Effective Sentence Scoring Method Using BERT for Speech Recognition

Joonbo Shin JBSHIN@SNU.AC.KR
Yoonhyung Lee CPI1234@SNU.AC.KR
Kyomin Jung KJUNG@SNU.AC.KR
Seoul National University

Editors: Wee Sun Lee and Taiji Suzuki

Abstract

In automatic speech recognition, language models (LMs) have been used in many ways to improve performance. Some of the studies have tried to use bidirectional LMs (biLMs) for rescoring the n-best hypothesis list decoded from the acoustic model. Despite their theoretical advantages over conventional unidirectional LMs (uniLMs), previous biLMs have not given notable improvements compared to the uniLMs in the experiments. This is due to the architectural limitation that the rightward and leftward representations are not fused in the biLMs. Recently, BERT addressed the same issue by proposing the masked language modeling and achieved state-of-the-art performances in many downstream tasks by fine-tuning the pre-trained BERT. In this paper, we propose an effective sentence scoring method by adjusting the BERT to the n-best list rescoring task, which has no fine-tuning step. The core idea of how we modify the BERT for the rescoring task is bridging the gap between training and testing environments by considering the only masked language modeling within a single sentence. Experimental results on the LibriSpeech corpus show that the proposed scoring method using our biLM outperforms uniLMs for the n-best list rescoring, consistently and significantly in all experimental conditions. Additionally, an analysis about where word errors occur in a sentence demonstrates that our biLM is more robust than the uniLM especially when a recognized sentence is short or a misrecognized word is at the beginning of the sentence. Consequently, we empirically prove that the left and right representations should be fused in biLMs for scoring a sentence.

Keywords: language model, bidirectional language model, speech recognition

1. Introduction

Language modeling is the task of assigning a probability to word sequence. A language model (LM) is an essential component in recent automatic speech recognition (ASR) systems. Since the LM captures the possibility of any word sequence, it helps to distinguish between words with similar sounds. Conventionally, LMs have been used to predict the probability of the next word given its preceding words. Many state-of-the-art speech recognition systems have achieved performance improvements with these *unidirectional* LMs (uniLMs), including n-gram LMs Heafield et al. (2013) and recurrent neural network (RNN) LMs Mikolov et al. (2010).

Recently, there have been several studies that use *bidirectional* LMs (biLMs) for ASR in order to capture the full context rather than just the previous words Arisoy et al. (2015);

Chen et al. (2017). They applied their biLMs for the n-best list rescoring task, which is the task of selecting the most likely sentence from the hypothesis list that is recognized from acoustic models. Even though bidirectional networks are superior to unidirectional ones in many applications from phoneme classification Graves and Schmidhuber (2005) to acoustic modeling Chan et al. (2016), previous biLMs for ASR did not show their excellence compared to the uniLMs when applying the LMs to the rescoring. This is because there is no interaction between the past and the future words in the biLMs, although the words on both sides are used to predict the current word. Namely, the left and the right representations are not fused in the biLMs since they use a shallow concatenation of independently encoded representations, and it may limit the biLM's potential.

The same issue has been addressed by the recently suggested model, BERT (Bidirectional Encoder Representations from Transformers) Devlin et al. (2018). In the BERT, the model is mainly trained to predict a masked word from its context in order to enable the model to fuse the left and the right representations, unlike the previous biLMs. Their work proves the importance of the interaction between the past and the future words in language understanding by achieving significant success on many downstream tasks such as text classification Socher et al. (2013) and question answering Rajpurkar et al. (2016). Therefore, the BERT is a promising biLM for the task of the n-best list rescoring Wang and Cho (2019).

In this paper, we develop a new biLM by adjusting the BERT to the rescoring task, and verify the empirical effectiveness of the biLM for ASR. The core idea of developing our biLM is to bridge the gap between training and testing environments. For training, we simplify the training objective and the input pipeline of the original BERT. Specifically, we focus only on the "masked word prediction" task and its relevant pipeline from the original BERT, and discard the "next sentence prediction" task because only one sentence is taken at inference. With our simplifications, we can build an effective sentence-level language scorer using the biLM.

For testing, we mask each word one at a time in a given sentence, and then make the biLM predict the probability of the original word at the masked position from its context. We consider that hiding the target word is essential to obtain the proper probability by preventing the biLM from getting a meaninglessly high probability of the exposed words. The score of the sentence is obtained by aggregating all the probabilities, and this score is used to rescore the n-best list of the speech recognition outputs. Although it may not be a meaningful sentence probability like perplexity, this sentence score can be interpreted as a measure of naturalness of a given sentence conditioned on the biLM.

We conduct experiments on the 1000-hour LibriSpeech ASR corpus Panayotov et al. (2015). We first obtain the n-best hypothesis lists from an acoustic model that we implement, and we use our biLM for rescoring them to reduce the final word error rates (WERs). Our biLM achieves 1.61% and 3.00% absolute reductions in WER on the test-clean and test-other sets, which are the subsets of LibriSpeech corpus, while the WER of the acoustic model is 7.26% and 20.37%. Moreover, we empirically prove that our sentence scoring method that uses biLM significantly outperforms not only the uniLM but the combination of the forward and backward LMs regardless of the experimental conditions. In addition, an analysis of where WERs occur in a sentence shows that the biLM is more robust than the uniLM especially when a recognized sentence is short or a misrecognized word is at the earlier part of the sentence. Through these results, we demonstrate that the left and right representations in the biLM should be fused for scoring a sentence.

To the best of our knowledge, this paper is the first study not only that the biLM is notably better than the uniLM for the n-best list rescoring, but also that the BERT is successfully applied to the task of measuring the naturalness of a given sentence without any fine-tuning procedure.

2. Related Works

There have been several studies on bidirectional recurrent neural network language models (biRNNLMs) in automatic speech recognition (ASR) Shi et al. (2013); Arisoy et al. (2015); Chen et al. (2017). Shi et al. (2013) interpolated the scores obtained from the forward and backward RNNLMs, which were trained independently. Arisoy et al. (2015); Chen et al. (2017) investigated the training of biRNNLMs with jointly conditioned on backward and forward representations. However, the biLMs achieved small or no improvements over their uniLM counterparts for the n-best list rescoring.

In natural language processing (NLP), many bidirectional language models have been studied He et al. (2016); Peters et al. (2018a); Devlin et al. (2018). He et al. (2016) investigated the training of biRNNLMs using noise contrastive estimation. ELMo (Embeddings from Language Models) Peters et al. (2018a) used bidirectional language models in order to obtain the contextualized word representations, which were trained with large plain text. Also, Peters et al. (2018a) proposed the method of transferring the forward and backward RNNLMs in order to improve the performances on many downstream tasks from text classification Socher et al. (2013) to question answering Rajpurkar et al. (2016).

Although most language models are based on RNN, self-attention network (SAN) LMs have recently shown competitive performance on sequence modeling with a slight trade-off between speed and accuracy Peters et al. (2018b); Tang et al. (2018). More recently, SANLMs have drawn a big interest in many NLP communities since BERT Devlin et al. (2018) was proposed and achieved state-of-the-art performances on many NLP tasks. To the best of our knowledge, however, no study is conducted on the n-best list rescoring using BERT. While Wang and Cho (2019) mentioned the rescoring using BERT, but they did not conduct experiments in practice.

3. Background

The main interest of this paper is to demonstrate the superiority of a bidirectional language model (biLM) for sentence scoring over a unidirectional LM (uniLM). For a fair comparison of the biLM and uniLM, the two LMs must have the same architecture. As our biLM is a variant of the BERT Devlin et al. (2018), which consists of the encoder part of the Transformer Vaswani et al. (2017), we also construct the uniLM based on the self-attention network (SAN). Although recurrent neural networks (RNNs) appear to be a natural choice for language modeling, SANLMs have recently shown competitive performance on sequence modeling with a slight trade-off between speed and accuracy Peters et al. (2018b); Tang et al. (2018). From these reasons, this paper only considers the bidirectional SANLM (biSANLM) and the unidirectional SANLM (uniSANLM) for the rescoring tasks.

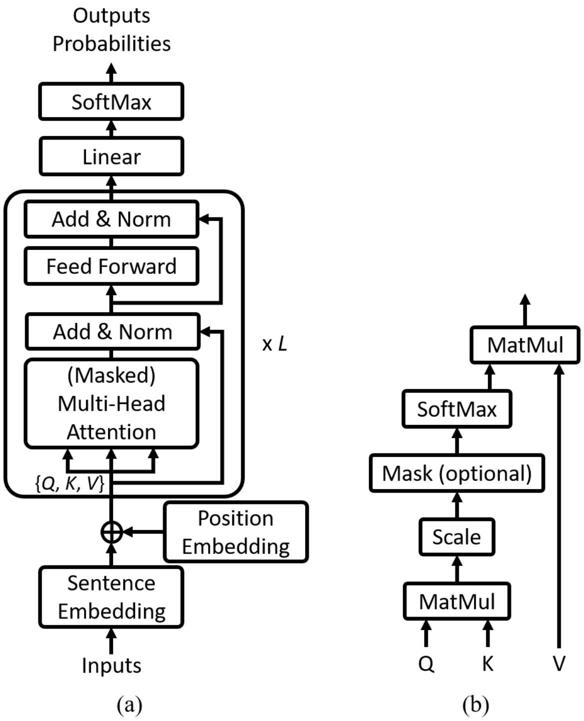

Figure 1: Architectures of (a) the self-attention network language model and (b) the scaled dot-product attention.

3.1. Self-Attention Network Language Model

We deal with the language models (LMs) that are based on the self-attention network (SAN) as shown in Figure 1. Self-attention is an attention mechanism that computes the representation of a single sequence by relating all positions by themselves. As shown in Figure 1b, this computation is done by using the scaled dot-product attention:

$$\text{Attention}(Q, K, V) = \text{Softmax}(\frac{QK^T}{\sqrt{d_k}})V, \tag{1}$$

where Q, K, V are query, key, value matrices respectively, which are generated from the input sequence $X^l \in \mathbb{R}^{n \times d}$ with the number of words n and the input dimension d. To leverage the capacity of the SAN, multi-head self-attention is applied:

$$\begin{aligned} \text{MultiHead}(Q, K, V) &= \text{Concat}(\text{head}_1, \dots, \text{head}_h)W^O, \\ \text{where head}_i &= \text{Attention}(QW_i^Q, KW_i^K, VW_i^V), \end{aligned} \tag{2}$$

$W_i^Q, W_i^K, W_i^V \in \mathbb{R}^{d \times d_k}$ and $W^O \in \mathbb{R}^{d_k h \times d}$ are the parameter matrices for projections with the number of heads h, and $d_k = d/h$ is used for reducing the computational cost. The position-wise feed-forward network, the layer normalization with the residual connection, and dropout are also used in the SAN module for effective training of the model as in the original Transformer Vaswani et al. (2017).

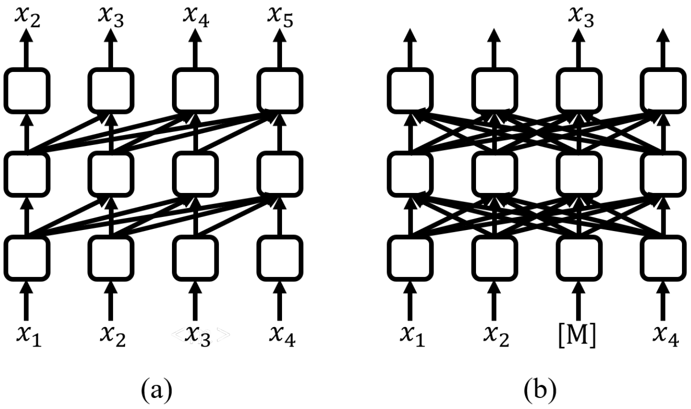

Figure 2: Schematic diagrams of (a) the unidirectional SANLM for next word prediction and (b) the bidirectional SANLM for masked word prediction.

In the unidirectional SANLM (uniSANLM), the optional operation of masking the key-query attention in the scaled dot-product attention should be used. This masking operation prevents words from attending to the future words by making the upper triangle of the key-query attention to be 0 as in the decoder of the Transformer Vaswani et al. (2017). To make the LM aware of the order of the words in the sequence, the position embeddings $X_P \in \mathbb{R}^{n\times d}$ is added to the sentence embeddings $X_S \in \mathbb{R}^{n\times d}$ at the bottom of the encoder, and thus we have $X^l = X_S + X_P$ as the input of the LM. On top of this input, we can build an encoder by stacking the SAN layers as many as we want. The output sequence of the highest SAN, $Y^L \in \mathbb{R}^{n\times d}$ with the number of layers L, is used to predict the probabilities of the words through the softmax layer with the linear projection. The uniSANLM is trained with the "next word prediction" task, and Figure 2a shows an example of the uniSANLM that predicts next word using only its preceding words. We consider the sum of all log-likelihoods of each word in an input sentence as the sentence score of the uniSANLM.

3.2. BERT

BERT is a recently proposed language representation model that consists of a multi-layer bidirectional Transformer encoder Devlin et al. (2018). Unlike the traditional LM that predicts a word from its left context, BERT predicts a word from its left and right context as depicted in Figure 2b. In training the BERT, we mask some words and let the model predict the original words before they are masked, and this task is called "masked language modeling". The BERT benefits a lot from the fusing of the left and right representations, and it can achieve state-of-the-art performances by transferring the pre-trained BERT to many downstream tasks Devlin et al. (2018).

The original BERT has one more training objective called "next sentence prediction", which is designed to learn the relationship between two sentences. For this objective, the BERT has an additional construction of adding segment embedding as well as the corresponding input processing. The next sentence prediction task is proven to be beneficial to some downstream tasks such as question answering and natural language inference, which need understanding sentence relationship.

4. Bidirectional Language Model for Sentence Scoring

In this section, we present the sentence scoring method which uses bidirectional language model (LM) for rescoring the n-best list. First, we outline the construction of our bidirectional SANLM (biSANLM) which is a variant of the BERT Devlin et al. (2018). we then introduce the procedure of the sentence scoring using biSANLM.

4.1. BERT as BiSANLM

We now explain the construction of our biSANLM, which is used for scoring a given sentence in the next section. The core idea of developing our biSANLM is to bridge the gap between the training and testing environments. Note that the architecture of the biSANLM is the same with that of uniSANLM for a fair comparison, including the summation of sentence embedding and position embeddings, the SAN layer, and the softmax layer (Figure 1a). Unlike the uniSANLM, however, the biSANLM do not use the masking operation so as to catch the left and right context during the sentence scoring.

Our training approach has many differences from that of the BERT because our purpose of training bidirectional LM is for scoring a sentence rather than for fine-tuning the model to the other task Devlin et al. (2018). To train our biSANLM, we only consider the masked word prediction task from the BERT Devlin et al. (2018), and make several adjustments: First, our training instance has a single sentence (maximum of 128 words) instead of multiple sentences. Second, we randomly sample some words from the sentence like in the BERT, but replace them by `[MASK]` tokens *all the time* unlike in the BERT Devlin et al. (2018). Lastly, the maximum number of masked tokens in a training instance is limited by the small number of 4, because our instance has only one sentence and too much loss in information is unhelpful to train the model. Note that we make the training instances have multiple masked tokens `[MASK]` for efficient training, while we make the inference instance have only one `[MASK]` for the sentence scoring.

In addition, we neglect the next sentence prediction task of the BERT for training our biSANLM because this objective is not designed to learn to evaluate the probability of a sentence. We also exclude the segment embedding from the input representation of the original BERT, and ignore the `[CLS]` and `[SEP]` tokens from input processing and our vocabulary. Even in our preliminary study, we observe that this task is not helpful to language modeling, but rather hampers the predicting the masked word. Discarding the next sentence prediction task with its corresponding input processing is another difference between our biSANLM and the original BERT.

4.2. Sentence Scoring Method Using BiSANLM

This section introduces our sentence scoring method that adopts the masked word prediction of the BERT Devlin et al. (2018). The basic principle of our sentence scoring is to mask one word in a given sentence and then compute the probability of the original word on the masked position using the trained biSANLM. Because the whole sentence with the masked word is taken to the model as an input, both past and future representations can be fused during prediction.

Our sentence scoring method takes the following procedure: First, we create a set of instances from a given sentence by replacing each word with the predefined token `[MASK]` one at a time. For example, if the sentence has seven words, we create seven instances as below:

- **A given sentence**:

  `move the vat over the hot fire`

- **A set of instances we create**:

  1. Input = `[MASK] the vat over the hot fire`

     Label = `move`
  2. Input = `move [MASK] vat over the hot fire`

     Label = `the`

     . . .
  7. Input = `move the vat over the hot [MASK]`

     Label = `fire`

After the creation, our bidirectional LM takes each instance and computes the likelihood of the original word in the masked position as shown in Figure 2b. Finally, the score of the given sentence is obtained by summing all log-likelihoods of the masked words from each input instance. Although it may not be a sentence probability as of traditional LM, the score can still be used for the n-best list rescoring conditioned on the biSANLM. We note that the past and future words are connected through the designated token `[MASK]` in the input instance, and thus we can make our biSANLM have interactions between both sides without making the prediction task trivial.

5. Experimental Setups

We evaluate the proposed approach on the LibriSpeech ASR task Panayotov et al. (2015). The 960-hour of training data is used to train an acoustic model, which is our base speech recognition system. We obtain the 100-best hypothesis list for each audio in development and test data using the acoustic model, and then use language models (LMs) to rescore these 100-best lists. For comparison, we use our biSANLM, the forward uniSANLM, and the backward uniSANLM. The details of the acoustic model and language model settings are explained in the following sections.

5.1. Acoustic Model

In this study, we use the attention-based seq2seq model *Listen, Attend and Spell* (LAS) Chan et al. (2016) as our acoustic model with some differences. First, there are additional bottleneck fully connected (FC) layers between every bidirectional long-short term memory (BLSTM) layer. Second, the number of time steps is reduced in half by just subsampling hidden states for even number time steps before the FC layer, instead of concatenating every two hidden states. Third, LAS is trained with additional CTC objective function because the left-to-right constraint of CTC helps LAS learn alignments between speech-text pairs Hori et al. (2017).

The details of our acoustic model follow the default settings provided in ESPNet toolkit v.0.2.0 Watanabe et al. (2018). For the input features, we use 80-band mel-scale spectrogram derived from the speech signal. The encoder consists of 5-layer pyramidal-BLSTM with subsampling after second and third layers. The decoder is comprised of 2-layer LSTM with location-aware attention mechanism Chorowski et al. (2015). The target sequence is processed in 5K case-insensitive sub-word units created via unigram byte-pair encoding Shibata et al. (1999). All the LSTM and FC layers have 1024 hidden units each. Our model is trained for 10 epochs using Adadelta optimizer Zeiler (2012) with learning rate of 1e-8. Using this acoustic model, we obtain 100-best decoded sentences for each input through hybrid CTC-attention based scoring Hori et al. (2017) method, and these 100-best lists will be used for rescoring. Table 1 shows the word error rates (WERs) obtained from the acoustic model and the oracle WERs, which is the best possible errors of the 100-best lists on the LibriSpeech tasks.

Table 1: Oracle WERs of the 100-best lists on LibriSpeech

Method	dev		test	
	clean	other	clean	other
1-best	7.17	19.79	7.26	20.37
100-best (oracle)	2.85	12.21	2.81	12.85

5.2. Language Model Setups

The model parameters of our language model (LM) are as follows: $L = 3$ for the number of layers, $d = 512$ for the dimensions of the model and the embeddings, $h = 8$ for the number of head. 2048 hidden units are used in the position-wise feed-forward layers. We use trainable positional embeddings with supported sequence lengths up to 128 tokens. We use a *gelu* activation Hendrycks and Gimpel (2016) rather than the standard *relu*, following Radford et al. (2018); Devlin et al. (2018). Weight matrix of the softmax layer is shared with the word embedding table. The word vocabulary is used in three sizes: 10k, 20k and 40k most frequent words. For a fair comparison in terms of the number of parameters, our biSANLM and uniSANLMs have the same architecture and parameters.

We train the LMs with the 1.5G normalized text-only data of the official LibriSpeech corpus. We use Adam optimizer Kingma and Ba (2014) with learning rate of 1e-4, $\beta_1 = 0.9$, $\beta_2 = 0.999$. We use a dropout probability of 0.1 on all layers. Batch size is set to 128 for biSANLMs and 64 for uniSANLMs, and all the LMs are trained for 1M iterations. We confirmed that all our LMs are converged before the 1M training steps.

5.3. Score Interpolation

In this work, we consider the sum of all log-likelihoods of a masked word in each input sentence as the sentence score of the LMs. Following the previous works on bidirectional language models for speech recognition Arisoy et al. (2015); Chen et al. (2017), we use our sentence score for rescoring the n-best hypotheses. We linearly interpolate the scores obtained by the acoustic model (AM) and the language model (LM):

$$\text{score} = (1 - \lambda) \cdot \text{score}_{\text{AM}} + \lambda \cdot \text{score}_{\text{LM}}, \tag{3}$$

where λ is the interpolation weight, which is determined empirically on development data. For a fair comparison in terms of information, we average the scores of the forward and the backward SANLMs like $\text{score}_{\text{LM}} = (\text{score}_{\text{uniSANLM}_{fw}} + \text{score}_{\text{uniSANLM}_{bw}})/2$, which is an ELMo-like biLM.

6. Results and Discussion

In this section, we compare the LMs for the n-best rescoring on all test sets of the LibriSpeech ASR corpus, in which the test sets are classified as "clean" or "other" set based on their difficulties. We first prepare 100-best hypotheses using our acoustic model (AM), which is a base speech recognition system in our experiments. For rescoring the 100-best list, the AM is linearly interpolated with one or two of LMs as in Equation 3.

Table 2: WERs for unidirectional and bidirectional SANLMs interpolated with the baseline model on LibriSpeech

Model	$\lvert V \rvert$	dev		test	
		clean	other	clean	other
AM only		7.17	19.79	7.26	20.37
+ biSANLM	10k	5.65	16.85	5.69	17.59
	20k	5.57	16.71	5.68	**17.37**
	40k	**5.52**	**16.61**	**5.65**	17.44
+ uniSANLM$_{fw}$	10k	6.09	17.50	6.08	18.33
	20k	6.05	17.48	6.11	18.25
	40k	6.08	17.32	6.11	18.13
+ uniSANLM$_{bw}$	10k	6.15	17.78	6.24	18.51
	20k	6.17	17.60	6.22	18.49
	40k	6.17	17.57	6.24	18.29
+ uniSANLM$_{fw}$ + uniSANLM$_{bw}$	10k	6.11	17.71	6.15	18.41
	20k	6.12	17.52	6.16	18.32
	40k	6.16	17.42	6.18	18.22

Table 2 shows rescoring results of the biSANLMs and the other LMs with different test sets and different vocabulary sizes $|V|$. The WER results show that the biSANLM with our approach is consistently and significantly better than the uniSANLM regardless of the test set and the vocabulary size. While WER reductions are also observed from the backward SANLMs, amounts of WER reduction are smaller than the forward SANLMs. Moreover, combining the forward and the backward SANLMs is not helpful to rescoring the n-best list. The results demonstrate that the fusion of the left and right representations is important to predict a score of a given sentence.

The interpolation weight is set to a value that achieves the best performance in the development sets. We find that $\lambda = 0.2$ and 0.3 are the best weights for dev-clean and dev-other sets respectively. Considering that the dev-other set is more difficult for the acoustic model to recognize, it is reasonable to have larger interpolation weight in dev-other ($\lambda = 0.3$) than in dev-clean ($\lambda = 0.2$).

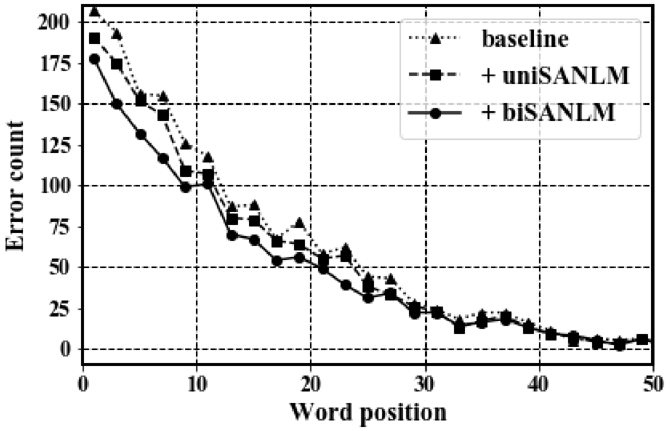

Figure 3: Error count by word position

To see where the "word error" occurs, we analyze the position of the misrecognized words. Figure 3 shows the total number of the misrecognized words for each model according to the position of the final hypotheses. It can be seen that the biSANLM is more robust than the uniSANLM at the earlier position (< 30) of a sentence. At the latter position (> 30) of a long sentence, however, the gap between the two LMs is reduced. This fact shows that, although the uniSANLM also performs well, our biSANLM is more effective particularly when a recognized sentence is short or a misrecognized word is at the beginning of the sentence. In this analysis, we use $|V| = 40$k for both LMs, and the choice of vocabulary size does not affect the tendency.

To see greater performance improvements, we conduct linear interpolation of the biSANLM and the uniSANLM for further improvements:

$$\text{score}_{\text{LM}} = (1 - \alpha) \cdot \text{score}_{\text{uniSANLM}} + \alpha \cdot \text{score}_{\text{biSANLM}},$$

where α is another interpolation weight and score_{LM} is used in Equation 3. We find $\alpha = 1$ shows the best performances on all dev sets, which means only the biSANLM is used for interpolation (log-linear interpolation of the two LMs shows the same phenomenon). Contrary to our first expectation, the biSANLM and the uniSANLM do not complement each other in our experiments.

Consequently, all experimental results demonstrate that our sentence scoring method using the biSANLM is almost strictly better than the traditional method using uniSANLM for the n-best list rescoring. As far as we know, this is the first study that the bidirectional language model significantly and consistently outperforms the unidirectional language model for speech recognition.

7. Conclusion

This paper proposed a sentence scoring method using a bidirectional language model (biLM) and verified its effectiveness for the n-best list rescoring in automatic speech recognition. By adapting BERT to the sentence scoring, the left and right representations are fused in our biLM unlike previous biLMs for ASR. Experimental results on the LibriSpeech ASR tasks show that the proposed sentence scoring method with our biLM significantly and consistently outperforms the conventional uniLM for rescoring the n-best list. In addition, we confirm that the biLM is more robust than the uniLM especially when a recognized sentence is short or the earlier part of the sentence is misrecognized.

Acknowledgments

K. Jung is with Automation and Systems Research Institute (ASRI), Seoul National University, Seoul, Korea. This work was supported by the National Research Foundation of Korea (NRF) funded by the Korean government (MSIT) (No. 2016M3C4A7952632), Industrial Strategic Technology Development Program (No. 10073144) funded by the Ministry of Trade, Industry & Energy (MOTIE, Korea).

References

Ebru Arisoy, Abhinav Sethy, Bhuvana Ramabhadran, and Stanley Chen. Bidirectional recurrent neural network language models for automatic speech recognition. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5421–5425. IEEE, 2015.

William Chan, Navdeep Jaitly, Quoc Le, and Oriol Vinyals. Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4960–4964. IEEE, 2016.

Xie Chen, Anton Ragni, Xunying Liu, and Mark JF Gales. Investigating bidirectional recurrent neural network language models for speech recognition. In *INTERSPEECH*, pages 269–273, 2017.

Jan K Chorowski, Dzmitry Bahdanau, Dmitriy Serdyuk, Kyunghyun Cho, and Yoshua Bengio. Attention-based models for speech recognition. In *Advances in neural information processing systems*, pages 577–585, 2015.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Alex Graves and Jürgen Schmidhuber. Framewise phoneme classification with bidirectional lstm and other neural network architectures. *Neural Networks*, 18(5-6):602–610, 2005.

Tianxing He, Yu Zhang, Jasha Droppo, and Kai Yu. On training bi-directional neural network language model with noise contrastive estimation. In *2016 10th International Symposium on Chinese Spoken Language Processing (ISCSLP)*, pages 1–5. IEEE, 2016.

Kenneth Heafield, Ivan Pouzyrevsky, Jonathan H Clark, and Philipp Koehn. Scalable modified kneser-ney language model estimation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 690–696, 2013.

Dan Hendrycks and Kevin Gimpel. Bridging nonlinearities and stochastic regularizers with gaussian error linear units. *arXiv preprint arXiv:1606.08415*, 2016.

Takaaki Hori, Shinji Watanabe, Yu Zhang, and William Chan. Advances in joint ctc-attention based end-to-end speech recognition with a deep cnn encoder and rnn-lm. *Proc. Interspeech 2017*, pages 949–953, 2017.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Tomáš Mikolov, Martin Karafiát, Lukáš Burget, Jan Černockỳ, and Sanjeev Khudanpur. Recurrent neural network based language model. In *Eleventh annual conference of the international speech communication association*, 2010.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: an asr corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5206–5210. IEEE, 2015.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, volume 1, pages 2227–2237, 2018a.

Matthew Peters, Mark Neumann, Luke Zettlemoyer, and Wen-tau Yih. Dissecting contextual word embeddings: Architecture and representation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1499–1509, 2018b.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. *URL https://s3-us-west-2. amazonaws. com/openai-assets/research-covers/languageunsupervised/language understanding paper. pdf*, 2018.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*, 2016.

Yangyang Shi, Martha Larson, Pascal Wiggers, and Catholijn M Jonker. Exploiting the succeeding words in recurrent neural network language models. 2013.

Yusuxke Shibata, Takuya Kida, Shuichi Fukamachi, Masayuki Takeda, Ayumi Shinohara, Takeshi Shinohara, and Setsuo Arikawa. Byte pair encoding: A text compression scheme that accelerates pattern matching. Technical report, Technical Report DOI-TR-161, Department of Informatics, Kyushu University, 1999.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642, 2013.

Gongbo Tang, Mathias Müller, Annette Rios, and Rico Sennrich. Why self-attention? a targeted evaluation of neural machine translation architectures. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4263–4272, 2018.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008, 2017.

Alex Wang and Kyunghyun Cho. Bert has a mouth, and it must speak: Bert as a markov random field language model. In *Proceedings of the Workshop on Methods for Optimizing and Evaluating Neural Language Generation*, pages 30–36, 2019.

Shinji Watanabe, Takaaki Hori, Shigeki Karita, Tomoki Hayashi, Jiro Nishitoba, Yuya Unno, Nelson-Enrique Yalta Soplin, Jahn Heymann, Matthew Wiesner, Nanxin Chen, et al. Espnet: End-to-end speech processing toolkit. *Proc. Interspeech 2018*, pages 2207–2211, 2018.

Matthew D Zeiler. Adadelta: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*, 2012.