KNOWLEDGE DISTILLATION USING OUTPUT ERRORS FOR SELF-ATTENTION END-TO-END MODELS

Ho-Gyeong Kim[*], *Hwidong Na*[*], *Hoshik Lee, Jihyun Lee,*
Tae Gyoon Kang, Min-Joong Lee, Young Sang Choi

Samsung Advanced Institute of Technology, Samsung Electronics, South Korea
{hogyeong.kim, hwidong.na}@samsung.com

ABSTRACT

Most automatic speech recognition (ASR) neural network models are not suitable for mobile devices due to their large model sizes. Therefore, it is required to reduce the model size to meet the limited hardware resources. In this study, we investigate sequence-level knowledge distillation techniques of self-attention ASR models for model compression. In order to overcome the performance degradation of compressed models, our proposed method adds an exponential weight to the sequence-level knowledge distillation loss function, which reflects the word error rate of the output of the teacher model based on the ground-truth word sequences. Evaluated on LibriSpeech dataset, the proposed knowledge distillation method achieves significant improvements over the student baseline model.

Index Terms— Automatic speech recognition, knowledge distillation, model compression, self-attention end-to-end model

1. INTRODUCTION

On-device automatic speech recognition (ASR) services are crucial for mobile devices. For customers, especially when online connection is unavailable, on-device ASR services are useful. For service providers, they reduce the cost of providing ASR services on cloud servers via network connection. For on-device ASR, however, it is required to reduce the size of a model in order to fit in the limited hardware specs of mobile devices. For example, an n-gram language model (LM) used in a server-based hybrid ASR system often occupies multiple gigabytes on disk [1]. Large recurrent neural networks (RNNs) tend to have huge memory footprint, which slows down the running time and drains battery on mobile devices.

Model compression techniques have been widely studied for image classification models. Knowledge distillation (KD) is one of the well-studied techniques [2]. An original *teacher* model, which performs the task well enough, guides a compressed *student* model. As a consequence, the student model retains the *distilled knowledge* of the teacher model. In general, a student model is trained with following loss function for KD.

$$\mathcal{L} = (1-\alpha)\mathcal{L}_{NLL} + \alpha\mathcal{L}_{KD} \quad (1)$$

The negative log-likelihood $\mathcal{L}_{NLL}$ is the training loss of the original task, and the KD loss function is interpolated to minimize the differences between the teacher and student models. For example, a token-level KD loss function makes the student learn the output of the teacher by minimizing the distributions of each token between models. A sequence-level KD loss function, on the other hand, gives the student the chance to learn the complete sequences generated by the teacher [3]. Instead of exponentially many candidate sequences in theory, beam search provides an approximation for the complete sequences. Recently, sequence-level KD loss functions were introduced for compressing end-to-end (E2E) ASR models [4] [5] [6].

In this paper, we reflect the quality of the output from the teacher model to minimize the sequence-level KD loss function via weighing scheme. Previous KD methods did not consider the quality of the recognition results from the teacher model, whether the recognition results are good or bad. In ASR systems, the quality of the recognition can be measured using the word error rate (WER) or character error rate (CER). We propose to penalize less accurate recognition results from the teacher, which have higher WERs. We conducted several experiments on publically available dataset, LibriSpeech [7]. For the ease of reproducibility[1], we utilized an open-source project [8].

2. WHY SELF-ATTENTION FOR E2E ASR?

In an E2E model, a single deep neural network can directly model from the input to the output sequence. An E2E model with attention mechanism was originally proposed for machine translation task [9]. An encoder neural network

[*] Equal contribution
[1] We strongly suggest referring the following information.
https://github.com/tensorflow/tensor2tensor/issues/896

converts the input sequence to the feature sequences, attention mechanism produces the context from the feature sequences, and a decoder neural network iteratively predicts the probability of the next token given the context. E2E models with attention mechanism using deep neural network architectures such as RNNs, CNNs, or self-attention networks were widely studied [10] [11] [12].

E2E models with attention mechanisms have also applied to ASR [1] [13] [14]. Instead of the input text, an E2E model with attention mechanism recognizes text having about 10~20 words or sub-words sequence from a few hundred frames of speech features. Since E2E ASR models learn pronunciation pattern of words or sub-words directly from speech, they are more robust for various speech patterns. For example, an E2E model based on long short-term memory (LSTM) model with attention mechanism (LSTM-E2E for short) outperforms the conventional hybrid ASR [1]. An E2E model based on self-attention (SA-E2E for short) also showed comparable accuracy to LSTM-E2E [14].

We insist that an SA-E2E is more efficient architecture than an LSTM-E2E for the same model size. First, memory footprint of an SA-E2E would be much smaller than that of an LSTM-E2E if properly reuse model parameters. The operations in the encoder of an SA-E2E have no temporal dependencies. The model parameters in the encoder therefore are loaded once on memory and reused for the whole input sequence. Meanwhile, LSTM-E2E parameters cannot be reused and should be loaded for each time step[2]. Second, an SA-E2E is much faster in training in compared to an LSTM-E2E model because no recurrent connection makes the effective parallelization of computation possible. For example, SA-E2Es achieved comparable accuracies of LSTM-E2Es but took less training time in [14]. Hereinafter, we focus on KD for SA-E2Es.

3. KNOWLEDGE DISTILLATION

The goal of KD is to effectively transfer the knowledge of the teacher model to the student model. In terms of model compression, the number of model parameters of the student model is usually much less than the number of parameters of the teacher model. In this case, the accuracy of the student model is often degraded. In order to overcome the accuracy degradation, it is important to transfer the knowledge of the teacher model as much as possible to the student model.

3.1. Sequence-level knowledge distillation

Sequence labeling problem using self-attention based attention model can be defined as the following equation. The training criteria for ASR is to minimize negative log-likelihood (NLL) for each sample $\mathbf{s}$ from the training data,

$$\mathcal{L}_{NLL} = -\sum_{j=1}^{J}\sum_{c=1}^{|C|}\delta(y_j, c)\log p(t_j = c|\mathbf{s};\theta) = -\log p(\mathbf{t} = \mathbf{y}|\mathbf{s};\theta) \quad (2)$$

where y_j and t_j is the j-th token of the ground-truth sequence $\mathbf{y}$ and the model output sequence $\mathbf{t}$, respectively, C indicates the output vocabulary, δ is the Kronecker delta, and $p(t_j = c|\mathbf{s};\theta)$ is the output distribution from θ which is a student model parameter. In other words, this objective can be seen as minimizing the cross-entropy between the ground-truth target distribution and the model output probability distribution.

Sequence-level KD introduces a loss function such that output sequences from teacher model are used as another ground-truth sequence to train student models [3]. Then, a student model is trained with output sequences from the teacher over all possible sequences as follows:

$$\mathcal{L}_{\text{SEQ-KD}} = -\sum_{\mathbf{t}\in\mathcal{T}} q(\mathbf{t}|\mathbf{s};\theta_T)\log p(\mathbf{t}|\mathbf{s};\theta) \approx -\log p(\mathbf{t} = \hat{\mathbf{y}}|\mathbf{s};\theta) \quad (3)$$

where $\hat{\mathbf{y}}$ denotes the top-1 output sequence of the teacher, q is the output distribution from θ_T which is pre-learned model parameters of the teacher model. As an approximation of the all possible sequences $\mathcal{T}$, we replace q with the beam search result.

3.2. Knowledge distillation using output errors

We can extend the sequence-level KD method not only using outputs of the teacher model, but also using error rates between ground-truth target and outputs from the teacher model. This can be expressed as follows.

$$\mathcal{L}_{\text{ERR-KD}} = \exp(-\beta\,\text{err}(\mathbf{y}, \hat{\mathbf{y}})) * \mathcal{L}_{\text{SEQ-KD}} \quad (4)$$

where β is an exponential weight for the output error term $\text{err}(\mathbf{y}, \hat{\mathbf{y}})$, which indicates an error rate of the output of the teacher model based on the ground-truth word sequence.

As a result, well-recognized training samples contribute to the training more. In other words, the high weights to the KD loss are set to the training samples with the lower WER. On the contrary, with the higher WER, the less is contributed to learning. Here, β acts as a smoothing factor. When β is high, the KD loss weights of training samples are set as small values compared to the case of small β. Regardless of the output token units (word-pieces in our case), sequence-level error rate can be measured by any units such as a words or characters. Different granularity

[2] It would be possible to reuse LSTM-E2E parameters if the size is very small. However, the size of an LSTM-E2E (>100Mbytes) is usually too large to load on static random-access memory (<2Mbytes) on a modern mobile device.

Table 1. The teacher and student model architectures. The hidden and filter sizes correspond to the hyper-parameters of baseline transformer models in `tensor2tensor`.

Model	# Enc. Layers	# Dec. Layers	Hidden Size	Filter Size	Size (Bytes)
Teacher	6	4	512	2,048	141M
Student A	8	2	256	1,024	34.5M
Student B	10	3	256	2,048	70.6M

Table 2. WERs [%] of the teacher and student baseline models on LibriSpeech dev and test sets.

Model	WER [%]			
	`dev-clean`	`dev-other`	`test-clean`	`test-other`
Teacher	5.62	14.19	5.75	14.36
Student A	7.89	17.96	7.99	18.37
Student B	5.99	14.69	6.30	14.39

provides the student model to learn the sequence generated by the teacher in different point of view. We used the WER as the error rate.

Although we define the sequence-level KD using only top-1 output sequence, our approach can be extended using the N-best output sequences. The minimum word error rate (MWER) criteria for ASR involved the WER to enhance the model accuracy [15]. We rather weigh the sequence-level KD using the WER, and leave the integration of the MWER criteria for the future work.

4. EXPERIMENTS

We used the LibriSpeech training dataset consisting of 960 hours of read audio books [7]. The test and development sets are divided according to the difficulty of transcription. We report the accuracy of the model for the four datasets (`dev-clean`, `dev-other`, `test-clean`, and `test-other`), where the decoding hyper-parameters (beam size, length-penalty, and maximum decoding length) were tuned to minimize WER on `dev-clean/other` for `test-clean/other`, respectively.

In our experiments, we designated two student models A and B. The teacher has approximately 4 and 2 times of parameters of student A and B, respectively. The teacher model has 8 heads for the encoder self-attention, and 2 heads for the decoder self-attention and encoder-decoder attention, while student models have half number of heads of teachers. We used 1024 word pieces as output token units. Table 1 describes the details of the model architectures. The numbers of model parameters are 37 million, 9 million and 18 million for teacher, student A, and student B models, respectively.

All experiments used the identical input feature processing to that of [1]. We employed label smoothing of value $\epsilon_{ls} = 0.15$, and scheduled sampling after 100k gradient updates. A probability to sample an utterance is 0.2 and a probability to sample a token is 0.2. We used the Adam optimizer with $\beta_1 = 0.9, \beta_2 = 0.98$ and $\epsilon = 10^{-9}$. It was increasing the learning rate linearly for 8,000 steps, and set to 0.02 until convergence.

We also applied dropout to all attention weights using rate of 0.1, to the output of rectified linear unit of 0.2, to the output of each sub-layer, before it was added to the sub-layer input using rate of 0.2, to the sums of the output of the input transform and the positional encodings in the encoder using rate of 0.2, and the sums of the target embedding and the positional encodings in the decoder using rate of 0.2.

The recognition accuracy of E2E ASR models can be improved by using an external language model. We trained a language model on the same word piece vocabulary as our SA-E2E models and used the normalized training corpus officially available[3] for LibriSpeech language model. Our LM also follows the self-attention architecture which consists of 3 decoder layers where hidden size = 1024, heads = 8, and filter size = 4096. We integrated the SA-LM with our SA-E2E models by shallow fusion which is a log-linear interpolation between the output-probabilities of two models at each time-step during beam search [16]. We performed a grid search to find the best language model and its interpolation weight for each E2E ASR model minimizing WER on `dev-clean`.

5. RESULTS

5.1. Error-based knowledge distillation results

Table 2 shows the accuracy of the teacher and student baseline models in WERs. As the model size decreased, the recognition accuracy was degraded more. For the half size model of the teacher, WERs were increased 9.57% and 0.21% on `test-clean` and `test-other`, respectively. When the model size was reduced to 75% of the teacher model size, WERs increased from 5.75% to 7.99% on `test-clean`, yielding a relative increase of 38.9%.

The effectiveness of the knowledge distillation methods is shown in Table 3. In the experiments, we tested $\alpha \in \{0.3, 0.7\}$ and explored the exponential weight factor β in a set of $\{0.5, 1.0, 2.0\}$. As shown in the table, overall knowledge distillation results outperformed the results of the student models which trained from the scratch. The WER of SEQ-KD on `test-clean` was 6.19% for student B model, yielding a relatively 1.75% reduction of the student baseline

[3] http://www.openslr.org/11/

Table 3. WERs [%] of knowledge distillation methods on LibriSpeech dev and test sets.

Model	KD (α, β)	WER [%]			
		dev-clean	dev-other	test-clean	test-other
Student A	SEQ-KD (0.7, 0.0)	7.69	17.79	7.82	17.90
	ERR-KD (0.7, 0.5)	7.66	17.75	7.82	17.84
	SEQ-KD (0.3, 0.0)	7.64	17.77	7.77	17.98
	ERR-KD (0.3, 2.0)	7.62	17.79	7.79	17.76
Student B	SEQ-KD (0.7, 0.0)	5.87	14.26	6.19	14.33
	ERR-KD (0.7, 2.0)	5.85	14.25	6.14	14.27
	SEQ-KD (0.3, 0.0)	5.86	14.33	6.21	14.30
	ERR-KD (0.3, 1.0)	5.83	14.26	6.16	14.22

when α is equal to 0.7. The WERs of 6.16% and 14.22% on `test-clean` and `test-other` for student B were achieved for the ERR-KD method, yielding a relative reduction of 2.22% and 1.18% over the student baseline. This indicated that errors of the teacher outputs provide the additional information to distill the teacher model better than the SEQ-KD. Interestingly, the knowledge distillation results on `test-other` set for student B outperformed the results of the teacher baseline. It revealed the student models preserved the power of the teacher model not only for the easy cases, but also for the difficult ones. For student A, substantial WER improvements of 2.5% and 3.3% for the ERR-KD method were achieved on `test-clean` and `test-other`, respectively.

5.2. Comparisons to related works

The authors of [4] conducted various compression techniques to the "Listen, Attend, and Spell (LAS)" model [13]. One of their methods is a sequence-level KD aims that a student to minimize KL divergence toward to the distribution of the teacher. Their sequence-level KD loss not only utilized the hypotheses as ground-truth, but also the distribution of tokens obtained from the teacher. E2E models based on the Connectionist Temporal Classification (CTC) were also compressed in [5] and [6], where the posterior probability of the phoneme for each frame is the target of KL divergence. Those models, however, were based on LSTMs which require heavy memory footprints, not suitable for mobile device environments.

Although previous works presented similar idea to this article, we report novel results that adopt the knowledge distillation on an SA-E2E to the best of our knowledge. Note that our teacher model without the external LM is a strong baseline, which achieved the WER of 14.36 on the `test-other`, comparing with the LAS model in [17]. The WER improvements of the proposed method with LM were similar for the teacher and the student model. Moreover, our proposed method combined with an SA-LM outperformed previous works on `test-clean` except LAS+LM, and on `test-other`, which is more difficult to recognize. Table 4 summarizes WERs of prior works and our implementations on LibriSpeech corpus.

Table 4. Comparison of WERs [%] on LibriSpeech dev and test sets.

Model	WER [%]			
	dev-clean	dev-other	test-clean	test-other
Prior works				
LAS [17]	4.87	14.37	4.87	15.39
LAS + LM [17]	3.54	11.52	3.82	12.76
CTC, PL [18]	5.10	14.26	5.42	14.70
CTC [19]	-	-	5.33	13.25
Our Implementation				
Teacher	5.62	14.19	5.75	14.36
Teacher + SA-LM	4.42	10.65	4.73	10.90
ERR-KD	5.83	14.26	6.16	14.22
ERR-KD + SA-LM	4.67	11.03	5.13	11.20

6. CONCLUSIONS

In this paper, we investigate KD-based model compression method for the self-attention end-to-end model to overcome the degraded performance of models with small size. Previous KD methods deal with the output sequences only from the pre-learned teacher model. However, those approaches do not consider the quality of the recognition results from the teacher model. To overcome the drawback of previous KD methods, we apply the error rate of the output sequences of the teacher model based on the ground-truth target sequences to the KD loss function. We demonstrated that the WERs were improved over the baseline on LibriSpeech corpus, clearly showing that the recognition performance was meaningful using the weighted loss function. Finally, we emphasize that the proposed KD method can be applied on other recognition systems that are capable of providing output errors given output sequences, which are generated from the teacher model.

7. REFERENCES

[1] C.-C. Chiu and T. N. Sainath, "State-of-the-art Speech Recognition With Sequence-to-Sequence Models," in *ICASSP,* 2018.

[2] G. Hinton, O. Vinyals and J. Dean, "Distilling the Knowledge in a Neural Network," in *CoRR,* 2015.

[3] Y. Kim and A. M. Rush, "Sequence-Level Knowledge Distillation," in *EMNLP*, 2016.

[4] R. Pang, T. N. Sainath, R. Prabhavalkar, S. Gupta, Y. Wu, S. Zhang and C.-c. Chiu, "Compression of End-to-End Models," in *Interspeech*, 2018.

[5] M. Huang, Y. You, Z. Chen, Y. Qian and K. Yu, "Knowledge Distillation for Sequence Model," in *Interspeech*, 2018.

[6] R. Takashima, S. Li and H. Kawai, "An Investigation of a Knowledge Distillation Method for CTC Acoustic Models," in *ICASSP*, 2018.

[7] V. Panayotov, G. Chen, D. Povey and S. Khudanpur, "LibriSpeech: an ASR corpus based on public domain audio books," in *ICASSP*, 2015.

[8] A. Vaswani, S. Bengio , E. Brevdo , F. Chollet, A. Gomez , S. Gouws , L. Jones, Ł. Kaiser, N. Kalchbrenner , N. Parmar, R. Sepassi , N. Shazeer and J. Uszkoreit, "Tensor2Tensor for Neural Machine Translation," in *CoRR,* 2018.

[9] D. Bahdanau, C. Kyunghyun and Y. Bengio, "Neural Machine Translation by Jointly Learning to Align and Translate," in *arXiv,* p. arXiv:1409.0473, 2014.

[10] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, J. Klingner, A. Shah, M. Johnson, X. Liu, Ł. Kaiser, S. Gouws and J. Dean, "Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation," in *arXiv,* p. arXiv:1609.08144, 2016.

[11] J. Gehring, M. Auli, D. Grangier, D. Yarats and Y. N. Dauphin, "Convolutional Sequence to Sequence Learning," in *ICML*, 2017.

[12] A. Waswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser and I. Polosukhin, "Attention Is All You Need," in *NIPS,* 2017.

[13] W. Chan, N. Jaitly, Q. V. Le and O. Vinyals, "Listen, Attend and Spell," in *ICASSP,* 2016.

[14] L. Dong, S. Xu and B. Xu, "Speech-transformer: a No-recurrence Sequence-to-Sequence Model for Speech Recognition," in *ICASSP*, 2018.

[15] R. Prabhavalkar, T. N. Sainath, Y. Wu, P. Nguyen, Z. Chen, . C.-C. Chiu and A. Kannan, "Minimum Word Error Rate Training for Attention-based Sequence-to-Sequence Models," in *ICASSP*, 2018.

[16] C. Gulcehre, O. Firat, K. Xu, K. Cho, L. Barrault, H.-C. Lin, F. Bougares, H. Schwenk and Y. Bengio, "On Using Monolingual Corpora in Neural Machine Translation," in *arXiv,* p. arXiv:1503.03535, 2015.

[17] A. Zeyer, K. Irie, R. Schluter and H. Ney, "Improved training of end-to-end attention models for speech recognition," in *ICASSP*, 2018.

[18] Y. Zhou, C. Xiong and R. Socher, "Improving End-to-End Speech Recognition with Policy Learning," in *ICASSP*, 2018.

[19] D. Amodei, . R. Anubhai, E. Battenberg, . C. Case, J. Casper, B. Catanzaro, J. Chen, . M. Chrzanowski, . A. Coates, G. Diamos , E. Elsen, J. Engel , L. Fan, . C. Fougner, . T. Han , A. Y. Hannun, B. Jun, P. LeGresley, L. Lin, . S. Narang, A. Y. Ng, S. Ozair, R. Prenger, J. Raiman, S. Satheesh, . D. Seetapun, S. Sengupta , Y. Wang , Z. Wang, C. Wang , B. Xiao , D. Yogatama , J. Zhan and Z. Zhu, "Deep speech 2: End-to-end speech recognition in english and mandarin," in *ICML*, 2016.