Interspeech 2018
2-6 September 2018, Hyderabad

Investigation on Estimation of Sentence Probability By Combining Forward, Backward and Bi-directional LSTM-RNNs

Kazuki Irie, Zhihong Lei, Liuhui Deng, Ralf Schlüter, Hermann Ney

Human Language Technology and Pattern Recognition Group,
Computer Science Department, RWTH Aachen University, D-52056 Aachen, Germany
{irie, schlueter, ney}@cs.rwth-aachen.de, {zhihong.lei, liuhui.deng}@rwth-aachen.de

Abstract

A combination of forward and backward long short-term memory (LSTM) recurrent neural network (RNN) language models is a popular model combination approach to improve the estimation of the sequence probability in the second pass N-best list rescoring in automatic speech recognition (ASR). In this work, we further push such an idea by proposing a combination of three models: a forward LSTM language model, a backward LSTM language model and a bi-directional LSTM based gap completion model. We derive such a combination method from a forward backward decomposition of the sequence probability. We carry out experiments on the Switchboard speech recognition task. While we empirically find that such a combination gives slight improvements in perplexity over the combination of forward and backward models, we finally show that a combination of the same number of forward models gives the best perplexity and word error rate (WER) overall.

Index Terms: language modeling, speech recognition, bi-directional LSTM, forward backward

1. Introduction

The language model estimates the probability $p(w_0^N)$ of a sequence of words w_0^N (where w_0 and w_N are sentence boundaries). This probability can be factorized using the chain rule of probability, as

$$p(w_0^N) = \prod_{k=1}^{N} p(w_k|w_0^{k-1}) \tag{1}$$

The task of language modeling then becomes the estimation of these conditional probabilities $p(w_k|w_0^{k-1})$.

This factorization tells that the computation of each conditional probability, which estimates the probability of the next token given its predecessors, should not make use of any future contexts, in order to come back to a properly normalized sequence probability. Therefore, the (uni-directional) recurrent neural network [1], which perfectly fits into this problem, is the most standard approach for neural language modeling.

In practice, the combination of multiple language models is usually used to achieve state-of-the-art results in speech recognition [2, 3]. In order to obtain some model diversity, LSTM-RNN language models [4, 5] are often trained by different random initializations of the same type of the model, by using different model architectures [3, 6] or different input features [2, 7]. Yet another common approach is to train LSTM-RNN language models in forward and backward directions and average the scores of both models on the sequence level [2, 7]. In this work, we further push such an idea, based on an alternative decomposition of the sequence probability from Eq. (1), in the perspective of model combination. We introduce three components: a forward language model, a (prefix-conditioned) backward language model and a gap completion model. We model them respectively by two LSTMs and a bi-directional LSTM [8, 9]. We carry out experiments on the Switchboard speech recognition task and show that such a combination brings slight improvements in terms of perplexity over the combination of forward and backward models. However, the use of these 3 models can only be motivated if they effectively offer a better diversity for model combination. We find that the combination of same number of forward models with different initialization gives the best performance in terms of both perplexity and WER, which empirically shows that the use uni-directional LSTM-RNN is sufficient for language modeling.

2. Gap completion model and computation of sequence probability

We denote a sentence: $w_0^N = w_0, w_1, w_2, ..., w_{N-1}, w_N$ where w_0 and w_N are the sentence boundaries.

We consider a factorization of the sentence probability $p(w_0^N)$ different than Eq. (1) in the perspective of model combination. For that, we consider an arbitrary position index $k \in \{2, ..., N-2\}$ and we factorize $p(w_0^N)$ in the forward direction up to the position $k-1$ and in backward direction from the position N to $k+1$ which gives:

$$\begin{aligned}
p(w_0^N) &= p(w_0^{k-1}) \cdot p(w_k, w_{k+1}^N|w_0^{k-1}) && (2)\\
&= p(w_0^{k-1}) \cdot p(w_k|w_0^{k-1}, w_N^{k+1}) \cdot p(w_N^{k+1}|w_0^{k-1}) && (3)\\
&\overset{\text{model}}{=} \underbrace{p(w_0^{k-1})}_{\text{Fwd LM}} \cdot \underbrace{p(w_k|w_0^{k-1}, w_N^{k+1})}_{\text{Completion Model}} \cdot \underbrace{p(w_N^{k+1})}_{\text{Bwd LM}} && (4)\\
&= p_{\text{fwd}}(w_0^{k-1}) \cdot p_{\text{cmp}}(w_k|w_0^{k-1}, w_N^{k+1}) \cdot p_{\text{bwd}}(w_N^{k+1})\\
&= f_k \text{ for any k} \in \{2, ..., N-2\} && (5)
\end{aligned}$$

In Eq. (3), we obtain the forward language model term $p(w_0^{k-1})$ which we model with a LSTM. We define the second term $p(w_k|w_0^{k-1}, w_N^{k+1})$ as a completion model which completes the gap of one word between the forward context w_0^{k-1} and backward context w_N^{k+1}. We parametrize that model with a bi-directional LSTM [8, 9]. The last term which is left is a backward term conditioned on the prefix context $p(w_N^{k+1}|w_0^{k-1})$. While such a term can also be directly modeled using one LSTM for encoding the prefix context and another LSTM for backward language modeling, in this work, we investigate this combination using the standard backward LM which does not make use of the prefix information.

10.21437/Interspeech.2018-1766

We extend this notation f_k to the boundary cases where we use the forward or backward model only:

$$f_1 = p_{\text{bwd}}(w_N^0)$$
$$f_{N-1} = p_{\text{fwd}}(w_0^N)$$

Since this quantity f_k can be computed for any position $k \in \{1, ..., N-1\}$, we therefore end up with (N-1) ways to combine the three models to estimate the same probability $p(w_0^N)$, which we can average:

$$p(w_0^N) = \frac{1}{N-1} \sum_{k=1}^{N-1} f_k$$

We note that the standard combination of forward and backward models can be written using our notation as:

$$p(w_0^N) = \frac{1}{2}(f_1 + f_{N-1})$$

We also note that the notation in the previous equations are only valid when $N > 2$. If the sentence only contains a single (non-boundary) word (i.e. $N = 2$), we also only combine the forward and backward models.

3. Related work and motivation

Several previous work [10, 11, 12] have investigated the possibility to define a "bi-directional language model", by directly replacing the uni-directional LSTM for the conditional probability $p(w_k|w_0^{k-1})$ in Eq. (1) by a bi-directional LSTM getting both the past and future contexts $p(w_k|w_0^{k-1}, w_N^{k+1})$ in Eq. (3). Such an approach is not straightforward, since the word probability conditioned on both past and future information alone has no direct relation to the computation of sentence probability (as can be seen in Eq. (3)). Chen et al. [13, 14] has introduced renormalization and report improvements in terms of WER, when the bi-directional model is combined with the standard uni-directional language model. However, the renormalization term is only computed approximately which still does not allow to compute the perplexity of the combined model.

The objective of this work is not to build a "bi-directional language model". Instead, we define $p(w_k|w_0^{k-1}, w_N^{k+1})$ simply as a *gap completion model* which can be used in the computation of sentence probability only when it is completed by forward and backward LSTM language model contribution up to position k. As shown in Sec. 2., we naturally introduce that quantity in the computation of sentence probability without requiring any renormalization. Also, such an approach is a natural extension of the combination of forward and backward language models, which is a common model combination with LSTM LM [2].

However, like previous work, the introduction of the bi-directional term in the Eq. (3) also does not have any theoretical advantage over the standard chain rule Eq. (1). This is for example in contrast to the noisy channel factorization, where the introduction of the prior (language model) term is motivated by the fact that it can be trained without labeled data. Like the combination of forward and backward models, our approach is only motivated by model diversity in the perspective of model combination. Therefore, the objective of this work is also to conclude on the empirical benefit of such a type of model combination which mixes language models trained on forward and backward directions. For that it is crucial to empirically check the advantage over the combination of forward models only.

Our approach is also related to the whole sentence modeling [15] as our combination is based on diverse estimations of the sentence probability.

4. Text based experiments

4.1. Model descriptions

We carry out text-based experiments on two tasks: 27M-word Switchboard Telephone speech conversation task and 50M-word Quaero English broadcast news task. The vocabulary sizes are respectively 30K and 130K and for both language models and completion models, we factorize the softmax output using word classes of size 200 and 1000 respectively. The statistics of the datasets are summarized in Table 1.

For both forward and backward neural language models, for Switchboard experiments, we use a input projection layer of 500 nodes and 1 LSTM layer with 500 nodes, followed by one feedforward layer with 500 nodes and gated linear unit activation function [16]. For Quaero experiments, we used the projection and two LSTM layers with dimension 600 each.

For Switchboard, the completion models have a input projection layer of 500 dimension and 1 LSTM layer of dimension 500 for each direction. The outputs of forward and backward LSTM are concatenated and fed to one feedforward layer with the gated linear unit activation.The same model architecture is used for Quaero experiments but with dimension 600. We trained all models using the stochastic gradient descent with batch size of 8 for language models and 16 for completion models on CPUs using `rwthlm` [17].

Table 1: *Number of running words, OOV rates and average sentence lengths in words of all data sets used. The vocabulary size is 30K for Switchboard and 130K for Quaero tasks.*

		Run. Words	OOV[%]	Avg. len.
Switchboard Train		26.7M	1.6	11
Switchboard CV		133K	0	13
Hub5_00	Total	45K	1.1	10
	CH	23K	1.6	9
	SWB	22K	0.7	12
Hub5e_01		65K	1.0	11
Quaero	Train	50M	1.0	16
	Dev	40K	0.4	28
	Eval	36K	0.5	30

4.2. Perplexity results

The perplexity results for Switchboard experiments are shown on Table 2. All perplexities reported in this work are computed without making use of context beyond sentence boundaries for both language models and completion models. The perplexities of the standalone models used for combination can be found on the top of Table 2. First we first confirm that the forward and backward LSTM models have the similar perplexity and that the combination of forward and backward models on sentence probability level gives improvements over the standalone models. However, we find that the combination of two forward models gives slightly better perplexity than the combination of forward and backward model. All forward models have the same

architecture and trained with the same hyper-parameters except the random seed for initialization and data shuffling.

Similar observation can be done for the forward, backward, completion combination: the combination gives better perplexity than the forward and backward combination but the combination of three forward models gives the best perplexities.

We confirm the same observation for the Quaero English task. The perplexities are shown on Table 3. The models used for combination are summarized on the top of Table 3. We again find that the combination of forward, backward and completion models give slightly improvements over the forward and backward combination. However, both approaches do not give better performance than the forward-only combination in our experiments. There is also no theoretical reason for the combination of two forward models to be better than the combination of forward and backward models neither. However, in our experiments, we found that the combination of forward and backward models does not have any empirical advantage over the forward only combination.

Table 2: *Perplexities on the **Switchboard** corpus. No context beyond sentence boundaries are used.*

LSTM LM Type	PPL	
	Hub5_00	Hub5e_01
Forward (1)	59.8	50.7
Forward (2)	60.2	51.5
Forward (3)	59.3	50.9
Backward	60.1	51.0
Forward + Backward	56.8	48.3
Forward (1) + (2)	56.3	48.0
Forward + Backward + Completion	55.6	48.1
Forward (1) + (2) + (3)	**55.1**	**47.0**

Table 3: *Perplexities on the **Quaero** corpus. No context beyond sentence boundaries are used.*

LSTM LM Type	PPL	
	Dev	Eval
Forward (a)	108.4	107.4
Forward (b)	110.1	109.0
Forward (c)	110.4	109.4
Backward	108.3	107.6
Forward + Backward	103.0	102.4
Forward (a) + (b)	100.1	99.3
Forward + Backward + Completion	101.3	100.6
Forward (a) + (b) + (c)	**97.5**	**96.8**

4.3. Performance of completion models

While we can only report perplexities when the completion model is combined with forward and backward models using Eq. (3), we can separately report the pseudo-perplexity [13] of the completion model based on $p(w_k|w_0^{k-1}, w_N^{k+1})$. We found the pseudo-perplexities of 16.6 and 13.4 respectively for Hub00 and Hub5e_01 sets of Switchboard, 29.7 and 29.8 for development and evaluation sets of Quaero. They are effectively much smaller than the perplexity of language models since it's prediction is conditioned on both forward and backward context.

5. ASR experiments

5.1. Baseline Setups

We carry out ASR experiments on the Switchboard dataset. Our acoustic model is based on bi-directional LSTM neural networks with 6 hidden layers. There are 500 LSTM units in each hidden layer and the output layer models generalized triphone state posterior probabilities for 9000 CART labels. The input is a 40 dimensional Gammatone features [18]. We apply dropout with a rate of 0.1 at the input of each hidden layer and train the model on the 300-hour training dataset of Switchboard with nadam optimizer [19] using the RETURNN toolkit [20].

The baseline Kneser-Ney smoothed [21] 4-gram count language model (KN4) is trained on the 27M-word Switchboard training data mentioned in Sec. 4.1. using SRILM toolkit [22]. We use this model for decoding and apply the neural language model in the second pass rescoring.

The application of forward and backward LSTM model combination, as well as the completion model for lattice rescoring [23] is not straightforward. In order to keep the comparison simple, we proceed 1000-best list rescoring to apply all neural language models.

5.2. Results

Table 4 shows the perplexity and WER results. The perplexity are reported after interpolation with the baseline 4-gram count model on the sentence-level. We observe that after interpolation with the count model, the perplexity performance order observed in the Table 2 are not always preserved across different subsets and the close perplexities make their correlation to WER results noisy. However, we find that the simple combination of the forward models perform the best overall.

Table 4: *ASR results on the **Switchboard** corpus. 300-hour training dataset is used. PPLs are reported after **linear interpolation** of neural LMs with the KN4 at the sentence level.*

LM Combination	Hub5_00				Hub5e_01	
	CH		SWB			
	PPL	WER	PPL	WER	PPL	WER
KN4	80.5	19.1	68.8	9.8	65.3	14.7
+ Forward	61.2	17.3	49.5	8.3	47.3	12.9
+ Backward	61.7	**17.1**	49.7	**8.1**	47.6	**12.7**
+ Completion	59.7	**17.1**	50.6	8.4	47.7	**12.7**
+ Forward $\times$ 2	59.6	**17.1**	48.3	8.2	45.9	12.8
+ Forward $\times$ 3	**58.7**	**17.1**	**47.6**	**8.1**	**45.7**	**12.7**

6. Conclusion

We proposed a well-defined method of using bi-directionality in language modeling via combination, which allows to compute the normalized sentence probability and perplexity. We investigated the computation of sentence probability by using bi-directional LSTM based completion model which is completed by the contribution of forward and backward LSTM language models. While our approach was motivated by the extension of the popular forward and backward language model combination, we found the simple combination of forward language models with different training hyper-parameters to overall perform the best in our experiments.

7. Discussion

We have not fully investigated the potential of the forward, backward and completion combination. In fact, we applied a

strong model assumption from Eq. (3) and Eq. (4) by discarding the prefix-context in the backward model, which should directly affect the performance of the backward model. However, removing this simplification would make the backward model computationally inefficient, since for each position k needed for the factorization, we need different prefix context encoding for the backward model. In order to alleviate such a problem, in the future work, we would like to investigate the possibility for a joint model of forward and backward components which would allow to tie the LSTM for encoding forward context between the forward language model and the backward model.

8. Acknowledgements

We thank Albert Zeyer for sharing his set-ups for Switchboard experiments. We thank Tamer Alkhouli for sharing his implementation of the bi-directional LSTM. This work has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 694537, project "SEQCLAS"). The work reflects only the authors' views and the ERC Executive Agency is not responsible for any use that may be made of the information it contains. The GPU cluster used for the experiments was partially funded by Deutsche Forschungsgemeinschaft (DFG) Grant INST 222/1168-1.

9. References

[1] T. Mikolov, M. Karafiát, L. Burget, J. Cernocký, and S. Khudanpur, "Recurrent neural network based language model," in *Proc. Interspeech*, Makuhari, Japan, Sep. 2010, pp. 1045–1048.

[2] W. Xiong *et al.*, "Toward human parity in conversational speech recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 12, pp. 2410–2423, 2017.

[3] G. Saon *et al.*, "English conversational telephone speech recognition by humans and machines," in *Proc. Interspeech*, Stockholm, Sweden, Aug. 2017, pp. 132–136.

[4] M. Sundermeyer, R. Schlüter, and H. Ney, "LSTM neural networks for language modeling." in *Proc. Interspeech*, Portland, OR, USA, Sep. 2012, pp. 194–197.

[5] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[6] G. Kurata, B. Ramabhadran, G. Saon, and A. Sethy, "Language modeling with highway LSTM," in *Proc. IEEE Automatic Speech Recog. and Understanding Workshop (ASRU)*, Okinawa, Japan, Dec. 2017, pp. 244–251.

[7] W. Xiong, L. Wu, J. Droppo, X. Huang, and A. Stolcke, "The microsoft 2017 conversational speech recognition system," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, Calgary, Canada, Apr. 2018, pp. 5934–5938.

[8] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997.

[9] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional LSTM and other neural network architectures," *Neural Networks*, vol. 18, no. 5-6, pp. 602–610, 2005.

[10] E. Arisoy, A. Sethy, B. Ramabhadran, and S. F. Chen, "Bidirectional recurrent neural network language models for automatic speech recognition," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, Brisbane, Australia, Apr. 2015, pp. 5421–5425.

[11] A. Peris and F. Casacuberta, "A bidirectional recurrent neural language model for machine translation," *Procesamiento del Lenguaje Natural*, no. 55, 2015.

[12] T. He, Y. Zhang, J. Droppo, and K. Yu, "On training bi-directional neural network language model with noise contrastive estimation," in *Proc. Int. Symposium on Chinese Spoken Language Processing (ISCSLP)*, Tianjin, China, Oct. 2016, pp. 1–5.

[13] X. Chen, A. Ragni, X. Liu, and M. J. Gales, "Investigating bidirectional recurrent neural network language models for speech recognition," in *Proc. Interspeech*, Stockholm, Sweden, Aug. 2017, pp. 269–273.

[14] X. Chen, X. Liu, A. Ragni, Y. Wang, and M. J. Gales, "Future word contexts in neural network language models," in *Proc. IEEE Automatic Speech Recog. and Understanding Workshop (ASRU)*, Okinawa, Japan, Dec. 2017, pp. 97–103.

[15] Y. Huang, A. Sethy, K. Audhkhasi, and B. Ramabhadran, "Whole sentence neural language model," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, Calgary, Canada, Apr. 2018, pp. 6089–6093.

[16] Y. N. Dauphin, A. Fan, M. Auli, and D. Grangier, "Language modeling with gated convolutional networks," in *Proc. Int. Conf. on Machine Learning (ICML)*, Sydney, Australia, Aug. 2017, pp. 933–941.

[17] M. Sundermeyer, R. Schlüter, and H. Ney, "rwthlm the RWTH Aachen University neural network language modeling toolkit," in *Proc. Interspeech*, Singapore, Sep. 2014, pp. 2093–2097.

[18] R. Schlüter, I. Bezrukov, H. Wagner, and H. Ney, "Gammatone features and feature combination for large vocabulary speech recognition," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, Honolulu, HI, USA, Apr. 2007, pp. 649–652.

[19] T. Dozat, "Incorporating nesterov momentum into adam," in *Int. Conf. on Learning Representations (ICLR), Workshop track*, San Juan, Puerto Rico, May 2016.

[20] P. Doetsch *et al.*, "RETURNN: the RWTH extensible training framework for universal recurrent neural networks," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, New Orleans, LA, USA, 2017.

[21] R. Kneser and H. Ney, "Improved backing-off for m-gram language modeling," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, Detroit, MI, USA, May 1995, pp. 181–184.

[22] A. Stolcke, "SRILM-an extensible language modeling toolkit." in *Proc. Interspeech*, Denver, CO, USA, 2002, pp. 901–904.

[23] M. Sundermeyer, Z. Tüske, R. Schlüter, and H. Ney, "Lattice decoding and rescoring with long-span neural network language models," in *Proc. Interspeech*, Singapore, Sep. 2014, pp. 661–665.