Multi-Stride Self-Attention for Speech Recognition

Kyu J. Han, Jing Huang, Yun Tang, Xiaodong He, Bowen Zhou

Abstract

In contrast to the huge success of self-attention based neural networks in various NLP tasks, the efficacy of self-attention in speech applications is yet limited. This is partly because the full effectiveness of the self-attention mechanism could not be achieved without proper down-sampling schemes in speech tasks. To address this issue, we propose a new self-attention mechanism suitable for speech recognition, namely, multi-stride self-attention. The proposed multi-stride approach lets each group of heads in self-attention process speech frames with a unique stride over neighboring frames. Thus, the entire attention mechanism would not be confined in a fixed frame shift and can have diverse contextual views for a given frame to determine attention weights more effectively. To validate our proposal we evaluated it on various speech corpora for speech recognition, both English and Chinese, and observed a consistent improvement, especially in terms of substitution and deletion errors, without the increase of model complexity. The average WER improvement of 7.5% (relative) obtained by the TDNNs having the multi-stride self-attention layer as compared to the baseline TDNN model shows the effectiveness of the proposed multi-stride self-attention mechanism.

Index Terms: multi-stride self-attention, speech recognition, word error rate (WER)

1. Introduction

Self-attention has been a huge success in a number of downstream natural language processing (NLP) tasks such as machine translation and question answering since it was introduced in [1, 2]. The basic principle for self-attention follows the common concept discussed in neural Turing machines [3] or neural machine translations [4, 5] or memory networks [6]. That is, for a given key or query vector, attention weights are distributed across surrounding or accessible word vectors to filter out irrelevant information while weighing on relevant one. What differentiates self-attention from the normal attention mechanisms is 1) to self-generate key, query and value vectors through learned projections and 2) to process them with multiple heads to offer diverse perspectives in determining attention weights for surrounding context. With this multi-head self-attention approach, both Transformer and BERT [1, 2] were able to achieve the state-of-the-art performances on many NLP tasks without recurrent units in long short-term memory (LSTM) networks, thus with better parallelization in training.

Inspired by the success of self-attention in the field of NLP, there have been a few efforts lately to utilize the idea in speech recognition. In [7], the authors incorporated the multi-head approach to the well-known Listen, Attend and Spell (LAS) [8] framework for end-to-end speech recognition by extending the single-head attention component to the one with multiple heads, and reported a noticeable WER improvement. More direct application of the self-attention mechanism in the original Transformer paper to speech recognition was presented in [9, 10] in the form of Speech-Transformer. In those works, the almost same encoder-decoder structure used in Transformer was applied to end-to-end speech recognition tasks for English and Chinese, having achieved head-to-head with or even better results than LSTM-based sequence-to-sequence systems. A variant of Transformer was also proposed in [11] where a stack of Transformer’s encoder blocks was trained with the Connectionist Temporal Classification (CTC) loss [12], but shown to be still far from the state-of-the-art hybrid DNN/HMM systems trained with the lattice-free maximum mutual information (LF-MMI) objective [13]. In [14, 15], a few tweaks were proposed to address practical issues when applying self-attention to speech recognition. To avoid prohibitively large projection matrices for key, query and value vectors, in [14], down-sampling or reshaping input speech frames via frame concatenation were discussed. In [15], restricting a range of neighboring speech frames with a fixed frame stride to limit the context length and reduce the number of surrounding frames for attention weight computation was introduced.

As addressed in [11, 14, 15], down-sampling speech frames is critical when applying self-attention to speech recognition. It is not only because down-sampling can reduce the entire sequence length to fit computations in memory but also because speech frames with a very short frame shift such as 10ms are highly correlated in a close proximity. Such neighboring frames are hardly considered as a distinct information units such as words or word pieces in NLP, and make the self-attention mechanism hard to effectively distribute attention weights across more informative units like phonemes. To tackle this problem in a different perspective from the previous works, in this paper, we propose a new self-attention mechanism more effective for speech recognition tasks, namely multi-stride self-attention. The proposed multi-stride self-attention mechanism processes speech frames with different frame strides in separate pipelines and combines them in a later stage. Thus, it can have the same benefit of down-sampling input frames to mitigate the burden of computations for self-attention, as well as compute attention weights on a diverse range of neighboring frames. To validate our proposed idea we evaluate it on various speech recognition corpora in both English and Chinese. From the evaluations we observe a consistent improvement across data sets, especially when replacing the last layer of the baseline time-delay neural networks (TDNNs) with the multi-stride self-attention layer, without increasing model complexity.

The paper is organized as follows. In Section 2 we introduce the proposed multi-stride self-attention mechanism, in comparison with the original time-restricted self-attention method in [15]. In Section 3 we share the details of the se-
tups and data used in the experiments discussed throughout the paper, and present experimental results both when stacking the multi-stride self-attention layers and when the proposed layer is combined with the factorized TDNNs [16] on various evaluation corpora. We conclude the paper in Section 4 with a few remarks on our contributions and notes for future directions.

2. Multi-Stride Self-Attention

The time-restricted self-attention mechanism [15] was motivated by [5] in restricting the context used for attention weight computation within a local region. In this paper, we show how effective our proposed multi-stride approach in this time-restricted setup for self-attention would be for various speech recognition tasks.

2.1. Time-restricted self-attention

To formulate the time-restricted self-attention mechanism in a mathematical manner, we adopt the following expressions from [1, 14, 15]. Given the time-restricted setup, we define an input matrix $X \in \mathbb{R}^{s \times d_{\text{model}}}$, where $T$ is the input sequence length restricted by the left and right context ($c_l$ and $c_r$, respectively), and thus $T = c_l + c_r + 1$. $d_{\text{model}}$ is the dimension of embedding vectors in self-attention. Note that the frame stride $f_s$ is applied to down-sample input speech frames, and it was fixed to 3 in [15]. For the projected query, key and value matrices, $Q_i$, $K_i$, and $V_i$, self-attention for the $i^{\text{th}}$ head is computed as follows:

$$
\text{Head}_i = \text{softmax} \left( \frac{Q_i K_i^T}{\sqrt{d_k}} \right) V_i
$$

where $Q_i = X W_i^Q \in \mathbb{R}^{d_{\text{model}} \times d_k}$ and $W_i^Q \in \mathbb{R}^{d_{\text{model}} \times d_k}$, $K_i = X W_i^K \in \mathbb{R}^{d_{\text{model}} \times d_k}$ and $W_i^K \in \mathbb{R}^{d_{\text{model}} \times d_k}$, $V_i = X W_i^V \in \mathbb{R}^{d_k \times d_k}$ and $W_i^V \in \mathbb{R}^{d_k \times d_k}$, where $d_k$ and $d_v$ are the dimensions of query and value vectors in self-attention, respectively. The output of the self-attention layer is given in the following form, after ReLu activation [17] and batch normalization [18] on the concatenated vector from the entire $n_h$ heads in self-attention:

$$
\text{SelfAttention} = \text{BatchNorm} \left( \text{ReLU} \left( \text{MultiHead} \right) \right)
$$

where $\text{MultiHead} = \text{Concat} \left( \text{Head}_1, \ldots, \text{Head}_{n_h} \right)$. The layer structure of the time-restricted self-attention mechanism is illustrated in Figure 1.(a).

2.2. Self-attention block in Transformer

In its implementation, the time-restricted self-attention mechanism is quite different from the self-attention block introduced in the original Transformer paper [1] or used in [11, 14]. As compared to the original self-attention block depicted in Figure 1.(b), it has batch normalization instead of layer normalization [19] and no position-wise feed-forward network after the multi-head self-attention sublayer. The mathematical formation for the self-attention block considering the time-restricted input matrix $X$ can be written as below:

$$
\text{MultiHeadProj} = \text{Concat} \left( \text{Head}_1, \ldots, \text{Head}_{n_h} \right) W_i^O
$$

$$
\text{MidLayer} = \text{LayerNorm} \left( \text{MultiHeadProj} + X \right)
$$

$$
\text{SelfAttention} = \text{LayerNorm} \left( \text{FF} \left( \text{MidLayer} \right) + \text{MidLayer} \right)
$$

where $W_i^O$ is the projection matrix for the concatenated multi-head vectors and FF is the position-wise feed-forward network, i.e., $\text{FF} = \text{Affine} \left( \text{ReLU} \left( \text{Affine} \left( \text{MidLayer} \right) \right) \right)$, with the hidden layer dimension $d_{ff}$. Note that there are skip connections involved for both MidLayer and SelfAttention in Eqs. (4) and (5).

2.3. Multi-stride approach

Down-sampling input speech frames is critical in self-attention for speech applications not only to fitting computations in memory but also to the effectiveness of the self-attention mechanism. Speech frames with very short frame shifts are highly correlated and could cause inefficient attention weight distribution. However it would be challenging to tackle this issue with such rigid approaches as simple down-sampling or frame reshaping in [11, 14, 15]. This requires more flexible consideration of how to down-sample speech frames. In this subsection, we introduce our proposal to address the issue, namely multi-stride self-attention.

The proposed multi-stride self-attention mechanism, illustrated in Figure 1.(c), processes speech frames in the framework of the self-attention block discussed in Section 2.2, in the following way:

1. Accepts speech frames with a different frame stride per group of heads. We use three different frame strides ($n_s = 3$ and $f_s = 1, 3, 5$) with the fixed context lengths ($c_l = c_r = 5$), thus have different context scopes (i.e., 5, 15, 25 for $f_s = 1, 3, 5$, respectively). For each group of heads $s$, a unique input matrix $X^s$ is thus processed.
2. Computes attention weights within each group of heads $s$, pipelining the process up to the layer normalization after the point-wise feed-forward network.

$$Head_i^s = \text{softmax} \left( \frac{Q_i^s K_i^s V_i^s}{\sqrt{d_k}} \right) V_i^s$$  

(6)

$$\text{MultiHeadProj}^s = \text{Concat} \left( Head_1^s, \ldots, Head_n^s \right) W^O$$  

(7)

$$\text{MidLayer}^s = \text{LayerNorm} \left( \text{MultiHeadProj}^s + X^s \right)$$  

(8)

$$\text{Stride}^d = \text{LayerNorm} \left( \text{FF} \left( \text{MidLayer}^s \right) + \text{MidLayer}^s \right)$$  

(9)

3. Concatenates the processed vector from each pipeline and performs the affine transform, ReLU activation, batch normalization and dropout [20] to generate the final output vector with the dimension of $d_{\text{model}}$.

$$\text{MultiStride} = \text{Concat} \left( \text{Stride}^1, \ldots, \text{Stride}^{n_p} \right) W^O$$  

(10)

$$\text{SelfAttention} = \text{Dropout} \left( \text{BatchNorm} \left( \text{ReLU} \left( \text{MultiStride} \right) \right) \right)$$  

(11)

To prevent the increase of model complexity, we keep the total number of heads in self-attention as is, by assigning $n_h/n_s$ heads per pipeline corresponding to a given frame stride as well as reducing the hidden layer dimension of the position-wise feed-forward networks to half. More details on the hyper-parameters in the multi-stride self-attention mechanism will be presented in Section 3 when discussing the experimental setups and results.

### 3. Experimental Results and Discussions

In this section we validate the effectiveness of our proposed multi-stride self-attention mechanism on various speech corpora for speech recognition, both in English and Chinese.

#### 3.1. Experimental setups and data

For the experiments discussed in the paper, we considered various evaluation data sets in TED-LIUM [21] for English broadcast speech in TED talks, HUB5 [22] for English telephone speech, Librispeech [23] for English read speech and AISHELL-2 iOS [24] for Mandarin Chinese read speech on the mobile channel condition. To train each system we followed Kaldi’s example recipe [25] with the corresponding training data up to the speaker adaptive training stages for Gaussian mixture models. Then, we trained different neural network acoustic models having the lattice alignments given by the GMMs as soft targets. For the telephony models, we used the 300hr Switchboard corpus [26] for acoustic modeling and added the Fisher data [27] to strengthen language models. To train language models we used the SRILM toolkit [28]. The first-pass decoding was conducted with the 3-gram LMs and the resultant lattices were rescored with the larger LMs with 4-grams later in the second-pass.

The LF-MMI objective was used to train the neural network acoustic models with the three regularization methods in cross-entropy, $L2$ and leaky HMM [13]. The gradual decrease of learning rates were exploited from $1.0 \times 10^{-4}$ to $1.0 \times 10^{-6}$ to make the entire training procedure stable and have better convergence. The trainings were conducted on either 8 or 16 GPUs on the Nvidia P40 chips, depending upon the size of the training materials. The number of the nodes in the last layer of each model should depend upon the number of tri-phone states in the corresponding HMM per corpus, ranging from 3.5K to 6K after tree clustering.

#### 3.2. Ablation on multi-stride self-attention

We first conducted the ablation tests on the proposed multi-stride self-attention mechanism in comparison with the original time-restricted self-attention method in [15]. Tables 1 and 2 show the experimental comparisons on the TED-LIUM dev set, as we stack the self-attention layers up to 5. For the experiments, we set $n_h = 12, d_q = d_k = 40, d_v = 60, d_{\text{model}} = 256, d_{\text{ff}} = 1, 024, c_0 = c_r = 5, f_s = 3$. For the multi-stride cases, we set $d_{\text{ff}} = 512$ not to increase the model complexity, and $f_s = 1, 3, 5$.

Table 1 presents the comparative results between the original time-restricted self-attention layer and the self-attention block in Transformer [1]. In Section 2.2, we pointed out the differences between them in terms of the normalization method and the position-wise feed-forward network with the skip connections. As for the normalization method, the results show layer normalization would not make a significant improvement over batch normalization, although it gives a marginal WER boost consistently as we stack the self-attention layers. In contrast, the self-attention block in Eq. (5), similar with the one in Transformer’s encoder stack [1], worked well in this time-restricted setup for self-attention. The average improvement of roughly 10% (relative) across stacked self-attention layers seems to come from adding the position-wise feed-forward network and the skip connections. Our proposed multi-stride self-attention mechanism has this self-attention block structure as the base.

Table 2 shows the effectiveness of the proposed multi-stride self-attention mechanism for speech recognition in comparison with the fixed frame stride cases. While the different frame strides would not make any big impact individually, the multi-
Table 3: WER (in %) comparison of the TDNN-F acoustic models equipped with the self-attention layers on various evaluation data sets. TDNN-F with the 10M parameters is the factorized TDNN baseline model with 6 layers. TR-SA: Time-Restricted Self-Attention [15], MS-SA: Multi-Stride Self-Attention.

<table>
<thead>
<tr>
<th>Acoustic Model</th>
<th>TED-LIUM</th>
<th>HUB5</th>
<th>Librispeech</th>
<th>AISHELL-2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DEV</td>
<td>TEST</td>
<td>DEV Clean</td>
<td>TEST Clean</td>
</tr>
<tr>
<td>TDNN-F (10M)</td>
<td>8.2</td>
<td>8.7</td>
<td>10.0</td>
<td>14.9</td>
</tr>
<tr>
<td>+TR-SA (11M)</td>
<td>8.0</td>
<td>8.4</td>
<td>9.5</td>
<td>14.0</td>
</tr>
<tr>
<td>+MS-SA (11M)</td>
<td>7.7</td>
<td>8.0</td>
<td>9.1</td>
<td>13.4</td>
</tr>
</tbody>
</table>

Table 4: Averaged WER (in %) comparison of the time-restricted self-attention layer and the proposed multi-stride self-attention layer on top of the TDNN-F model over the conversational evaluation sets in TED-LIUM and HUB5.

<table>
<thead>
<tr>
<th></th>
<th>Sub.</th>
<th>Del.</th>
<th>Ins.</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>TDNN-F</td>
<td>6.3</td>
<td>3.0</td>
<td>1.4</td>
<td>10.7</td>
</tr>
<tr>
<td>+TR-SA</td>
<td>6.0</td>
<td>2.9</td>
<td>1.3</td>
<td>10.2</td>
</tr>
<tr>
<td>+MS-SA</td>
<td>5.7</td>
<td>2.6</td>
<td>1.3</td>
<td>9.6</td>
</tr>
</tbody>
</table>

Motivated by these promising observations, we plan to extend our research effort to end-to-end speech recognition systems with the proposed multi-stride self-attention mechanism. Incorporating the multi-stride approach to self-attention in Transformer-like encoder-decoder architectures would be a reasonable first step toward that direction.

5. References


