HYBRID CTC/ATTENTION ARCHITECTURE FOR END-TO-END SPEECH RECOGNITION

JINNAN LI

TTIC 31110

CONTENT

I. Introduction and Motivation

II. *Connectionist Temporal Classification*

III. Improving LSTM-CTC based ASR

IV. *AttentionMechanism*

V. HYBRID CTC/ATTENTION

VI. Experiements and Results

INTRODUCTION AND MOTIVATION

- **There are two major types of end- to-end architectures for ASR; attention-based methods use an attention mechanism 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.**

- The basic attention-based mechanism has an advantage that it does not require any conditional independence assumptions. However, it is too flexible in the sense that it allows extremely non-sequential alignments.

- The paper proposed a mechanism to take advantage of the constrained CTC alignment in a hybrid CTC/attention-based system.

CONNECTIONIST TEMPORAL CLASSIFICATION

- Follows from Bayes decision theory.
- Define an an augmented letter sequence, C'

$$C' = \{<b>,c1,<b>,c2,<b>,...,cL,<b>\}$$

$$=\{c'l \in UU\{<b>\} | l=1,...,2L+1\}$$

and framewise letter sequence with an additional blank symbol,

$$Z = \{zt \in U \cup \{< b >\} | t = 1, \ldots, T\}.$$

- We can solve the problem by finding argmax C in C' of the objective

$$p(C|Z) = \frac{p(Z|C)p(C)}{p(Z)}$$
$$= \prod_{t=1}^{T} p(z_t|z_1,\ldots,z_{t-1},C)\frac{p(C)}{p(Z)}$$
$$\approx \prod_{t=1}^{T} p(z_t|z_{t-1},C)\frac{p(C)}{p(Z)},$$

$$p(Z|X) = \prod_{t=1}^{T} p(z_t|z_1,\ldots,z_{t-1},X)$$
$$\approx \prod_{t=1}^{T} p(z_t|X).$$

$$p(C|X) = \sum_Z p(C|Z,X)p(Z|X)$$
$$\approx \sum_Z p(C|Z)p(Z|X).$$

$$p(C|X) \approx \underbrace{\sum_Z \prod_{t=1}^{T} p(z_t|z_{t-1},C)p(z_t|X)}_{\triangleq p_{ctc}(C|X)} \frac{p(C)}{p(Z)}$$

IMPROVING LSTM-CTC BASED ASR

- Long Short Term Memory (LSTM), and in general, recurrent neural network (RNN) based ASR systems trained with connectionist temporal classification (CTC) when data is abundant.
- We want to improve the performance of this mechanism in domains with limited training data.

IMPROVING LSTM-CTC BASED ASR

Several improving schemes:

- Instead of initializing weights and biases to random values in the interval [-0.1,0.1], we initialize the forget gate bias, bf , to a large value, say 1, that will force the gate to be initialized in an open position and allow memory cell gradients in time to flow more readily.
- Data Augmentation
- Stacking and striding frames
- Dropout on feedforward connections

IMPROVING LSTM-CTC BASED ASR

Librispeech	1h CER (clean-dev)	1h valid WER (clean-dev)	10h valid CER (clean-dev)	10h valid WER (clean-dev)
Wav2vec+CTC (baseline)	11.31	27.05	8.5	22.71
Wav2vec+DNN+CTC	100	100	83.95	100
Wav2vec+LSTM+CTC (random initialization)	100	100	99.51	100
Wav2vec+LSTM+CTC (b_f=1)	100	100	100	100
Wav2vec+LSTM+CTC (b_f=1, minimum 6 epochs)	100	100	99.72	100
Wav2vec+LSTM+CTC (b_f=1, minimum 8 epochs)	100	100	100	100

ATTENTION MECHANISM

- Compared with the CTC approach, *the attention-based approach does not make any conditional independence assumptions*, and directly estimates the posterior, p(C|X), on the basis of a probabilistic chain rule, as follows:

$$p(C|X) = \underbrace{\prod_{l=1}^{L} p(c_l|c_1, \ldots, c_{l-1}, X)}_{\triangleq p_{\text{att}}(C|X)}, \tag{24}$$

where $p_{\text{att}}(C|X)$ is an attention-based objective function. $p(c_l|c_1, \ldots, c_{l-1}, X)$ is obtained by

$$\mathbf{h}_t = \text{Encoder}(X), \tag{25}$$

$$a_{lt} = \begin{cases} \text{ContentAttention}(\mathbf{q}_{l-1}, \mathbf{h}_t) \\ \text{LocationAttention}(\{a_{l-1}\}_{t=1}^{T}, \mathbf{q}_{l-1}, \mathbf{h}_t) \end{cases}, \tag{26}$$

$$\mathbf{r}_l = \sum_{t=1}^{T} a_{lt} \mathbf{h}_t, \tag{27}$$

$$p(c_l|c_1, \ldots, c_{l-1}, X) = \text{Decoder}(\mathbf{r}_l, \mathbf{q}_{l-1}, c_{l-1}). \tag{28}$$

HYBRID CTC/ATTENTION

- *Multiobjective Learning*
- uses a CTC objective function as an auxiliary task to train the attention model encoder within the multiobjective learning (MOL) framework.
- the forward–backward algorithm of CTC can enforce a monotonic alignment between speech and label sequences during training.
- Speed up the alighment process

$$\mathcal{L}_{\text{MOL}} = \lambda \log p_{\text{ctc}}(C|X) + (1-\lambda) \log p^{*}_{\text{att}}(C|X), \qquad (38)$$

HYBRID CTC/ATTENTION

- Joint decoding

to find the most probable letter sequence $\hat{C}$ given the speech input X, i.e.

$$\hat{C} = \arg\max_{C \in \mathcal{U}^*} \log p(C|X). \quad (39)$$

In attention-based ASR, $p(C|X)$ is computed by (24), and $\hat{C}$ is found by a beam search technique.

HYBRID CTC/ATTENTION

- Let Ω_l be a set of partial hypotheses of the length l.
- For each step, we expand each partial hypothese in Ω_l-1 by adding one more letter and adding into Ω_l. Find each hypothese h in Ω_l, we compute the score
- h = g + c

$$\alpha(h, X) = \alpha(g, X) + \log p(c|g_{l-1}, X),$$

- If c is a special symbol that represents the end of a sequence, <eos>, h is added to Ωˆ but not Ω_l , where Ωˆ denotes a set of complete hypotheses. Finally, Cˆ is obtained by

$$\hat{C} = \arg\max_{h \in \hat{\Omega}} \alpha(h, X).$$

HYBRID CTC/ATTENTION

- Attention-based ASR may be prone to include deletion and insertion errors because of its flexible alignment property;
- it may prematurely predict the end-of-sequence label, even when it has not attended to all of the encoder frames, making the hypothesis too short
- it may predict the next label with a high probability by attending to the same portions as those attended to before. This case the hypothesis becomes very long and includes repetitions of the same label sequence.

HYBRID CTC/ATTENTION

- A conventional way to solve those problems is to add a length penalty.

$$\hat{C} = \arg\max_{C \in \mathcal{U}^*} \{\log p(C|X) + \gamma|C| + \eta \cdot \text{coverage}(C|X)\},$$

- In our case, we will add the attention objective as a penalty.

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

- The CTC probability enforces a monotonic alignment that does not allow large jumps or looping of the same. Furthermore, it can avoid premature prediction of the end-of-sequence label, which is not handled by the coverage term..

HYBRID CTC/ATTENTION

- it is nontrivial to combine the CTC and attention-based scores in the beam search, because the attention decoder performs it output-label-synchronously while CTC perfomrs it frame- synchronously.
- The first method is a two-pass approach, in which the first pass obtains a set of complete hypotheses using the beam search, where only the attention-based sequence probabilities are considered. The second pass rescores the complete hypotheses using the CTC and attention probabilities.

$$\hat{C} = \arg\max_{h \in \hat{\Omega}} \{\lambda \alpha_{\text{ctc}}(h, X) + (1 - \lambda)\alpha_{\text{att}}(h, X)\}, \quad (46)$$

where

$$\begin{cases} \alpha_{\text{ctc}}(h, X) \triangleq \log p_{\text{ctc}}(h|X) \\ \alpha_{\text{att}}(h, X) \triangleq \log p_{\text{att}}(h|X) \end{cases} \quad (47)$$

HYBRID CTC/ATTENTION

- The second method is one-pass de- coding, in which we compute the probability of each partial hypothesis using CTC and an attention model.
- Here, we uti- lize the CTC prefix probability defined as the cumulative probability of all label sequences that have h as their prefix:

$$p_{\mathrm{ctc}}(h, \ldots | X) = \sum_{\nu \in (\mathcal{U} \cup \{\texttt{<eos>}\})^+} p_{\mathrm{ctc}}(h \cdot \nu | X),$$

and we define the CTC score as

$$\alpha_{\mathrm{ctc}}(h, X) \triangleq \log p_{\mathrm{ctc}}(h, \ldots | X),$$

- where ν represents all possible label sequences except the empty string.

HYBRID CTC/ATTENTION

$$\gamma_t^{(n)}(\texttt{<sos>}) = 0,$$

$$\gamma_t^{(b)}(\texttt{<sos>}) = \prod_{\tau=1}^{t} \gamma_{\tau-1}^{(b)}(\texttt{<sos>}) \cdot p(z_\tau = \texttt{<b>} \,|X),$$

Algorithm 1: Joint CTC/attention One-pass Decoding.

procedure ONEPASSBEAMSEARCH ($X, L_{\max}$)
  $\Omega_0 \leftarrow \{\texttt{<sos>}\}$
  $\hat{\Omega} \leftarrow \emptyset$
  for $l = 1 \ldots L_{\max}$ **do**
    $\Omega_l \leftarrow \emptyset$
    while $\Omega_{l-1} \neq \emptyset$ **do**
      $g \leftarrow \text{HEAD}(\Omega_{l-1})$
      DEQUEUE(Ω_{l-1})
      for each $c \in \mathcal{U} \cup \{\texttt{<eos>}\}$ **do**
        $h \leftarrow g \cdot c$
        $\alpha(h) \leftarrow \lambda\alpha_{\text{ctc}}(h, X) + (1-\lambda)\alpha_{\text{att}}(h, X)$
        if $c = \texttt{<eos>}$ **then**
          ENQUEUE($\hat{\Omega}, h$)
        else
          ENQUEUE(Ω_l, h)
          if $|\Omega_l| > beamWidth$ **then**
            REMOVEWORST(Ω_l)
          end if
        end if
      end for
    end while
    if ENDDETECT($\hat{\Omega}, l$) = `true` **then**
      break ▷ exit for loop
    end if
  end for
  return $\arg\max_{C \in \hat{\Omega}} \alpha(C)$
end procedure

Algorithm 2: CTC Label Sequence Score.

function α_{ctc} h, X
  $g, c \leftarrow h$ ▷ split h into the last label c and the rest g
  if $c = \texttt{<eos>}$ **then**
    return $\log\{\gamma_T^{(n)}(g) + \gamma_T^{(b)}(g)\}$
  else
    $\gamma_1^{(n)}(h) \leftarrow \begin{cases} p(z_1 = c|X) & \text{if } g = \texttt{<sos>} \\ 0 & \text{otherwise} \end{cases}$
    $\gamma_1^{(b)}(h) \leftarrow 0$
    $\Psi \leftarrow \gamma_1^{(n)}(h)$
    for $t = 2 \ldots T$ **do**
      $\Phi \leftarrow \gamma_{t-1}^{(b)}(g) + \begin{cases} 0 & \text{if last}(g) = c \\ \gamma_{t-1}^{(n)}(g) & \text{otherwise} \end{cases}$
      $\gamma_t^{(n)}(h) \leftarrow \left(\gamma_{t-1}^{(n)}(h) + \Phi\right) p(z_t = c|X)$
      $\gamma_t^{(b)}(h) \leftarrow \left(\gamma_{t-1}^{(b)}(h) + \gamma_{t-1}^{(n)}(h)\right) p(z_t = \texttt{<b>}\,|X)$
      $\Psi \leftarrow \Psi + \Phi \cdot p(z_t = c|X)$
    end for
    return $\log(\Psi)$
  end if
end function

RESULT

Librispeech	10hr CER (dev-clean)	10hr WER (dev-clean)
Wav2vec+cnn+seq2seq	117.3	162.4
MOL (lambda=0.2)	96.07	100
MOL (lambda=0.5)	84.26	100
MOL (lambda=0.9)	96.31	100

CITATION

- 1. *A pytorch speech toolkit*. SpeechBrain. (n.d.). Retrieved June 2, 2022, from https://speechbrain.github.io/
- 2. Miao, H., Cheng, G., Zhang, P., Li, T., & Yan, Y. (2019). Online hybrid CTC/attention architecture for end-to-end speech recognition. *Interspeech 2019*. https://doi.org/10.21437/interspeech.2019-2018
- 3. Moriya, T., Ochiai, T., Karita, S., Sato, H., Tanaka, T., Ashihara, T., Masumura, R., Shinohara, Y., & Delcroix, M. (2020). Self-distillation for improving CTC-transformer-based ASR Systems. *Interspeech 2020*. https://doi.org/10.21437/interspeech.2020-1223