Monotonic Truncated Attention

Haoran Miao, Gaofeng Cheng

February 10, 2020

1 Backgrounds

Online automatic speech recognition (ASR) models transcribe speech to text as the speaker begins speaking. In attention-based encoder-decoder ASR models [1], the widely-used global attention network performs on top of the full sequential input representations, which hampers the online deployment of such ASR models. In [2], we have proposed a stable monotonic chunk-wise attention (sMoChA), which was designed for online attention-based encoder-decoder models. In this blog, we describe the monotonic truncated attention (MTA) network, which is in submission to a journal, to simplify the sMoChA and solve the training-and-decoding mismatch problem [3] of sMoChA. MTA truncates the sequential input representations and computes the attention weights from the truncation end-point to the beginning of input representation sequence. It should be noted that this is not a formal publication and just intends to show how MTA works.

2 Decoding of Monotonic Truncated Attention

In the decoding stage, MTA always begins a search from the previous end-point and decides the next qualified end-point. Suppose the encoder converts the audio frames into sequential input representations $\mathbf{H} = (\mathbf{h}_1, \mathbf{h}_2, \cdots)$ and the decoder generates the sequential hidden state $\mathbf{Q} = (\mathbf{q}_1, \mathbf{q}_2, \cdots)$ at the $(i-1)$-th step. We denote i and j as the indices of $\mathbf{Q}$ and $\mathbf{H}$, respectively, and t_i as the truncation end-point in the input representation sequence for the i-th decoder step. MTA searches t_i as follows, for $j = 1, 2, \cdots$:

$$p_{i,j} = \mathrm{Sigmoid}(g\frac{\mathbf{v}^\top}{||\mathbf{v}||}\tanh(\mathbf{W}_1\mathbf{q}_{i-1} + \mathbf{W}_2\mathbf{h}_j + \mathbf{b}) + r), \tag{1}$$

$$z_{i,j} = \mathbb{I}(p_{i,j} > 0.5 \wedge j \geq t_{i-1}), \tag{2}$$

where matrices $\mathbf{W}_1$, $\mathbf{W}_2$, vectors $\mathbf{b}$, $\mathbf{v}$ and scalar g, r are trainable parameters. In Eq. 1, we define $p_{i,j}$ as the truncation probability. In Eq. 2, we define $z_{i,j}$ as the indicator of truncating or not-truncating at $\mathbf{h_j}$, and $\mathbb{I}$ is an indicator function. When the truncation probability $p_{i,j}$ is larger than the predefined threshold (i.e. 0.5) and the index j is greater or equal to the previous end-point t_{i-1}, we set $t_i = j$. Then, MTA computes a label-wise representation $\mathbf{r}_i$ as follows, for $j = 1, 2, \cdots, t_i$:

$$w_{i,j} = p_{i,j}\prod_{k=1}^{j-1}(1 - p_{i,k}), \tag{3}$$

$$\mathbf{r}_i = \sum_{j=1}^{t_i} w_{i,j}\mathbf{h}_j. \tag{4}$$

In Eq. 3, we denote $w_{i,j}$ as the attention weights. In Eq. 4, $\mathbf{r}_i$ is the weighted sum of the input representations before the truncation end-point. This label-wise representation $\mathbf{r}_i$ are passed into the decoder to predict the i-th label.

3 Training of Monotonic Truncated Attention

In the training stage, the label-wise representation $\mathbf{r}_i$ is computed based on the full sequential input representations as follows:

$$\mathbf{r}_i = \sum_{j=1}^{T} w_{i,j}\mathbf{h}_j, \tag{5}$$

where T denotes the length of input representation sequence. In Eq 3, the computation of $w_{i,j}$ can be rewritten in the following recursive form:

$$w_{i,j} = \frac{p_{i,j}}{p_{i,j-1}} \cdot (1 - p_{i,j-1}) \cdot w_{i,j}. \tag{6}$$

In Eq. 6, $w_{i,j}$ exponentially decays by $(1 - p_{i,j-1})$ along the index j. To prevent $w_{i,j}$ from vanishing, we enforce the mean of $(1 - p_{i,j-1})$ to be close to 1 by initializing r in Eq. 1 to a negative value, e.g. $r = -4$.

4 Experiments

We evaluated different attention networks on HKUST Mandarin conversational telephone (HKUST) [4] corpus. We used ESPnet [5] toolkit and followed the configurations on [6] to build the hybrid CTC/attention end-to-end models with recurrent neural networks [7]. We open source codes of sMoChA on [8]. The results are listed in Table 1. It shows that MTA outperformed sMoChA and exhibited comparable performance with location-aware attention network.

Table 1: Character error rates (CERs) of different attention types on HKUST.

Encoder Type	Attention Type	Chunk Width	CER (%)	
			dev	eval
BLSTM	Location-aware Attention	–	28.7	27.6
BLSTM	Stable Monotonic Chunk-wise Attention (sMoChA)	1	28.8	27.8
		3	28.6	27.8
		6	28.9	27.7
BLSTM	Monotonic Truncated Attention (MTA)	–	28.5	27.6
LC-BLSTM			29.4	27.8

References

[1] C. Chiu, T. N. Sainath, Y. Wu, R. Prabhavalkar, P. Nguyen, Z. Chen, A. Kannan, R. J. Weiss, K. Rao, E. Gonina, N. Jaitly, B. Li, J. Chorowski, and M. Bacchiani. State-of-the-art speech recognition with sequence-to-sequence models. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4774–4778, 2018.

[2] H. Miao, G. Cheng, P. Zhang, L. Ta, and Y. Yan. Online Hybrid CTC/Attention Architecture for End-to-End Speech Recognition. In *Proc. Interspeech 2019*, pages 2623–2627, 2019.

[3] M. He, Y. Deng, and L. He. Robust Sequence-to-Sequence Acoustic Modeling with Stepwise Monotonic Attention for Neural TTS. In *Proc. Interspeech 2019*, pages 1293–1297, 2019.

[4] Y. Liu, P. Fung, Y. Yang, C. Cieri, S. Huang, and D. Graff. Hkust/mts: A very large scale mandarin telephone speech corpus. In *International Symposium on Chinese Spoken Language Processing*, pages 724–735. Springer, 2006.

[5] S. Watanabe, T. Hori, S. Karita, T. Hayashi, J. Nishitoba, Y. Unno, N. E. Y. Soplin, J. Heymann, M. Wiesner, N. Chen, A. Renduchintala, and T. Ochiai. Espnet: End-to-end speech processing toolkit. In *Proc. Interspeech 2018*, pages 2207–2211, 2018.

[6] ESPnet. End-to-end speech processing toolkit. `https://github.com/espnet/espnet/tree/v.0.2.0/egs/hkust`, 2018. GitHub repository.

[7] S. Watanabe, T. Hori, S. Kim, J. R. Hershey, and T. Hayashi. Hybrid ctc/attention architecture for end-to-end speech recognition. *IEEE Journal of Selected Topics in Signal Processing*, 11(8):1240–1253, 2017.

[8] H. Miao. Streaming attention. `https://github.com/HaoranMiao/streaming-attention`, 2019. GitHub repository.