented in a multi-layer neural network contains four layers: the input layer projects (first layer) all words in the context h_i onto the projection layer (second layer); the hidden layer (third layer) and the output layer (fourth layer) achieve the non-liner probability estimation and calculate the LM probability $P(w_i|h_i)$ for the given context (Schwenk, 2007).

CSLM is able to calculate the probabilities of all words in the vocabulary of the corpus given the context. However, due to too high computational complexity, CSLM is mainly used to calculate the probabilities of a subset of the whole vocabulary (Schwenk, 2007). This subset is called a *short-list*, which consists of the most frequent words in the vocabulary. CSLM also calculates the sum of the probabilities of all words not included in the short-list by assigning a neuron with the help of BNLM. The probabilities of other words not in the short-list are obtained from an BNLM (Schwenk, 2007; Schwenk, 2010; Wang et al., 2013a).

Let w_i and h_i be the current word and history, respectively. CSLM with a BNLM calculates the probability $P(w_i|h_i)$ of w_i given h_i, as follows:

$$P(w_i|h_i) = \begin{cases} \frac{P_c(w_i|h_i)}{\sum_{w \in V_0} P_c(w|h_i)} P_s(h_i) & \text{if } w_i \in V_0 \\ P_b(w_i|h_i) & \text{otherwise} \end{cases} \quad (1)$$

where V_0 is the short-list, $P_c(\cdot)$ is the probability calculated by CSLM, $\sum_{w \in V_0} P_c(w|h_i)$ is the summary of probabilities of the neuron for all the words in the short-list, $P_b(\cdot)$ is the probability calculated by the BNLM, and

$$P_s(h_i) = \sum_{v \in V_0} P_b(v|h_i). \quad (2)$$

We may regard that CSLM redistributes the probability mass of all words in the short-list, which is calculated by using the n-gram LM.

2.2 Existing Converting Methods

As baseline systems, our approach proposed in (Wang et al., 2013a) only re-writes the probabilities from CSLM into the BNLM, so it can only conduct a convert LM with the same size as the original one. The main difference between our proposed method in this paper and our previous approach is that n-grams outside the corpus are generated firstly and the probabilities using CSLM are calculated by using the same method as our previous approach. That is, the proposed new method is the same as our previous one when no grown n-grams are generated.

The method developed by Arsoy and colleagues (Arsoy et al., 2013; Arsoy et al., 2014) adds all the words in the short-list after the tail word of the i-grams to construct the (i+1)-grams. For example, if the i-gram is "*I want*", then the (i+1)-grams will be "*I want* *", where "*" stands for any word in the short list. Then the probabilities of the (i+1)-grams are calculated using (i+1)-CSLM. So a very large intermediate (i+1)-grams will have to be grown[1], and then be pruned into smaller suitable size using an entropy-based LM pruning method modified from (Stolcke, 1998). The (i+2)-grams are grown using (i+1)-grams, recursively.

[1]In practice, the probabilities of all the target/tail words in the short list for the history i-grams can be calculated by the neurons in the output layer at the same time, which will save some time. According to our experiments, the time cost for Arsoy's growing method is around 4 times more than our proposed method, if the LMs which are 10 times larger than the original one are grown with other settings all the same.

3 Bilingual LM Growing

The translation output of a phrase-based SMT system can be regarded as a concatenation of phrases in the phrase table (except unknown words). This leads to the following procedure:

Step 1. All the n-grams included in the phrase table should be maintained at first.

Step 2. The connecting phrases are defined in the following way.

The w_a^b is a target language phrase starting from the a-th word ending with the b-th word, and $\beta w_a^b \gamma$ is a phrase including w_a^b as a part of it, where β and γ represent any word sequence or none. An i-gram phrase $w_1^k w_{k+1}^i$ $(1 \leq k \leq i-1)$ is a connecting phrase[2], if :

(1) w_1^k is the right (rear) part of one phrase βw_1^k in the phrase table, or

(2) w_{k+1}^i is the left (front) part of one phrase $w_{k+1}^i \gamma$ in the phrase table.

After the probabilities are calculated using C-SLM (Eqs.1 and 2), we combine the n-grams in the phrase table from Step 1 and the connecting phrases from Step 2.

3.1 Ranking the Connecting Phrases

Since the size of connecting phrases is too huge (usually more than one Terabyte), it is necessary to decide the usefulness of connecting phrases for SMT. The more useful connecting phrases can be selected, by ranking the appearing probabilities of the connecting phrases in SMT decoding.

Each line of a phrase table can be simplified (without considering other unrelated scores in the phrase table) as

$$f \;|||\; e \;|||\; P(e|f), \tag{3}$$

where the $P(e|f)$ means the translation probability from $f(source\ phrase)$ to $e(target\ phrase)$, which can be calculated using bilingual parallel training data. In decoding, the probability of a target phrase e appearing in SMT should be

$$P_t(e) = \sum_f P_s(f) \times P(e|f), \tag{4}$$

where the $P_s(f)$ means the appearing probability of a source phrase, which can be calculated using source language part in the bilingual training data.

Using $P_t(e)$[3], we can select the connecting phrases e with high appearing probabilities as the n-grams to be added to the original n-grams. These n-grams are called '*grown n-grams*'. Namely, we build all the connecting phrases at first, and then we use the appearing probabilities of the connecting phrases to decide which connecting phrases should be selected. For an i-gram connecting phrase $w_1^k w_{k+1}^i$, where w_1^k is part of βw_1^k and w_{k+1}^i is part of $w_{k+1}^i \gamma$ (the βw_1^k and $w_{k+1}^i \gamma$ are from the phrase table), the probability of the connecting phrases can be roughly estimated as

$$P_{\text{con}}(w_1^k w_{k+1}^i) = \sum_{k=1}^{i-1} \Big(\sum_{\beta} P_t(\beta w_1^k) \times \sum_{\gamma} P_t(w_{k+1}^i \gamma) \Big). \tag{5}$$

A threshold for $P_{\text{con}}(w_1^k w_{k+1}^i)$ is set, and only the connecting phrases whose appearing probabilities are higher than the threshold will be selected as the grown n-grams.

3.2 Calculating the Probabilities of Grown N-grams Using CSLM

To our bilingual LM growing method, a 5-gram LM and n-gram (n=2,3,4,5) CSLMs are built by using the target language of the parallel corpus, and the phrase table is learned from the parallel corpus.

The probabilities of unigram in the original n-gram LM will be maintained as they are. The n-grams from the bilingual phrase table will be grown by using the '*connecting phrases*' method. As the whole connecting phrases are too huge, we use the ranking method to select the more useful connecting phrases. The distribution of different n-grams (n=2,3,4,5) of the grown LMs are set as the same as the original LM.

The probabilities of the grown n-grams (n=2,3,4,5) are calculated using the 2,3,4,5-CSLM, respectively. If the tail (target) words of the grown n-grams are not in the short-list of C-SLM, the $P_b(\cdot)$ in Eq. 1 will be applied to calculate their probabilities.

[2] We are aware that connecting phrases can be applied to not only two phrases, but also three or more. However the appearing probabilities (which will be discussed in Eq. 5 of next subsection) of connecting phrases are approximately estimated. To estimate and compare probabilities of longer phrases in different lengths will lead to serious bias, and the experiments also showed using more than two connecting phrases did not perform well (not shown for limited space), so only two connecting phrases are applied in this paper.

[3] This $P_t(e)$ hence provides more bilingual information, in comparison with using monolingual target LMs only.

We combine the n-grams (n=1,2,3,4,5) together and re-normalize the probabilities and backoff weights of the grown LM. Finally the original BNLM and the grown LM are interpolated. The entire process is illustrated in Figure 1.

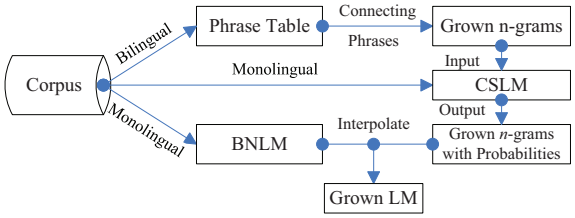

Figure 1: NN based bilingual LM growing.

4 Experiments and Results

4.1 Experiment Setting up

The same setting up of the NTCIR-9 Chinese to English translation baseline system (Goto et al., 2011) was followed, only with various LMs to compare them. The Moses phrase-based SMT system was applied (Koehn et al., 2007), together with GIZA++ (Och and Ney, 2003) for alignment and MERT (Och, 2003) for tuning on the development data. Fourteen standard SMT features were used: five translation model scores, one word penalty score, seven distortion scores, and one LM score. The translation performance was measured by the case-insensitive BLEU on the tokenized test data.

We used the patent data for the Chinese to English patent translation subtask from the NTCIR-9 patent translation task (Goto et al., 2011). The parallel training, development, and test data sets consist of 1 million (M), 2,000, and 2,000 sentences, respectively.

Using SRILM (Stolcke, 2002; Stolcke et al., 2011), we trained a 5-gram LM with the interpolated Kneser-Ney smoothing method using the 1M English training sentences containing 42M words without cutoff. The 2,3,4,5-CSLMs were trained on the same 1M training sentences using CSLM toolkit (Schwenk, 2007; Schwenk, 2010). The settings for CSLMs were: input layer of the same dimension as vocabulary size (456K), projection layer of dimension 256 for each word, hidden layer of dimension 384 and output layer (short-list) of dimension 8192, which were recommended in the CSLM toolkit and (Wang et al., 2013a)[4].

[4]Arsoy used around 55 M words as the corpus, including 84K words as vocabulary, and 20K words as short-list. In this paper, we used the same setting as our previous work, which covers 92.89% of the frequency of words in the training corpus, for all the baselines and our method for fair comparison.

4.2 Results

The experiment results were divided into four groups: the original BNLMs (BN), the CSLM Re-ranking (RE), our previous converting (WA), the Arsoy's growing, and our growing methods. For our bilingual LM growing method, 5 bilingual grown LMs (BI-1 to 5) were conducted in increasing sizes. For the method of Arsoy, 5 grown LMs (AR-1 to 5) with similar size of BI-1 to 5 were also conducted, respectively.

For the CSLM re-ranking, we used CSLM to re-rank the 100-best lists of SMT. Our previous converted LM, Arsoy's grown LMs and bilingual grown LMs were interpolated with the original BNLMs, using default setting of SRILM[5]. To reduce the randomness of MERT, we used two methods for tuning the weights of different SMT features, and two BLEU scores are corresponding to these two methods. The **BLEU-s** indicated that the **same** weights of the BNLM (BN) features were used for all the SMT systems. The **BLEU-i** indicated that the MERT was run **independently** by three times and the average BLEU scores were taken.

We also performed the paired bootstrap resampling test (Koehn, 2004)[6]. Two thousands samples were sampled for each significance test. The marks at the right of the BLEU score indicated whether the LMs were significantly better/worse than the Arsoy's grown LMs with the same IDs for SMT ("++/−−": significantly better/worse at $\alpha = 0.01$, "+/−": $\alpha = 0.05$, no mark: not significantly better/worse at $\alpha = 0.05$).

From the results shown in Table 1, we can get the following observations:

(1) Nearly all the bilingual grown LMs outperformed both BNLM and our previous converted LM on PPL and BLEU. As the size of grown LMs is increased, the PPL always decreased and the BLEU scores trended to increase. These indicated that our proposed method can give better probability estimation for LM and better performance for SMT.

(2) In comparison with the grown LMs in Ar-

[5]In our previous work, we used the development data to tune the weights of interpolation. In this paper, we used the default 0.5 as the interpolation weights for fair comparison.

[6]We used the code available at `http://www.ark.cs.cmu.edu/MT`

Table 1: Performance of the Grown LMs

LMs	n-grams	PPL	BLEU-s	BLEU-i	ALH
BN	73.9M	108.8	32.19	32.19	3.03
RE	N/A	**97.5**	32.34	32.42	N/A
WA	73.9M	104.4	32.60	32.62	3.03
AR-1	217.6M	103.3	32.55	32.75	3.14
AR-2	323.8M	103.1	32.61	32.64	3.18
AR-3	458.5M	103.0	32.39	32.71	3.20
AR-4	565.6M	102.8	32.67	32.51	3.21
AR-5	712.2M	102.5	32.49	32.60	3.22
BI-1	223.5M	101.9	32.81+	33.02+	3.20
BI-2	343.6M	101.0	32.92+	33.11++	3.24
BI-3	464.5M	100.6	33.08++	**33.25**++	3.26
BI-4	571.0M	100.3	**33.15**++	33.12++	3.28
BI-5	705.5M	100.1	33.11++	33.24++	3.31

soy's method, our grown LMs obtained better PPL and significantly better BLEU with the similar size. Furthermore, the improvement of PPL and BLEU of the existing methods became saturated much more quickly than ours did, as the LMs grew.

(3) The last column was the Average Length of the n-grams Hit (ALH) in SMT decoding for different LMs using the following function

$$ALH = \sum_{i=1}^{5} P_{i-gram} \times i, \tag{6}$$

where the P_{i-gram} means the ratio of the i-grams hit in SMT decoding. There were also positive correlations between ALH, PPL and BLEUs. The ALH of bilingual grown LM was longer than that of the Arsoy's grown LM of the similar size. In another word, less back-off was used for our proposed grown LMs in SMT decoding.

4.3 Experiments on TED Corpus

The TED corpus is in special domain as discussed in the introduction, where large extra monolingual corpora are hard to find. In this subsection, we conducted the SMT experiments on TED corpora using our proposed LM growing method, to evaluate whether our method was adaptable to some special domains.

We mainly followed the baselines of the IWSLT 2014 evaluation campaign[7], only with a few modifications such as the LM toolkits and n-gram order for constructing LMs. The Chinese (CN) to English (EN) language pair was chosen, using dev2010 as development data and test2010 as evaluation data. The same LM growing method was applied on TED corpora as on NTCIR corpora. The results were shown in Table 2.

[7] https://wit3.fbk.eu/

Table 2: CN-EN TED Experiments

LMs	n-grams	PPL	BLEU-s
BN	7.8M	87.1	12.41
WA	7.8M	85.3	12.73
BI-1	23.1M	79.2	12.92
BI-2	49.7M	78.3	13.16
BI-3	73.4M	**77.6**	**13.24**

Table 2 indicated that our proposed LM growing method improved both PPL and BLEU in comparison with both BNLM and our previous CSLM converting method, so it was suitable for domain adaptation, which is one of focuses of the current SMT research.

5 Conclusion

In this paper, we have proposed a neural network based bilingual LM growing method by using the bilingual parallel corpus only for SMT. The results show that our proposed method can improve both LM and SMT performance, and outperforms the existing LM growing methods significantly without extra corpus. The connecting phrase-based method can also be applied to LM adaptation.

Acknowledgments

We appreciate the helpful discussion with Dr. Isao Goto and Zhongye Jia, and three anonymous reviewers for valuable comments and suggestions on our paper. Rui Wang, Hai Zhao and Bao-Liang Lu were partially supported by the National Natural Science Foundation of China (No. 60903119, No. 61170114, and No. 61272248), the National Basic Research Program of China (No. 2013CB329401), the Science and Technology Commission of Shanghai Municipality (No. 13511500200), the European Union Seventh Framework Program (No. 247619), the Cai Yuanpei Program (CSC fund 201304490199 and 201304490171), and the art and science interdiscipline funds of Shanghai Jiao Tong University (A study on mobilization mechanism and alerting threshold setting for online community, and media image and psychology evaluation: a computational intelligence approach). The corresponding author of this paper, according to the meaning given to this role by Shanghai Jiao Tong University, is Hai Zhao.

References

Ebru Arsoy, Stanley F. Chen, Bhuvana Ramabhadran, and Abhinav Sethy. 2013. Converting neural network language models into back-off language models for efficient decoding in automatic speech recognition. In *Proceedings of ICASSP-2013*, Vancouver, Canada, May. IEEE.

Ebru Arsoy, Stanley F. Chen, Bhuvana Ramabhadran, and Abhinav Sethy. 2014. Converting neural network language models into back-off language models for efficient decoding in automatic speech recognition. *IEEE/ACM Transactions on Audio, Speech, and Language*, 22(1):184–192.

Michael Auli, Michel Galley, Chris Quirk, and Geoffrey Zweig. 2013. Joint language and translation modeling with recurrent neural networks. In *Processings of EMNLP-2013*, pages 1044–1054, Seattle, Washington, USA, October. Association for Computational Linguistics.

Jerome R Bellegarda. 2004. Statistical language model adaptation: review and perspectives. *Speech Communication*, 42(1):93–108. Adaptation Methods for Speech Recognition.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. 2003. A neural probabilistic language model. *Journal of Machine Learning Research (JMLR)*, 3:1137–1155, March.

Mauro Cettolo, Christian Girardi, and Marcello Federico. 2012. Wit^3: Web inventory of transcribed and translated talks. In *Proceedings of EAMT-2012*, pages 261–268, Trento, Italy, May.

Philip Clarkson and A.J. Robinson. 1997. Language model adaptation using mixtures and an exponentially decaying cache. In *Proceedings of ICASSP-1997*, volume 2, pages 799–802 vol.2, Munich,Germany.

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 ACL-2014*, pages 1370–1380, Baltimore, Maryland, June. Association for Computational Linguistics.

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*, pages 559–578, Tokyo, Japan, December.

Rukmini Iyer, Mari Ostendorf, and Herbert Gish. 1997. Using out-of-domain data to improve in-domain language models. *Signal Processing Letters, IEEE*, 4(8):221–223.

Zhongye Jia and Hai Zhao. 2013. Kyss 1.0: a framework for automatic evaluation of chinese input method engines. In *Proceedings of IJCNLP-2013*, pages 1195–1201, Nagoya, Japan, October. Asian Federation of Natural Language Processing.

Zhongye Jia and Hai Zhao. 2014. A joint graph model for pinyin-to-chinese conversion with typo correction. In *Proceedings of ACL-2014*, pages 1512–1523, Baltimore, Maryland, June. Association for Computational Linguistics.

Philipp Koehn and Josh Schroeder. 2007. Experiments in domain adaptation for statistical machine translation. In *Proceedings of ACL-2007 Workshop on Statistical Machine Translation*, pages 224–227, Stroudsburg, PA, USA. 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 ACL-2007*, pages 177–180, Prague, Czech Republic, June. Association for Computational Linguistics.

Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of EMNLP-2004*, pages 388–395, Barcelona, Spain, July. Association for Computational Linguistics.

Hai-Son Le, Ilya Oparin, Alexandre Allauzen, J Gauvain, and François Yvon. 2011. Structured output layer neural network language model. In *Proceedings of ICASSP-2011*, pages 5524–5527, Prague, Czech Republic, May. IEEE.

Shujie Liu, Nan Yang, Mu Li, and Ming Zhou. 2014. A recursive recurrent neural network for statistical machine translation. In *Proceedings of ACL-2014*, pages 1491–1500, Baltimore, Maryland, June. Association for Computational Linguistics.

Tomas Mikolov, Martin Karafiát, Lukas Burget, Jan Cernockỳ, and Sanjeev Khudanpur. 2010. Recurrent neural network based language model. In *Proceedings of INTERSPEECH-2010*, pages 1045–1048.

Jan Niehues and Alex Waibel. 2012. Continuous space language models using restricted boltzmann machines. In *Proceedings of IWSLT-2012*, pages 311–318, Hong Kong.

Thomas Niesler and Phil Woodland. 1996. A variable-length category-based n-gram language model. In *Proceedings of ICASSP-1996*, volume 1, pages 164–167 vol. 1.

Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51, March.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *Proceedings of ACL-2003*, pages 160–167, Sapporo, Japan, July. Association for Computational Linguistics.

Eric Sven Ristad and Robert G. Thomas. 1995. New techniques for context modeling. In *Proceedings of ACL-1995*, pages 220–227, Cambridge, Massachusetts. Association for Computational Linguistics.

Holger Schwenk, Daniel Dchelotte, and Jean-Luc Gauvain. 2006. Continuous space language models for statistical machine translation. In *Proceedings of COLING ACL-2006*, pages 723–730, Sydney, Australia, July. Association for Computational Linguistics.

Holger Schwenk, Anthony Rousseau, and Mohammed Attik. 2012. Large, pruned or continuous space language models on a gpu for statistical machine translation. In *Proceedings of the NAACL-HLT 2012 Workshop: Will We Ever Really Replace the N-gram Model? On the Future of Language Modeling for HLT*, WLM '12, pages 11–19, Montreal, Canada, June. Association for Computational Linguistics.

Holger Schwenk. 2007. Continuous space language models. *Computer Speech and Language*, 21(3):492–518.

Holger Schwenk. 2010. Continuous-space language models for statistical machine translation. *The Prague Bulletin of Mathematical Linguistics*, pages 137–146.

Vesa Siivola, Teemu Hirsimki, and Sami Virpioja. 2007. On growing and pruning kneser-ney smoothed n-gram models. *IEEE Transactions on Audio, Speech, and Language*, 15(5):1617–1624.

Manhung Siu and Mari Ostendorf. 2000. Variable n-grams and extensions for conversational speech language modeling. *IEEE Transactions on Speech and Audio*, 8(1):63–75.

Le Hai Son, Alexandre Allauzen, Guillaume Wisniewski, and François Yvon. 2010. Training continuous space language models: some practical issues. In *Proceedings of EMNLP-2010*, pages 778–788, Cambridge, Massachusetts, October. Association for Computational Linguistics.

Le Hai Son, Alexandre Allauzen, and François Yvon. 2012. Continuous space translation models with neural networks. In *Proceedings of NAACL HLT-2012*, pages 39–48, Montreal, Canada, June. Association for Computational Linguistics.

Andreas Stolcke, Jing Zheng, Wen Wang, and Victor Abrash. 2011. SRILM at sixteen: Update and outlook. In *Proceedings of INTERSPEECH 2011*, Waikoloa, HI, USA, December.

Andreas Stolcke. 1998. Entropy-based pruning of backoff language models. In *Proceedings of DARPA Broadcast News Transcription and Understanding Workshop*, pages 270–274, Lansdowne, VA, USA.

Andreas Stolcke. 2002. Srilm-an extensible language modeling toolkit. In *Proceedings of INTERSPEECH-2002*, pages 257–286, Seattle, USA, November.

Ashish Vaswani, Yinggong Zhao, Victoria Fossum, and David Chiang. 2013. Decoding with large-scale neural language models improves translation. In *Proceedings of EMNLP-2013*, pages 1387–1392, Seattle, Washington, USA, October. Association for Computational Linguistics.

Rui Wang, Masao Utiyama, Isao Goto, Eiichro Sumita, Hai Zhao, and Bao-Liang Lu. 2013a. Converting continuous-space language models into n-gram language models for statistical machine translation. In *Proceedings of EMNLP-2013*, pages 845–850, Seattle, Washington, USA, October. Association for Computational Linguistics.

Xiaolin Wang, Hai Zhao, and Bao-Liang Lu. 2013b. Labeled alignment for recognizing textual entailment. In *Proceedings of IJCNLP-2013*, pages 605–613, Nagoya, Japan, October. Asian Federation of Natural Language Processing.

Xiao-Lin Wang, Yang-Yang Chen, Hai Zhao, and Bao-Liang Lu. 2014. Parallelized extreme learning machine ensemble based on minmax modular network. *Neurocomputing*, 128(0):31 – 41.

Qiongkai Xu and Hai Zhao. 2012. Using deep linguistic features for finding deceptive opinion spam. In *Proceedings of COLING-2012*, pages 1341–1350, Mumbai, India, December. The COLING 2012 Organizing Committee.

Jingyi Zhang and Hai Zhao. 2013. Improving function word alignment with frequency and syntactic information. In *Proceedings of IJCAI-2013*, pages 2211–2217. AAAI Press.

Xiaotian Zhang, Hai Zhao, and Cong Hui. 2012. A machine learning approach to convert CCGbank to Penn treebank. In *Proceedings of COLING-2012*, pages 535–542, Mumbai, India, December. The COLING 2012 Organizing Committee.

Hai Zhao, Masao Utiyama, Eiichiro Sumita, and Bao-Liang Lu. 2013. An empirical study on word segmentation for chinese machine translation. In Alexander Gelbukh, editor, *Computational Linguistics and Intelligent Text Processing*, volume 7817 of *Lecture Notes in Computer Science*, pages 248–263. Springer Berlin Heidelberg.