Neural Transducer for Speech Recognition

Zhengkun Tian
Institute of Automation, Chinese Academy of Sciences
Intelligent Interaction Team
20-Jan-19

Contents

Connectionist Temporal Classification

CTC

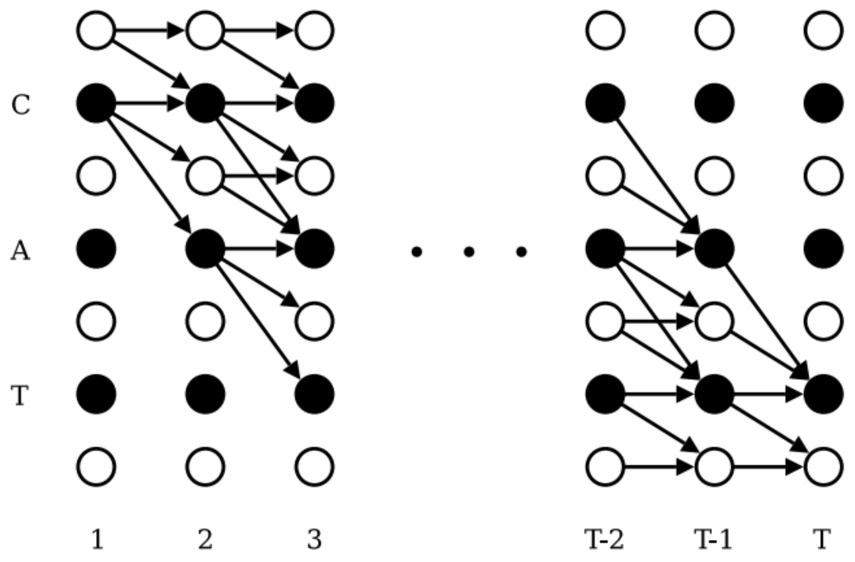
Motivation

◎ Traditional hybrid models need forced alignment to provide frame-level label information.

◎ Hybrid systems do not exploit the full potential of RNNs for sequence modelling.

Proposed Methods

The basic idea is to interpret the network outputs as a probability distribution over all possible label sequences, conditioned on a given input sequence.



Example:

Input Sequence: 100 frames
Target Sequence: hello

Traditional Hybrid Model:
Label: h h h h e e e e e e e e l l l l l l l l l l l l o o o o

CTC:
Possible Sequence: _ _ h _ e e _ _l l l _ l l _ _ o_ _
_ h h_ _ e _ _ _ l _ _ l l l _ o_ _
.....

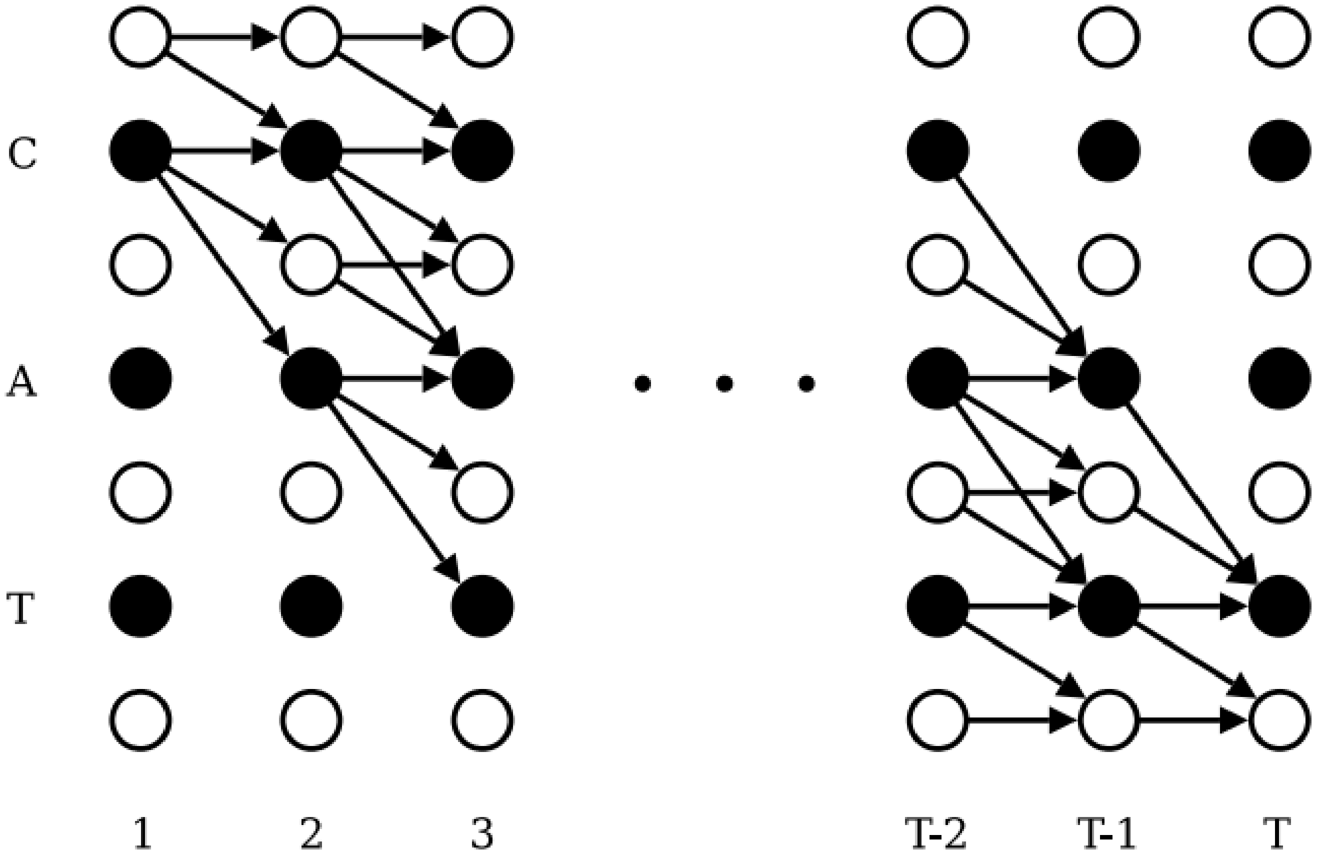
Forward-Backward Algorithm

$$\alpha(t,u) = \sum_{\pi \in V(t,u)} \prod_{i=1}^{t} y^i_{\pi_i}$$

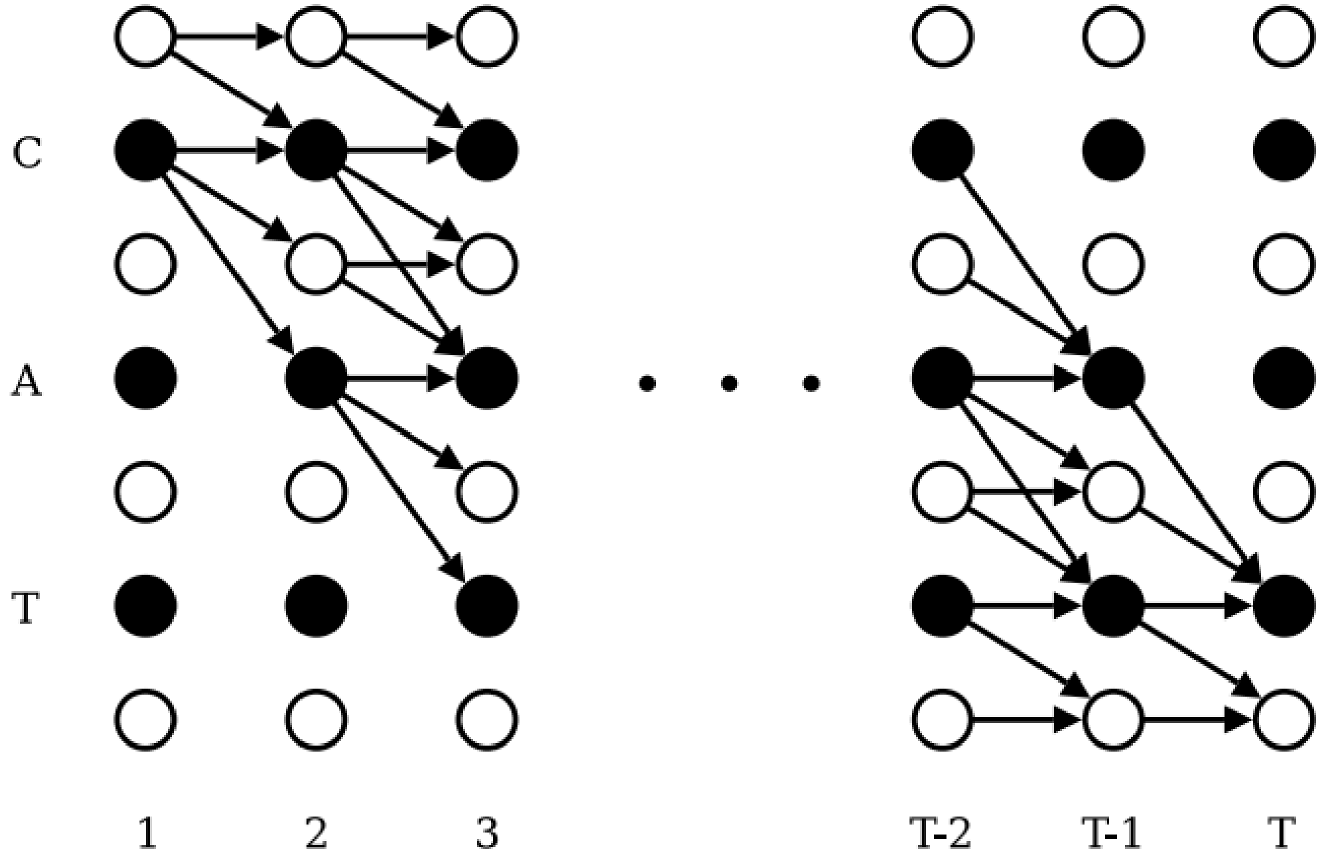
$$\alpha(t,u) = y^t_{l'_u} \sum_{i=f(u)}^{u} \alpha(t-1,i)$$

$$p(\mathbf{l}|\mathbf{x}) = \alpha(T,U') + \alpha(T,U'-1)$$

$$f(u) = \begin{cases} u-1 & \text{if } l'_u = \text{blank or } l'_{u-2} = l'_u \\ u-2 & \text{otherwise} \end{cases}$$



Forward-Backward Algorithm

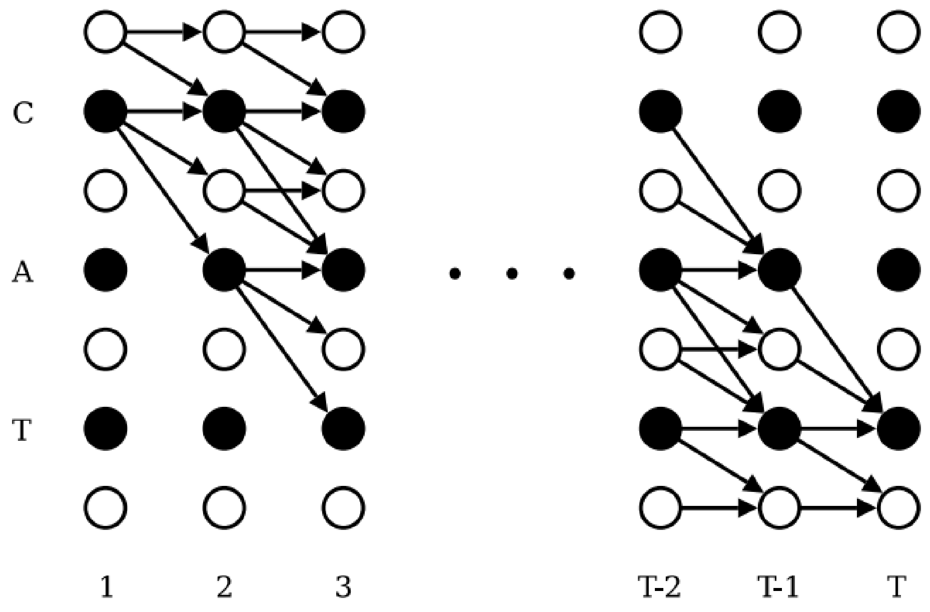


$$\beta(t,u) = \sum_{\pi \in W(t,u)} \prod_{i=1}^{T-t} y_{\pi_i}^{t+i}$$

$$\beta(t,u) = \sum_{i=u}^{g(u)} \beta(t+1,i) y_{l'_i}^{t+1}$$

$$g(u) = \begin{cases} u+1 & \text{if } l'_u = \text{blank or } l'_{u+2} = l'_u \\ u+2 & \text{otherwise} \end{cases}$$

Loss Function

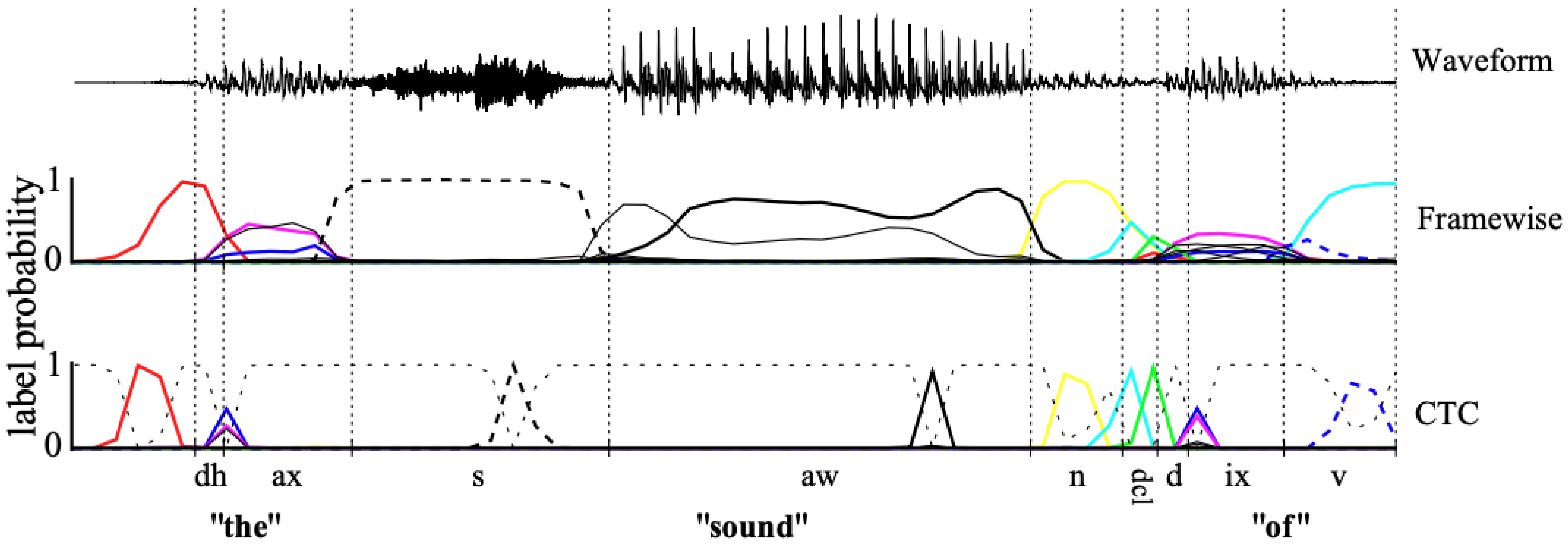


$$p(\mathbf{z}|\mathbf{x}) = \sum_{u=1}^{|\mathbf{z}'|} \alpha(t,u)\beta(t,u)$$

$$\mathcal{L}(S) = -\ln \prod_{(\mathbf{x},\mathbf{z})\in S} p(\mathbf{z}|\mathbf{x}) = -\sum_{(\mathbf{x},\mathbf{z})\in S} \ln p(\mathbf{z}|\mathbf{x})$$

$$\frac{\partial \mathcal{L}(\mathbf{x},\mathbf{z})}{\partial y_k^t} = -\frac{\partial \ln p(\mathbf{z}|\mathbf{x})}{\partial y_k^t} = -\frac{1}{p(\mathbf{z}|\mathbf{x})}\frac{\partial p(\mathbf{z}|\mathbf{x})}{\partial y_k^t}$$

Comparison



Waveform
Framewise
CTC
label probability
1
0
1
0
dh
ax
s
aw
n
dcl
d
ix
v
"the"
"sound"
"of"

Advantages

◎ CTC does not require alignment information.

◎ Thanks to the existence of a large number of spaces, the decoding speed of the CTC model is greatly improved.

Neural Transducer

Why We Need Neural Transducer？

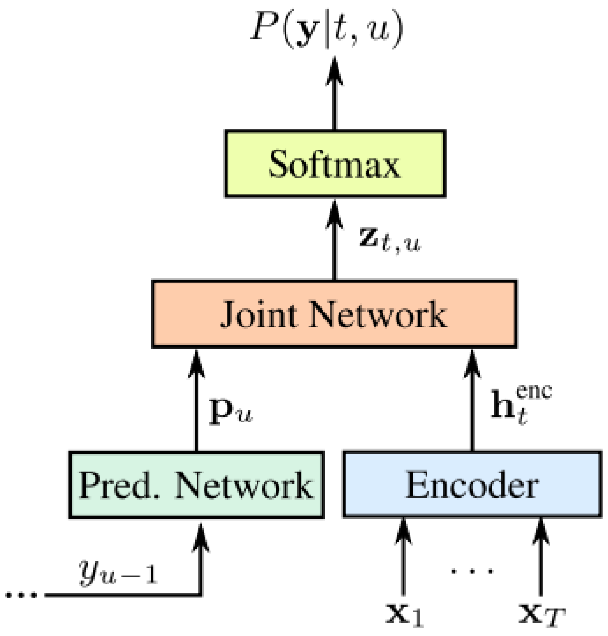
Motivation

◎ CTC does not model the interdependencies between the outputs.

◎ CTC cannot perform end-to-end joint optimization with language models.

◎ CTC requires that the output sequence is not longer than input sequence.

RNN-Transducer



(b.) RNN-Transducer

RNN-Transducer has three parts.

(1) Transcription Net (Encoder) is similar to an acoustic model in a traditional ASR systems.

(2) Prediction Net (Decoder) can be regard as a language model.

(3) Joint Net can combine the encoder outputs and the decoder outputs to compute output logits.

Joint Net

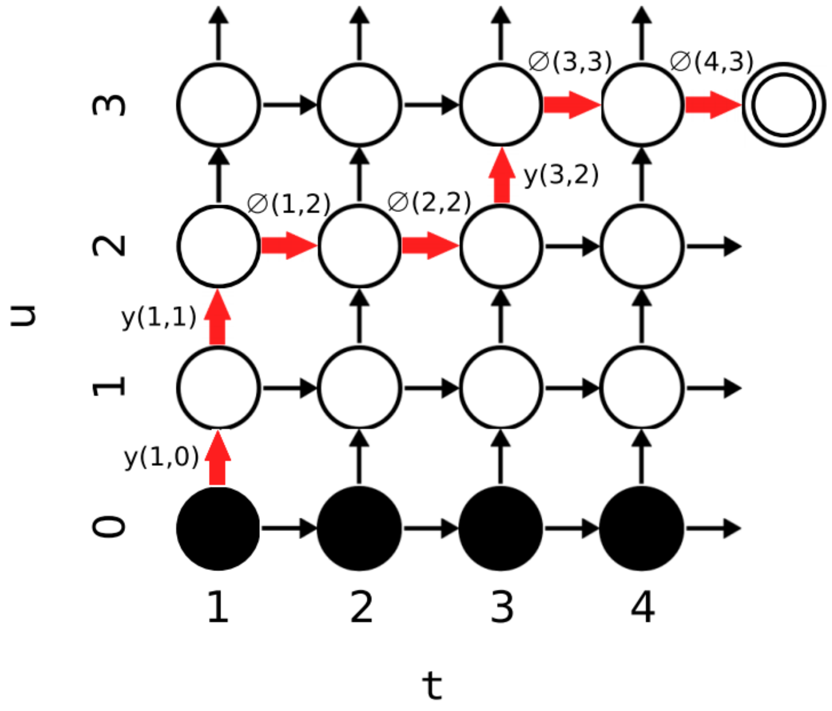
$$h(k,t,u) = \exp\left(f_t^k + g_u^k\right)$$

The Decoder Output

The Encoder Output

$$\Pr(k \in \bar{\mathcal{Y}} | t,u) = \frac{h(k,t,u)}{\sum_{k' \in \bar{\mathcal{Y}}} h(k',t,u)}$$

Output Probability Lattice



Forward Algorithm

$$\alpha(t,u) = \alpha(t-1,u)\varnothing(t-1,u) + \alpha(t,u-1)y(t,u-1)$$

$$\Pr(\boldsymbol{y}|\boldsymbol{x}) = \alpha(T,U)\varnothing(T,U)$$

Backward Algorithm

$$\beta(t,u) = \beta(t+1,u)\varnothing(t,u) + \beta(t,u+1)y(t,u)$$

$$\beta(T,U) = \varnothing(T,U)$$

Loss Function

$$\Pr(\boldsymbol{y}^*|\boldsymbol{x}) = \sum_{(t,u):t+u=n} \alpha(t,u)\beta(t,u)$$

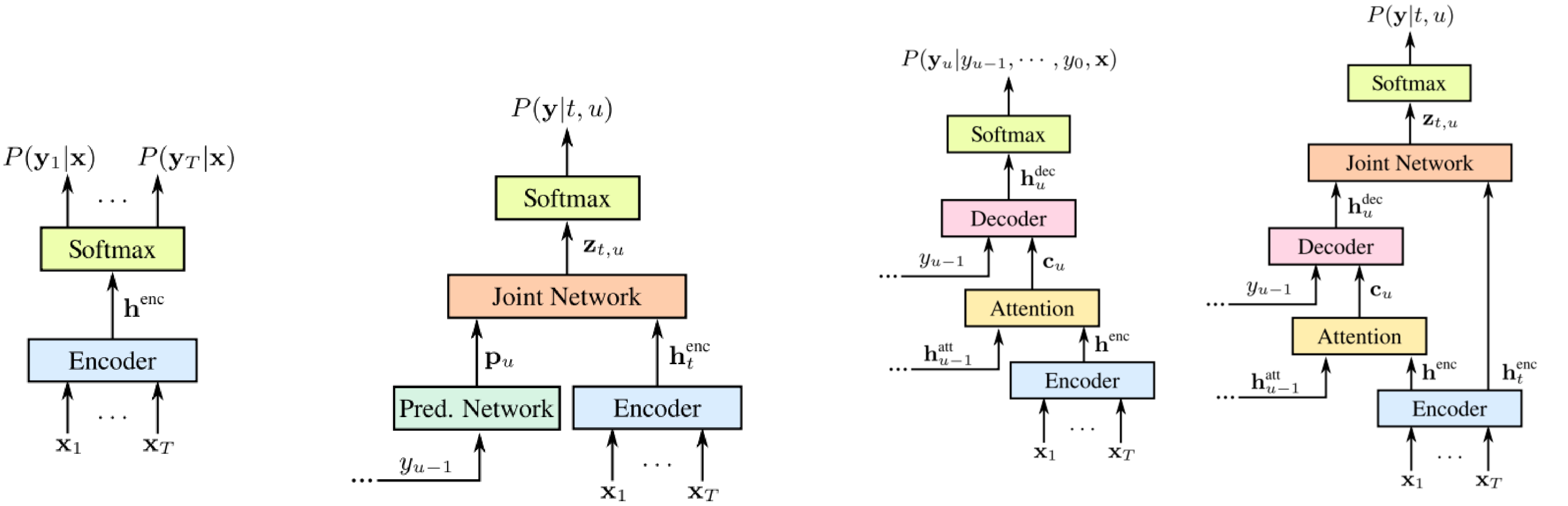
$$\mathcal{L} = -\ln \Pr(\boldsymbol{y}^*|\boldsymbol{x})$$

$$\frac{\partial \mathcal{L}}{\partial f_t^k} = \sum_{u=0}^{U} \sum_{k' \in \bar{\mathcal{Y}}} \frac{\partial \mathcal{L}}{\partial \Pr(k'|t,u)} \frac{\partial \Pr(k'|t,u)}{\partial f_t^k}$$

$$\frac{\partial \mathcal{L}}{\partial g_u^k} = \sum_{t=1}^{T} \sum_{k' \in \bar{\mathcal{Y}}} \frac{\partial \mathcal{L}}{\partial \Pr(k'|t,u)} \frac{\partial \Pr(k'|t,u)}{\partial g_u^k}$$

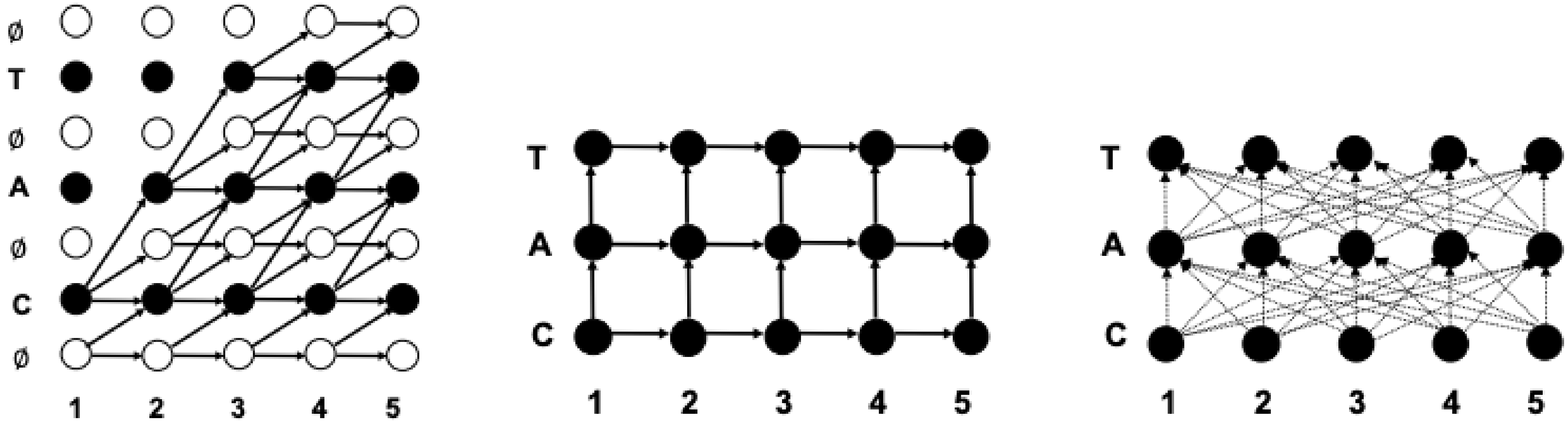
$$\frac{\partial \mathcal{L}}{\partial \Pr(k|t,u)} = -\frac{\alpha(t,u)}{\Pr(\boldsymbol{y}^*|\boldsymbol{x})} \begin{cases} \beta(t,u+1) \text{ if } k = y_{u+1} \\ \beta(t+1,u) \text{ if } k = \varnothing \\ 0 \text{ otherwise} \end{cases}$$

Comparisons



(a.) CTC (b.) RNN-Transducer (c.) Attention-based Model (d.) RNN-Transducer with Attention

Comparisons



(a) CTC (b) RNN-Transducer (c) Attention

[1] Exploring Neural Transducers for End-to-End Speech Recognition

Comparisons

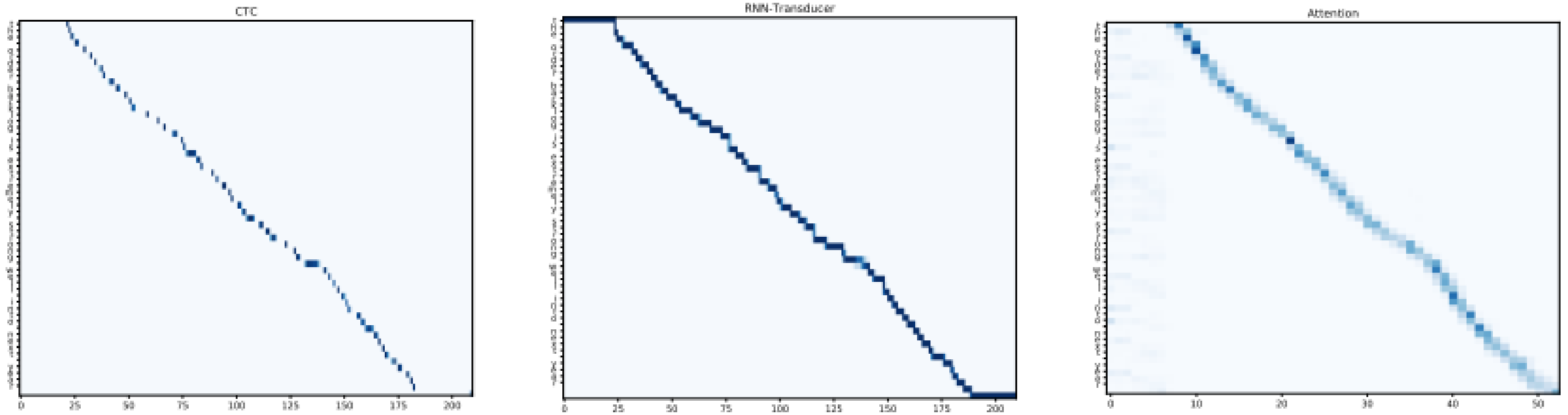


Fig. 3. Visualization of learned alignments for the same utterance using CTC (left), RNN-Transducer (middle), and Attention (right). The alignments are between ground-truth text (y-axis) and audio features fed into the decoder(x-axis). Note that Attention does two more time-scale downsampling, which results in $4\times$ shorter sequences (x axis) compared to the other two.

Advantages

◎ No conditional independence assumption between the predictions at each output step

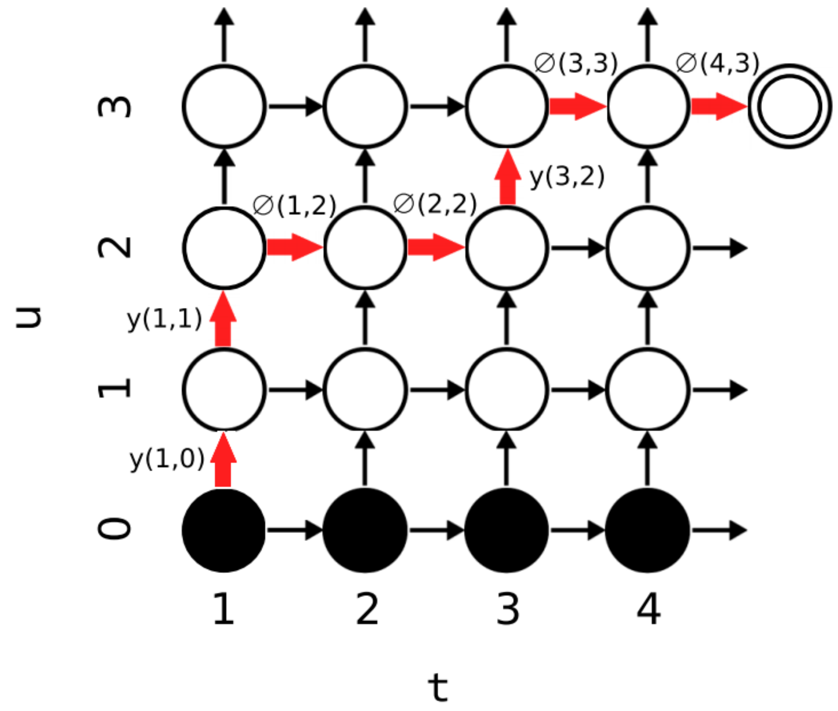
◎ Integrated language model

◎ End-to-end joint optimization

◎ Online decoding capability

Improved Neural Transducer

Recurrent Neural Aligner

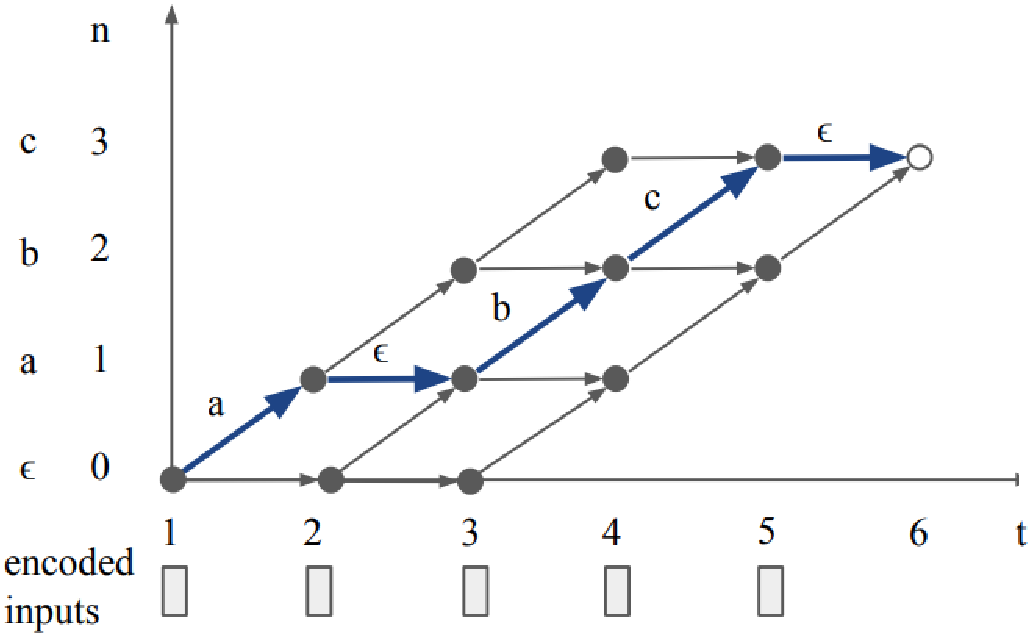


Disadvantages of Neural Transducer

◎ High Computational Costs

◎ There are some unreasonable paths.

Recurrent Neural Aligner

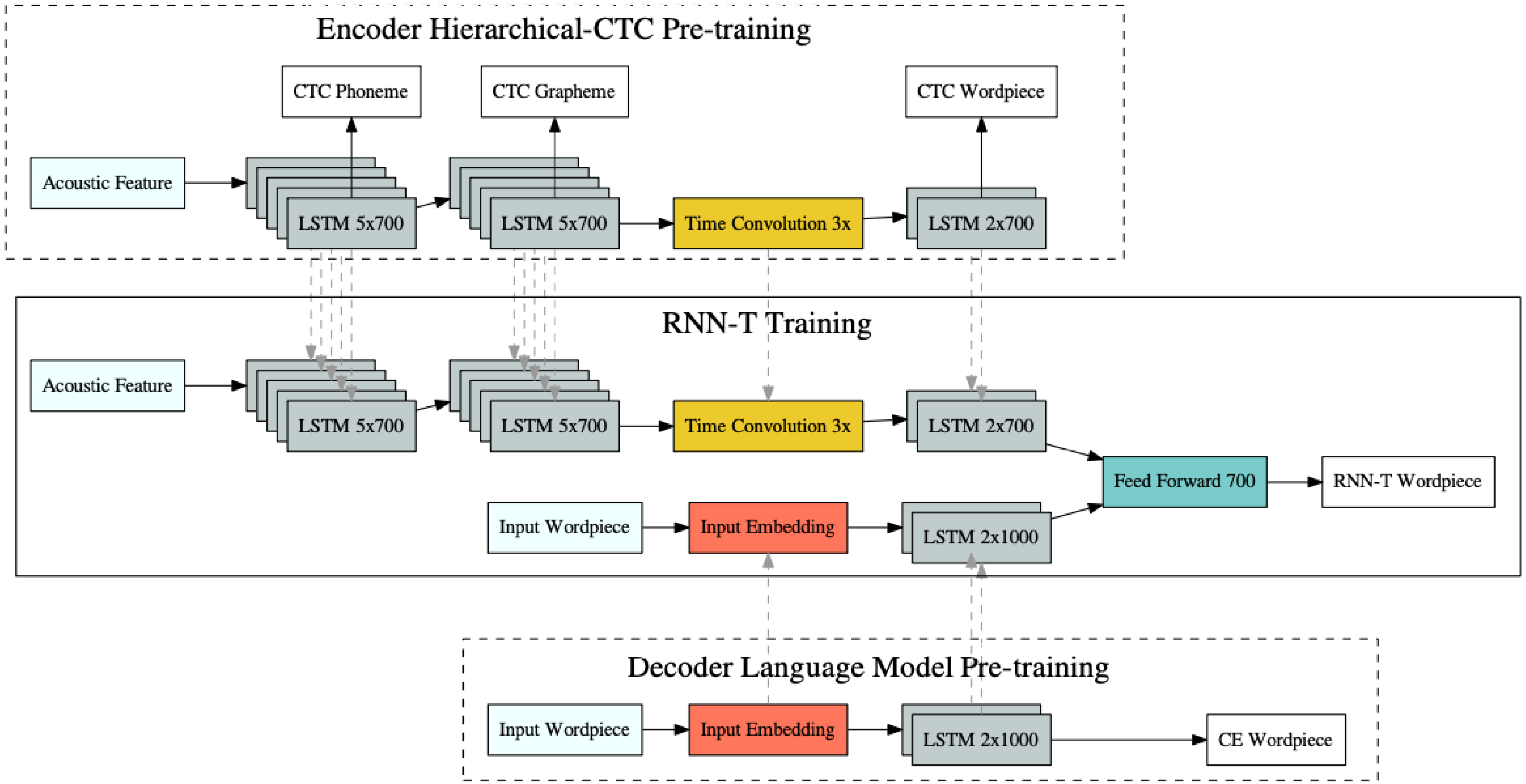


$$\alpha(t,n) = \alpha(t-1,n-1)p(y_n|t-1,n-1) \\ + \alpha(t-1,n)p(\epsilon|t-1,n)$$

$$\beta(t,n) = \beta(t+1,n+1)p(y_{n+1}|t,n) \\ + \beta(t+1,n)p(\epsilon|t,n)$$

Expected Loss $$L = \sum_{\mathbf{z}} P(\mathbf{z}|\mathbf{x})\mathrm{loss}(\mathbf{x},\mathbf{z},\mathbf{y})$$

Multi-stages of Training a Wordpiece RNN-T



[1] Exploring Architectures, Data and Units For Streaming End-to-End Speech Recognition with RNN-Transducer

Take home messages

◎ Neural Transducer’s performance is better than CTC, but slightly worse than Attention

◎ Neural Transducer is very hard to train, so pre-training is important.

◎ Neural Transducer is very suitable for online decoding.

Reference

[1] Graves, Alex, et al. "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks." Proceedings of the 23rd international conference on Machine learning. ACM, 2006.

[2] Graves A. Supervised sequence labelling[M]//Supervised sequence labelling with recurrent neural networks. Springer, Berlin, Heidelberg, 2012: 5-13.

[3] Graves A. Supervised sequence labelling[M]//Supervised sequence labelling with recurrent neural networks. Springer, Berlin, Heidelberg, 2012: 5-13.

[4] Sak H, Shannon M, Rao K, et al. Recurrent neural aligner: An encoder-decoder neural network model for sequence to sequence mapping[C]//Proc. Interspeech. 2017: 1298-1302.

[5] Prabhavalkar R, Rao K, Sainath T N, et al. A comparison of sequence-to-sequence models for speech recognition[C]//Proc. Interspeech. 2017: 939-943.

[6] Prabhavalkar R, Rao K, Sainath T N, et al. A comparison of sequence-to-sequence models for speech recognition[C]//Proc. Interspeech. 2017: 939-943.

[7] Exploring Architectures, Data and Units For Streaming End-to-End Speech Recognition with RNN-Transducer

Thanks

Q & A