Hybrid CTC/Attention Architecture with Self-Attention and Convolution Hybrid Encoder for Speech Recognition

Mingxin Nie*, Zhichao Lei
School of Wuhan University of Technology, Wuhan 430070, China
*Corresponding author e-mail: 624968304@qq.com, 2653088672@qq.com

Abstract. The success of self-attention in NLP has led to recent applications in end-to-end encoder-decoder architectures for speech recognition. The transformer model based on self-attention get a promising result. As you can see, the hybrid model based on Connectionist Temporal Classification (CTC)/Attention has very prominent advantages in decoding, which can combine the excellent sequence-to-sequence modeling ability of attention, and can also combine CTC to achieve temporal alignment. We propose SA-Conv-CTC/Attention model, which apply a Self-Attention and shallow Convolution based hybrid encoder to Hybrid CTC/Attention Architecture, and we also explored the method of decoding with Self-Attention language models. The CTC network sits on top of the encoder and is jointly trained with the attention-based decoder, even participated in decoding. We achieve a 0.8-4.75% error reduction compared to other hybrid CTC/Attention systems on WSJ and HKUST dataset.

1. Introduction
A typical ASR system is factorized into several modules including acoustic, lexicon, and language models based on a probabilistic noisy channel model [1, 2]. In recent years, end-to-end speech recognition systems have become more and more popular in academia. Unlike traditional hybrid models, end-to-end speech recognition systems directly convert input speech into characters, words, or other tokens [1-4]. The end-to-end speech recognition system only independently trains a complete system based on paired voice data and language data without relying on other independent components. Therefore, the approach potentially makes it possible to build ASR systems without prior knowledge [1].

There are mainly three types of end-to-end architectures for Automatic Speech Recognition: attention-based methods [11, 12] and Connectionist Temporal Classification (CTC) [1, 10] and hybrid CTC/attention model [2, 4, 5]. Attention-based methods use an attention mechanism [17, 12] to perform alignment between acoustic frames and recognized symbols, and connectionist temporal classification (CTC) uses Markov assumptions to efficiently solve sequential problems by dynamic programming [10]. It is a fact that using the attention mechanism alone cannot achieve temporal alignment in spite of the strong modeling capabilities [2, 4]. Though Connectionist Temporal Classification (CTC), uses Markov assumptions to efficiently solve sequential problems by dynamic programming, CTC requires several conditional independences [5] assumptions to obtain the label sequence probabilities, this is not perfect for speech signals that are closely related. Subsequently, various hybrid models have been proposed to solve the above problems [1, 2, 4, 5].
With the application of self-attention mechanism get promising result in machine translation [8], transformer-based models have also been quickly applied to automatic speech recognition (ASR) [6, 7, 9]. The use of transformer, which is based on multi-head self-attention, has achieved state of the art results in speech recognition. However, for a desired result, we need massive data to train transformer based model [6, 9]. But when the database is not large, the hybrid model can also perform satisfactorily.

Inspired by the knowledge above, I proposed SA-Conv-CTC/Attention model, a Self-Attention and a shallow Convolution shared hybrid encoder to replace the original encoder component for hybrid CTC/Attention model [1, 3, 5]. Recurrent neural networks (RNNs) with long short-term memory (LSTM) units have defined the state-of-the-art large-scale speech recognition [1]. The key behind the success of RNNs is their capacity to learn temporal correlations in sequential signals through the recurrent connections when the networks are trained with the back-propagation through time (BPTT) [14] algorithm. However, a well-known weakness of RNNs is the gradient vanishing or explosion problem due to BPTT, and the recurrent connections in RNNs make it challenging to parallelize the computations in both training and inference stages [13]. Although CNNs have a strong ability to extract features, it require multiple layers to capture the correlations between the two features which are very distant in the time space [3]. Self-Attention can model sequences of any length without the problems of the disappearance of gradients in the LSTM; Convolution layer can well correlate related sequence features and improve the robustness. Therefore, we propose a hybrid self-attention/Convolution encoder to extract hidden representations of speech features to improve the hybrid CTC/Attention systems proposed by [3]. In order to take full advantage of the self-attention modeling capability, I also used a language model based on the self-attention mechanism.

2. Model Description

2.1. Hybrid Encoder

We proposed self-Attention/Convolution hybrid encoder in this paper. As shown in Figure 1, in the first downsampling the input sequence, and then followed by position-coding. Inspired by [16, 17], we use 1D convolutional layers to implicitly encode the relative positional information. Convolutional embedding implicitly performs frame stacking as well as learns useful short-range spectral temporal patterns [16, 18]. Convolution has excellent capabilities in feature extraction and has been widely used in object detection [15]. The convolution network performs excellent in translation invariance in the time-frequency domain, making the model more robust (noise resistance), while at the same time, it can learn the local correlations. Self-attention, on the other hand, is expected to capture the global information.

Multi-head attention allows the model to jointly attend to information from different positions. Each head $h_i$ is a complete self-attention component. Here, we use scaled Dot-Product Attention, an effective self-attention mechanism demonstrated in [16]. Let $Q \in \mathbb{R}^{t_q \times d_q}$ be the queries, $K \in \mathbb{R}^{t_k \times d_k}$ be the keys and $V \in \mathbb{R}^{t_v \times d_v}$ be the values, where $t_*$ are the element numbers in different inputs and $d_*$ are the corresponding element dimensions. Normally, $t_k = t_v$, $d_q = d_k$. The outputs of self-attention are computed [6] as:

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

Multi-head attention is a core module of our encoder, which is applied to leverage different attending representations jointly. As the Figure 1 shows, multi-head attention calculates $h$ times Scaled Dot-Product Attention, where $h$ means the head number. The output from different attention heads are then concatenated and projected before being fed into the next layer [19]. It can be expressed as:
MultiHead(H_{mal}) = \text{Concat}(H_1, ..., H_h)W^O \tag{2}

where \( H_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V) \) \tag{3}

\begin{align*}
\text{MidLayer}(H_m) &= \text{LN}(\text{MultiHead}(H_c) + H_c) \tag{4} \\
\text{SelfAttentionLayer}(H_l) &= \text{LN}(\text{FFN}(\text{MidLayer}(H_m))) + \text{MidLayer}(H_m) \tag{5}
\end{align*}

Downsampling is very important and necessary for the Transformer-like networks, which can speed up the training and convergence of the network. We can use dilated convolution to make it.

2.2. **Positional Encoding**

The attention does not use the information of the order of the input sequences. It is possible that reordering the elements, which will get result in the same attention vector after the attention operation, since Eq. (1) is only a weighted sum of the elements [9]. The authors in [8] proposed a sinusoidal function as:
This position encoding method is also used in our paper.

2.3. Hybrid CTC/Attention architecture

Inspired by [3], we adopted a hybrid CTC/Attention architecture and applied an SA-Conv Encoder and self-attention language model base on it, as Figure 2 shows. To address the irregular alignments problem in attention mechanism, the authors in [1, 3] propose the hybrid CTC/Attention architecture. It utilizes a CTC objective function as an auxiliary task to train the encoder network within a multi-task learning (MTL) framework. During training, the forward-backward algorithm of CTC can enforce monotonic alignment between speech and label sequences [3]. The objective to be maximized is a logarithmic linear combination of the CTC and attention objectives as (8):

\[ \mathcal{L}_{MTL} = \lambda \mathcal{L}_{ctc} + (1 - \lambda) \mathcal{L}_{att} = \lambda \log p_{ctc}(y|x) + (1 - \lambda) \log p_{att}(y|x) \]  

\[ \hat{C} = \arg\max_{C \in \mathcal{U}} \{ \lambda \log p_{ctc}(C|X) + (1 - \lambda) \log p_{att}(C|X) \}. \]  

with a tunable parameter \( \lambda : 0 \leq \lambda \leq 1 \).

We have adopted the method of rescoring to combine CTC and Attention during decoding [3], in which the decoder first obtains a set of complete hypotheses using the beam search only with the attention model, and rescores each hypothesis using Eq. (9). \( X \) means the input sequence, \( C \) denotes the prediction sequences, and \( \hat{C} \) refers to the final hypotheses.

\[ \mathcal{L}_{CTC} = \sum_{t=1}^{T} \delta(t, C(t)) \]

\[ \mathcal{L}_{Attention} = \sum_{t=1}^{T} \delta(t, \text{Attention}(t)) \]

\[ \mathcal{L}_{CTC + Attention} = \lambda \mathcal{L}_{CTC} + (1 - \lambda) \mathcal{L}_{Attention} \]

Figure 2. SA-Conv-CTC/Attention architecture.

2.4. Decoder with SA-LM

SA-LM means self-attention language module [20]. We combine a SA-LM network in parallel with the CTC/Attention decoder, which can be trained separately or jointly, where the SA-LM is trained with character sequences without word-level knowledge. Incorporating a character-based language model (LM) into the attention-based models has shown great effectiveness [1, 3, 5]. The language model states purely dependent on the past output label sequence in the decoder to predict the next label.
3. Explore and Experiment

3.1. Datasets and Implementation Details

We demonstrate the effectiveness of proposed SA-Conv-CTC/Attention framework firstly in two different ASR datasets, famous English clean speech corpora Wall Street Journal (WSJ) [3] and HKUST Mandarin Chinese [20]. We fixed the depth of the SA-Conv to be 6 layers, and \( d_k \) in Eq (1) to be 512. The number of hidden units in the feedforward layer is 2048. The kernel size for each convolution layer is 3 without stride. The total number of parameters is around 26M. All the models were trained on 80-dimensional raw log mel-scales filter-banks (FBANKs) [3] which are extracted and normalized per conversation. We also adopted a simple down-sampling method in which we stacked 4 consecutive features vectors to reduce the length of input sequences by a factor of 4, as the memory cost of self-attention is in the order of \( O(T^2) \), where T is the length of the acoustic sequence, lower frame rate would significantly cut down the memory cost, and enable the training of much deeper models. Beside the filter-bank features, we did not employ any auxiliary features. We apply the dropout before residual connection and on the attention weights at 0.1. We also apply label smoothing with \( \varepsilon = 0.1 \). We use RTX2080Ti GPU to train our entire model and infer using the CPU.

3.2. Results

When using WSJ and HKUST to test the model we proposed, [1, 3] are chosen as the baseline, the author proposed hybrid CTC/Attention Architecture with a decoder of LSTM. When we experimented with the Wall Street Journal (WSJ) dataset, training on si284, validating on dev93 and evaluating on eval92 set, we choose the same training sets and test sets as the authors mentioned above about WSJ. For the decoding of the attention and MOL models, we used a conventional beam search algorithm similar to [3] with a beam size of 20 to reduce the computational cost. HKUST Mandarin Chinese conversational telephone speech recognition has 5 hours of recording for evaluation, and we extracted an additional 5 hours from the training data as a development set, and used the rest (167 hours) as a training set, as summarized in Table 2. We used \( \lambda = 0.5 \) for training and decoding instead of 0.1 on the basis of our preliminary investigation [3].

| Table 1. Character error rates (CERs) of baseline for WSJ and HKUST |
|---------------------------------------------------------------|
| Model                                | WSJ0 dev | WSJ1 dev | WSJ0 eval | WSJ1 eval | HKUST dev | HKUST eval |
|--------------------------------------|----------|----------|-----------|-----------|------------|------------|
| Attention                            | 28.02    | 20.06    | 13.68     | 11.08     | 40.3       | 37.8       |
| MTL (\( \lambda = 0.2 \)) + LSTM    | **23.03**| **14.53**| **11.27** | **7.36**  | 38.7       | 36.6       |
| MTL (\( \lambda = 0.5 \)) + LSTM    | 26.28    | 16.24    | 12.00     | 8.31      | –          | –          |
| MTL (\( \lambda = 0.8 \)) + LSTM    | 32.21    | 21.30    | 11.71     | 8.45      | –          | –          |
| MTL + joint decoding (one pass)      | –        | –        | –         | –         | 35.5       | 33.9       |
| MTL + VGG net + joint decoding       | –        | –        | –         | –         | 30.0       | 28.9       |
| MTL + VGG net + RNN-LM+ joint decoding| –        | –        | –         | –         | **29.1**   | **28.0**   |

In Table 1, if the value of \( \lambda \) does not specified, it is given the optimal value 0.1. ‘-’ means the model of the corresponding row has not been tested on the corresponding data set, same as the meaning in Table 2.
Table 2. Character error rates (CERs) of the model we proposed for WSJ and HKUST

| Model | WSJ0 dev | WSJ1 dev | WSJ1 eval | HKUST dev | HKUST eval |
|-------|---------|---------|---------|--------|---------|
| MTL (λ = 0.05) + SA-Conv Encoder | 21.21 | 13.39 | 7.26 | – | – |
| MTL (λ = 0.1) + SA-Conv Encoder | 21.18 | 13.42 | 10.47 | 7.23 | 25.77 | 25.92 |
| MTL (λ = 0.3) + SA-Conv Encoder +SA-LM (separated decoding) | 22.97 | 15.83 | 11.18 | 8.26 | – | – |
| MTL (λ = 0.1) + SA-Conv Encoder +SA-LM (joint decoding) | – | – | – | – | 24.62 | 25.30 |
| MTL (λ = 0.1) + SA-Conv Encoder +SA-LM (joint decoding) | – | – | – | – | 24.35 | 24.93 |

In Table 2, we also use WSJ and HKUST to explore our proposed model. Similarly, we explored the optimal value of λ using the WSJ dataset and explored the effect of the self-attention based language model. We can see that the influence of CTC related to λ is smaller than the baseline, the most suitable value of λ is around 0.1. The effect of the language model is clearly shown in the table, and the effect achieved when performing combination training is the most. Overall, our proposed model performs better than baseline. We achieve a 0.8-4.75% error reduction compared to baseline hybrid CTC/Attention systems on WSJ and HKUST dataset.

4. Conclusion

In this paper, we propose SA-Conv-CTC/Attention model, which apply a Self-Attention and shallow Convolution based hybrid encoder to Hybrid CTC/Attention Architecture, and we also explore the effect of language models based on self-attention mechanisms. Compared to baseline, our network is deeper and the parameters are more complex. Accordingly, the character error rate is lower. In addition to speed the convergence of the network, CTC can also make up for the disadvantage that the attention mechanism cannot achieve temporal alignment. It might be the reason that the position coding is added to our model, so the impact of CTC is smaller than baseline. From the test results, it can be seen that the SA-LM is helpful to reduce character error rates, but which one is more effective than the traditional RNN-LM is not clear. We will study it in the next work.

Acknowledgments

This work has been supported by Project of Hubei Provincial Highway Bureau under Grant Numbers 20141h0288.

References

[1] Kim S, Hori T, Watanabe S. Joint CTC-attention based end-to-end speech recognition using multi-task learning[C]//2017 IEEE international conference on acoustics, speech and signal processing (ICASSP). IEEE, 2017: 4835-4839.
[2] Novoa J, Fredes J, Poblete V, et al. Uncertainty weighting and propagation in DNN–HMM-based speech recognition[J]. Computer Speech & Language, 2018, 47: 30-46.
[3] Watanabe S, Hori T, Kim S, et al. Hybrid CTC/attention architecture for end-to-end speech recognition[J]. IEEE Journal of Selected Topics in Signal Processing, 2017, 11(8): 1240-1253.
[4] Moritz N, Hori T, Le Roux J. Triggered Attention for End-to-end Speech Recognition[C]//ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2019: 5666-5670.
[5] Wu L, Li T, Wang L, et al. Improving Hybrid CTC/Attention Architecture with Time-Restricted Self-Attention CTC for End-to-End Speech Recognition[J]. Applied Sciences, 2019, 9(21): 4639.
[6] Dong L, Xu S, Xu B. Speech-transformer: a no-recurrence sequence-to-sequence model for speech recognition[C]//2018 IEEE International Conference on Acoustics, Speech and Signal
[7] Povey D, Hadjicostis C, Ghahremani P, et al. A time-restricted self-attention layer for asr[C]/2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2018: 5874-5878.

[8] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[C]/Advances in neural information processing systems. 2017: 5998-6008.

[9] Karita S, Chen N, Hayashi T, et al. A comparative study on transformer vs rnn in speech applications[J]. arXiv preprint arXiv:1909.06317, 2019.

[10] Menendez-Pidal X, Chen R. Initialization of ctc speech recognition with standard hmm: U.S. Patent Application 15/645,985[P]. 2019-1-10.

[11] Shan C, Zhang J, Wang Y, et al. Attention-based end-to-end speech recognition on voice search[C]/2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2018: 4764-4768.

[12] Zeyer A, Irie K, Schlüter R, et al. Improved training of end-to-end attention models for speech recognition[J]. arXiv preprint arXiv:1805.03294, 2018.

[13] Lu L. A Transformer with Interleaved Self-attention and Convolution for Hybrid Acoustic Models[J]. arXiv preprint arXiv:1910.10352, 2019.

[14] Werbos P J. Backpropagation through time: what it does and how to do it[J]. Proceedings of the IEEE, 1990, 78(10): 1550-1560.

[15] Nie M, Wang K. Pavement Distress Detection Based on Transfer Learning[C]/2018 5th International Conference on Systems and Informatics (ICSAI). IEEE, 2018: 435-439.

[16] Han K J, Prieto R, Wu K, et al. State-of-the-Art Speech Recognition Using Multi-Stream Self-Attention With Dilated 1D Convolutions[J]. arXiv preprint arXiv:1910.00716, 2019.

[17] Sperber M, Niehues J, Neubig G, et al. Self-attentional acoustic models[J]. arXiv preprint arXiv:1803.09519, 2018.

[18] Pham N Q, Nguyen T S, Niehues J, et al. Very Deep Self-Attention Networks for End-to-End Speech Recognition[J]. arXiv preprint arXiv:1904.13377, 2019.

[19] Wang Y, Mohamed A, Le D, et al. Transformer-based Acoustic Modeling for Hybrid Speech Recognition[J]. arXiv preprint arXiv:1910.09799, 2019.

[20] Dong L, Wang F, Xu B. Self-attention Aligner: A Latency-control End-to-end Model for ASR Using Self-attention Network and Chunk-hopping[C]/ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2019: 5656-5660.