Joint CTC-Attention based End-to-End Speech Recognition using Multi-task Learning

Suyoun Kim [1 2], Takaaki Hori [1], and Shinji Watanabe [1]

[1]Mitsubishi Electric Research Laboratories (MERL)
[2]Carnegie Mellon University (CMU)

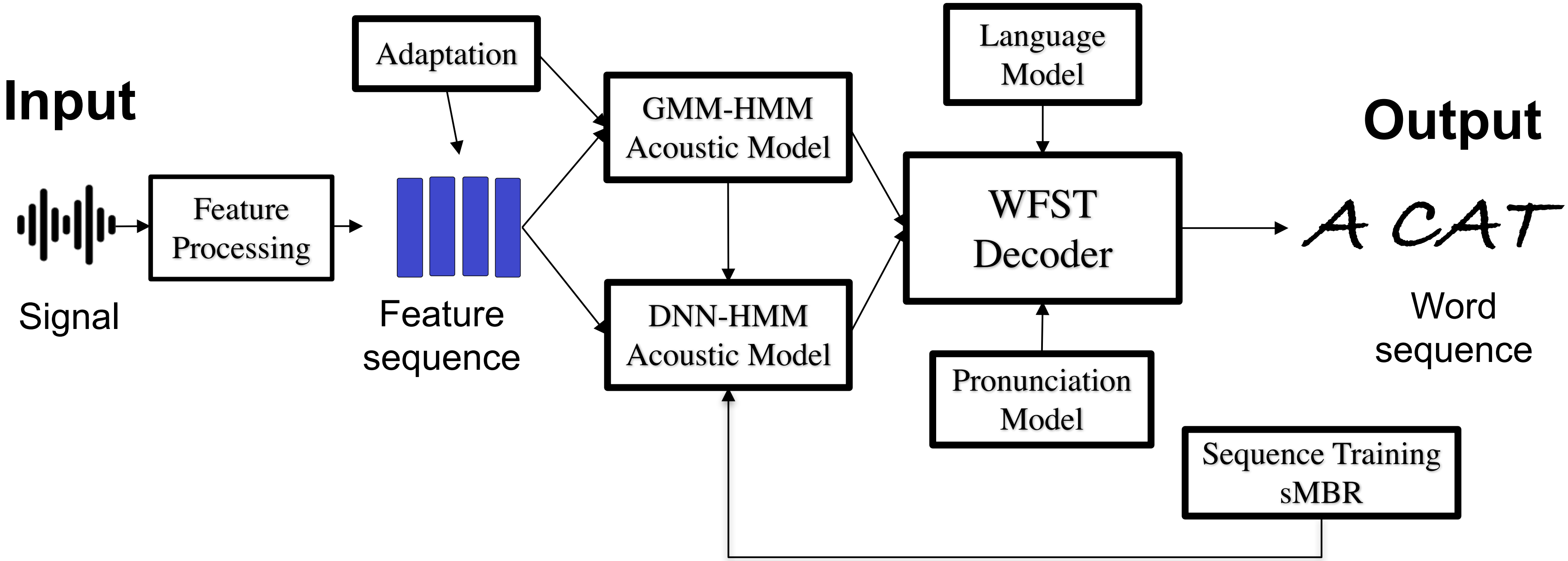
ICASSP March 6 2017
Presenter: Suyoun Kim

Outline

- Introduction and motivation
- Our proposed model: Joint CTC/Attention
- Experiments and results
- Conclusion

Automatic Speech Recognition (ASR)

- ASR is transcribing speech signal to text
- Conventional ASR system is split into multiple sub-components



Conventional ASR is Complicated

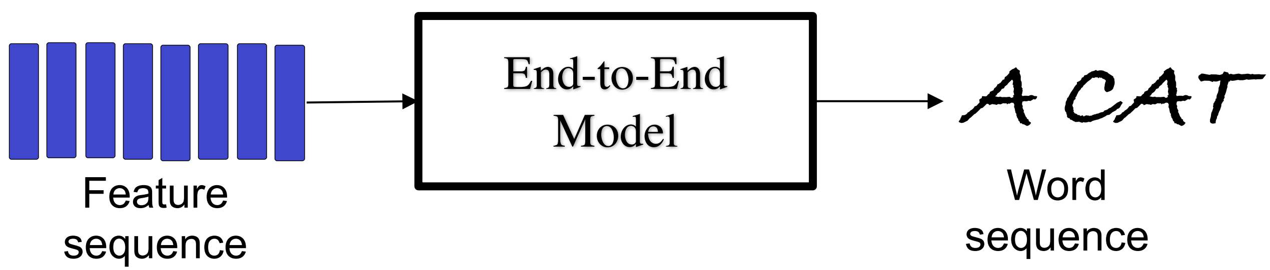
- **Many sub-components**
  - System development is **complicated**
  - Separate modeling may cause **suboptimal**
  - Decoding algorithm is **complex**
- **Many assumptions**
  - Assumes future process only depends on current state not previous state (Markovian, Stationary)
    - $P(s_{t+1}|s_{1:T}) = P(s_{t+1}|s_t)$
    - $P(s_{t_1+1} = i|s_{t_1} = j) = P(s_{t_2+1} = i|s_{t_2} = j)$ for any t_1 and t_2
  - Assumes observations are independent given state (Conditional independent)
    - $P(x_t|x_{1:T}, s_{1:T}) = P(x_t|s_t)$
  - Assumes all pronunciations can be represented by several phonemes (hand-crafted knowledge)
    - Linguistic expertise is required

End-to-End ASR is transcribing speech signal to text directly with a single model, one step training

Big data and **powerful computational engine**

Deep Learning e.g. Recurrent Neural Network(RNN)

End-to-End approach



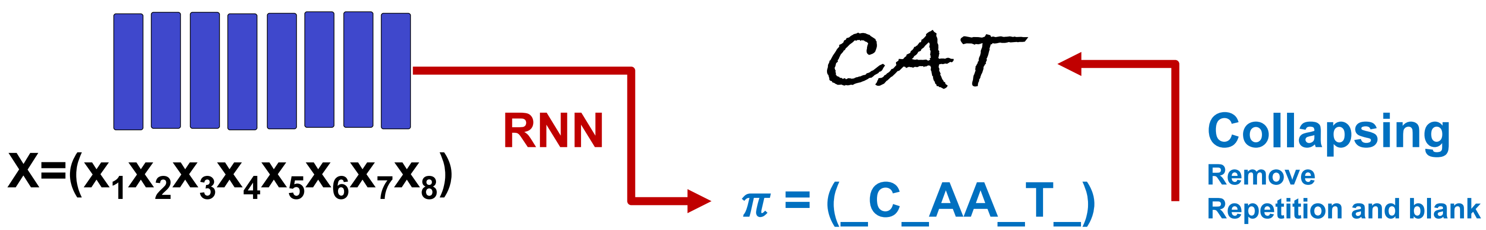
Our Joint CTC/Attention model for End-to-End ASR

- Key insight:
  - We can address the weaknesses of two main End-to-End approaches **1) CTC**, and **2) Attention model** by combining the two, as they have complementary characteristics

CTC + **Attention model** → **Our Joint CTC/Attention**

End-to-End approach 1:
Connectionist Temporal Classification (CTC) [Graves(2006)]

- It uses **intermediate label representation π** allowing **repetitions** and **blank** labels "_"
- It maximizes the **total probability** of all possible label sequence π
- It uses **forward-backward algorithm** for the efficient training



$+$ P(π = _C_AA_T_ | $x_1x_2x_3x_4x_5x_6x_7x_8$;θ)
$+$ P(π = _CCA_TT_ | $x_1x_2x_3x_4x_5x_6x_7x_8$;θ)
…
$= \Sigma$ **P(π|X;θ)**

Strength: There is no need for pronunciation model

Weakness: It still relies on conditional independence assumption, typically separate LM is combined

Our Joint CTC/Attention model for End-to-End ASR

- We keep our model **simple**
  - By using Attention model to learn LM jointly

CTC

✕ It requires separate LM

\+

Attention model

○ It can learn LM jointly

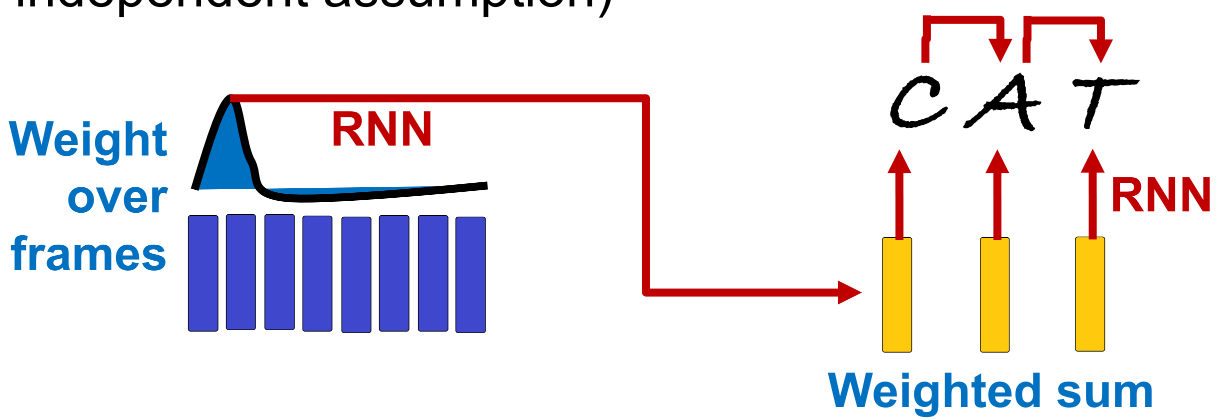
→

Our Joint CTC/Attention

○ It can learn LM jointly
=> **Simple**

End-to-End approach 2:
Attention-based Encoder-Decoder [Chorowski(2014)]

- It uses two RNNs 1) Encoder 2) AttentionDecoder
- For each output step, it estimates weight vector(alignment) over inputs and then decoder uses **weighted sum input**
- Decoder estimates each label **conditioning on previous outputs** (no conditional independent assumption)



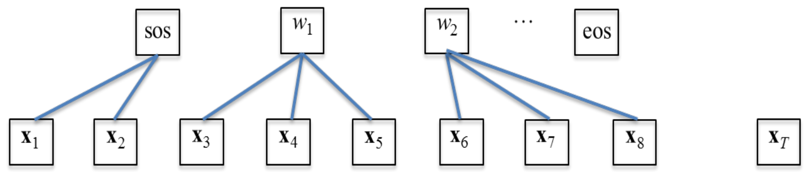
Strength: It can learn acoustic and language model within a single network

Weakness: The alignment can be easily distorted

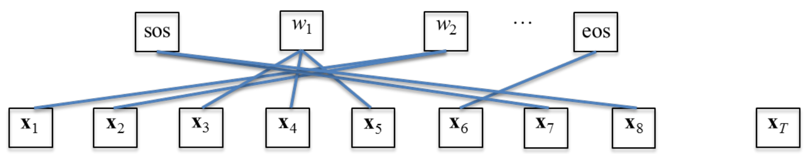
We regularize input/output alignment of attention

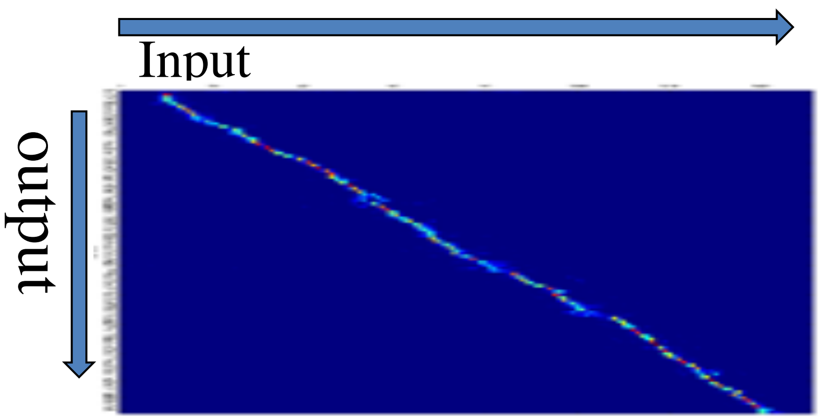
- Unlike CTC, Attention model does not preserve order of inputs
- Our desired alignment in ASR task is **monotonic**
- Not regularized alignment makes the model **hard to learn** from scratch

HMM or CTC case

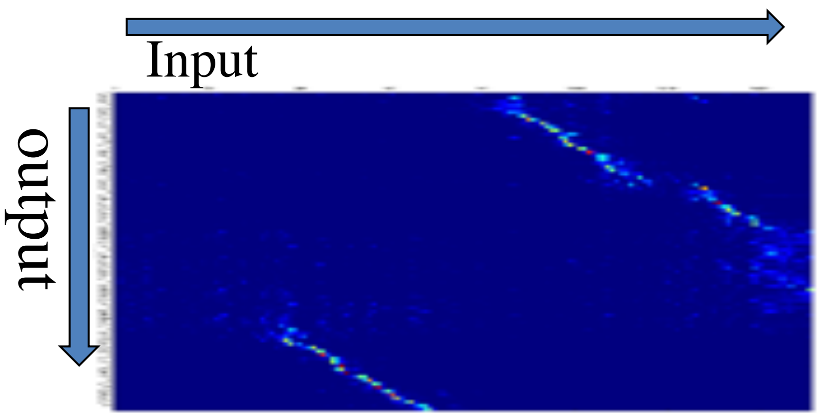


Attention model case





Example of monotonic alignment!



Example of distorted alignment!

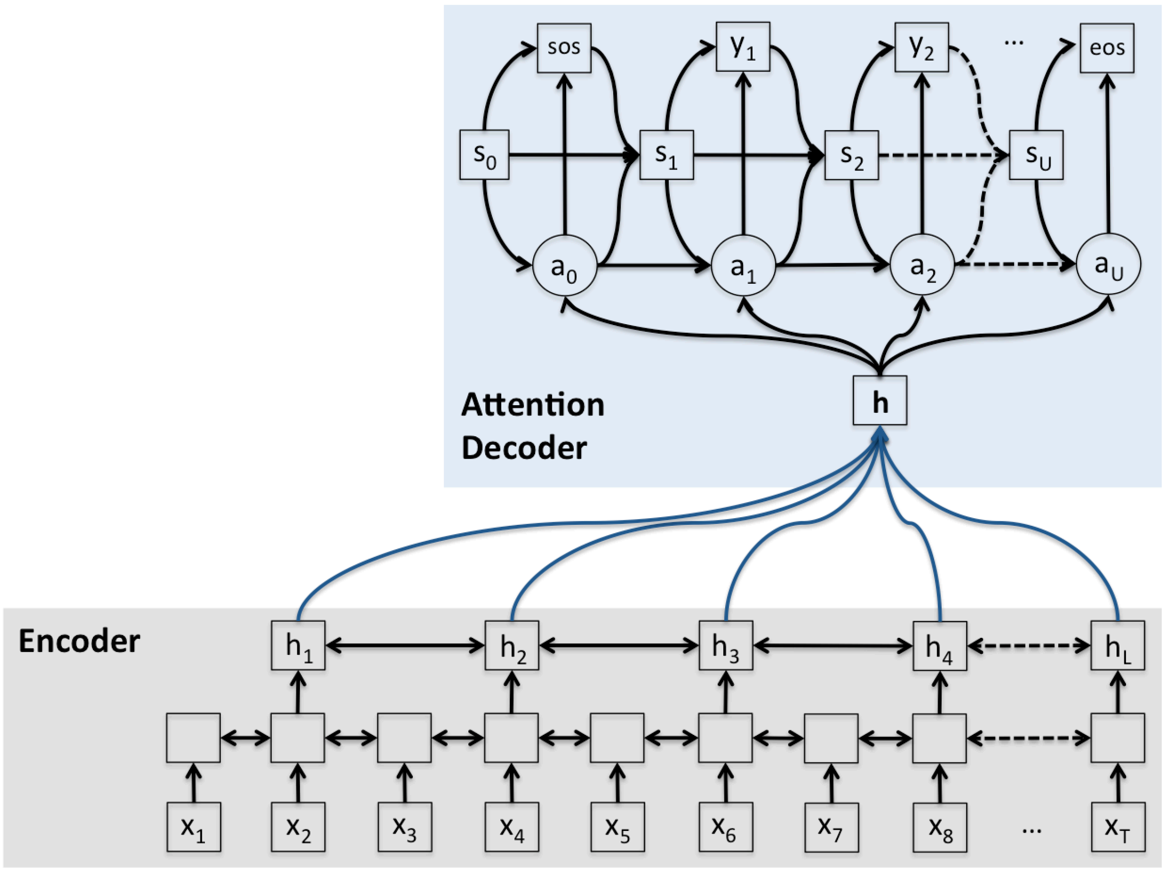
Our Joint CTC/Attention model for End-to-End ASR

- We keep our model **simple**
  - By using Attention model to learn inter-character dependencies jointly
- We improve the **learning speed** and **performance**
  - By using CTC to regularize the input/output alignment

CTC		Attention model		Our Joint CTC/Attention
✕ It requires separate LM	+	○ It can model LM jointly	⇒	○ It can learn LM jointly => **Simple**
○ It preserves input/output order		✕ Input/output alignment is easily distorted ✕ It is hard to train from scratch		○ It can regularize the input/output alignment ⇒ **Faster convergence** ⇒ **Better performance**

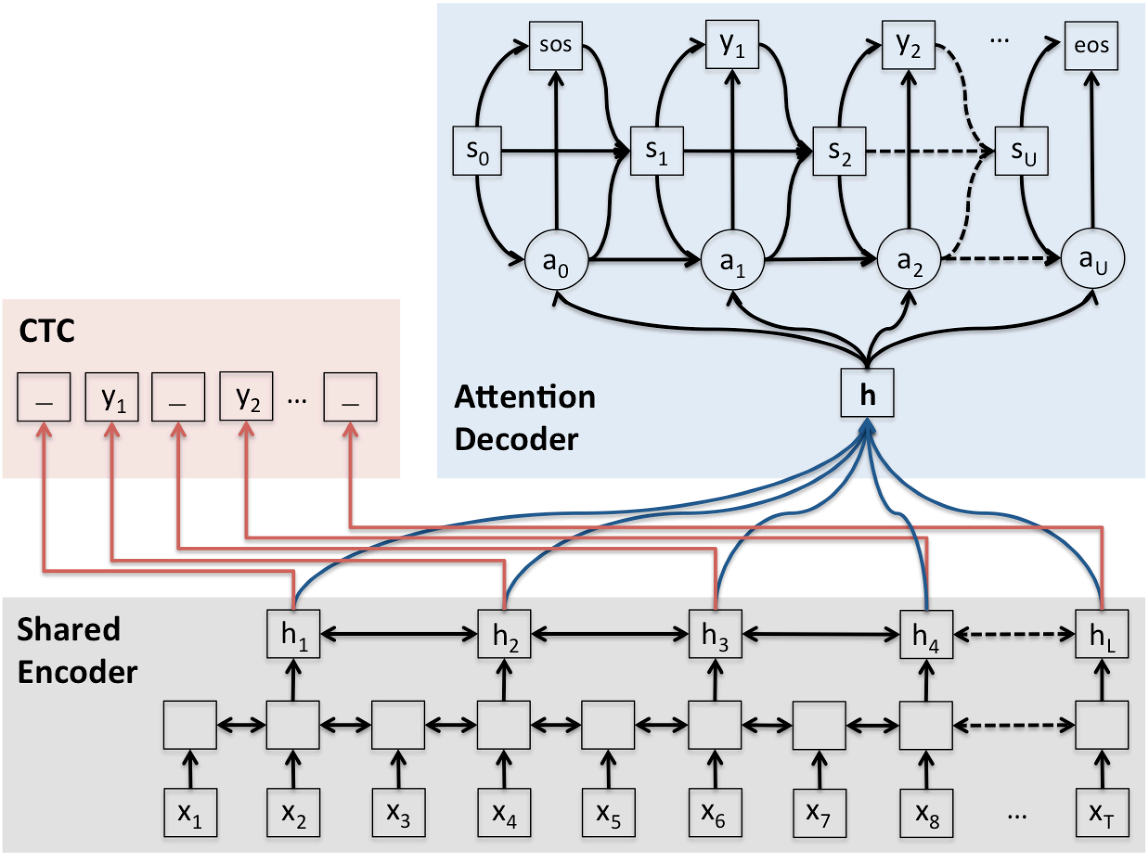
Our Joint CTC/Attention model for End-to-End ASR

- Standard Attention model



Our Joint CTC/Attention model for End-to-End ASR

- Multi-task learning framework



Our Joint CTC/Attention model for End-to-End ASR

1. We share the encoder part
2. We train Attention model with CTC jointly

<table>
<tr><td colspan="2">Larger λ will give more weight on CTC objective.
$\mathcal{L}_{\text{MTL}} = \lambda \mathcal{L}_{\text{CTC}} + (1 - \lambda) \mathcal{L}_{\text{Attention}}$</td></tr>
<tr><td>Global normalization
$\mathcal{L}_{\text{CTC}} \triangleq -\ln P(\boldsymbol{y}^* \mid \boldsymbol{x}) = -\ln \sum_{\boldsymbol{\pi} \in \Phi(\boldsymbol{y}')} P(\boldsymbol{\pi} \mid \boldsymbol{x})$</td><td>Local normalization
$\mathcal{L}_{\text{Attention}} \triangleq -\ln P(\boldsymbol{y}^* \mid \boldsymbol{x}) = -\sum_{u} \ln P(y^*_u \mid \boldsymbol{x}, y^*_{1:u-1})$</td></tr>
</table>

3. We use AttentionDecoder on decoding mode
   - The cost for CTC exists only on training mode

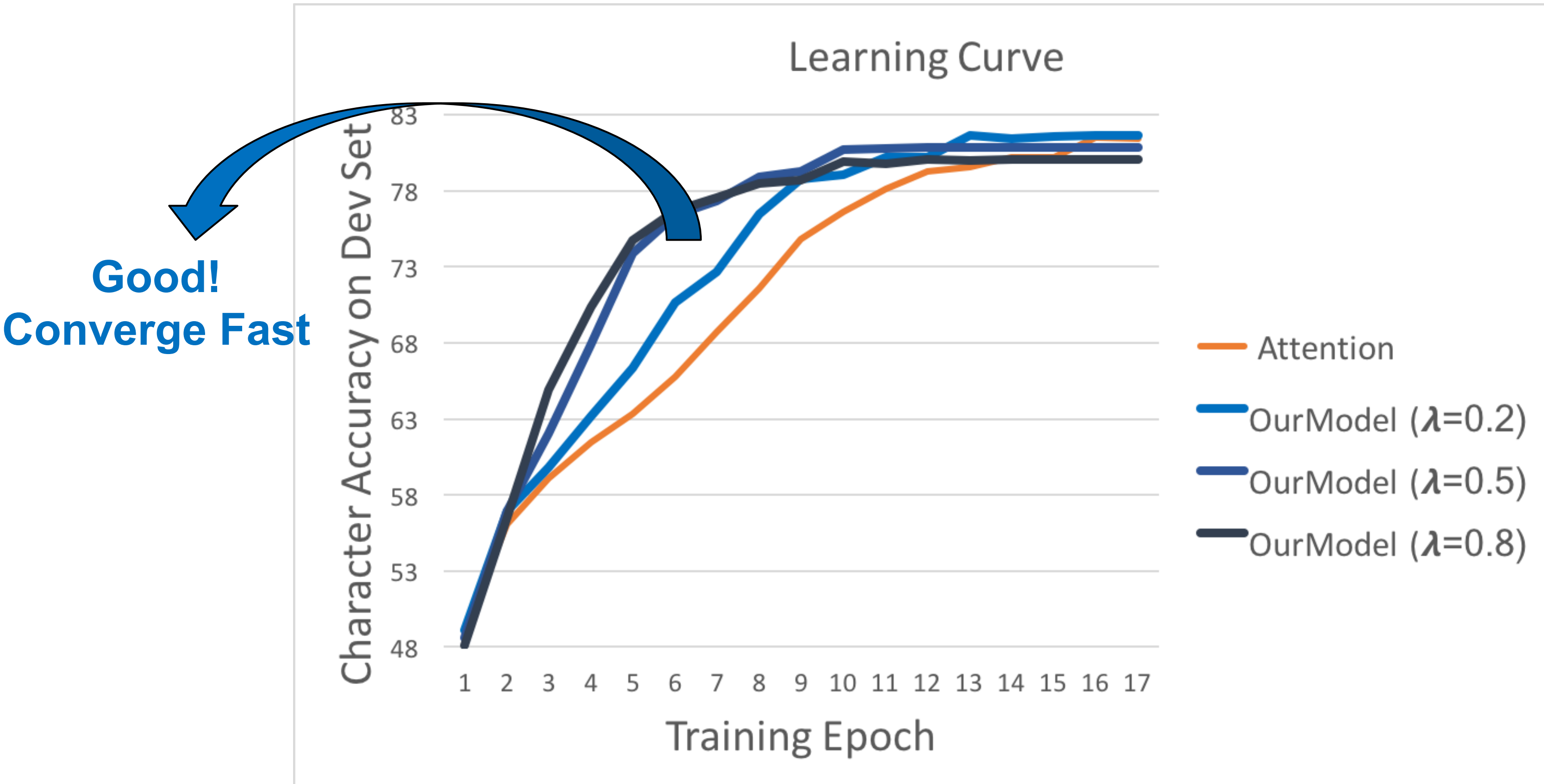
Outline

- Introduction and motivation
- Our proposed model: Joint CTC/Attention
- Experiments and results
- Conclusion

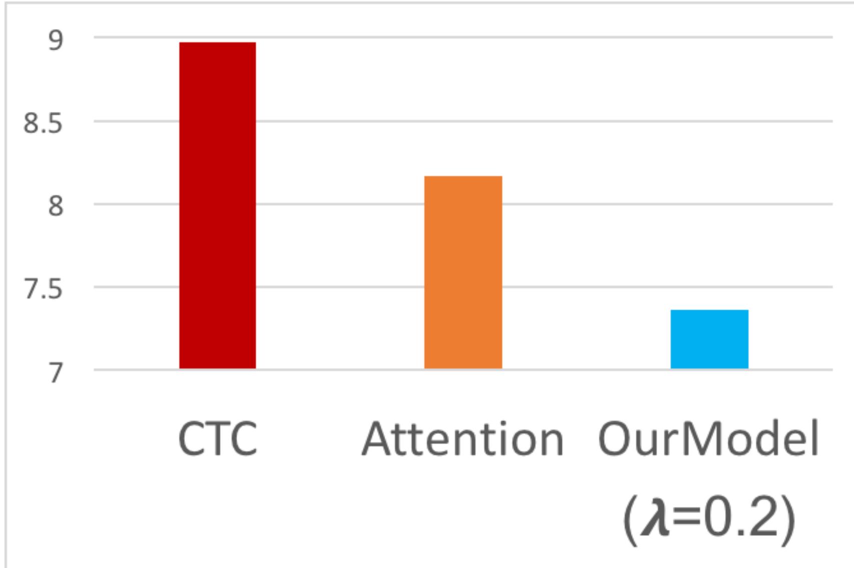
Experiment setup

- Dataset
  - WSJ0 (si84) – 15 hours clean
  - WSJ1 (si284) – 80 hours clean
  - CHiME4 – 18 hours noisy
  - Input – 120d filterbank (+d, +dd)
  - Output – 32 distinct label (+26 char, + apostrophe, period, …, sos/eos)
- Baselines
  - CTC – 4 layer BLSTM (320 cells)
  - Attention – 4 layer BLSTM encoder (320 cells) + 1 layer LSTM decoder (320 cells), location-based attention mechanism
- Our Joint CTC/Attention model
  - 4 layer BLSTM encoder (320 cells) + 1 layer LSTM decoder (320 cells)
  - With λ= {0.2 0.5 0.8}
- Evaluation
  - Character Error Rate (CER)

Faster convergence compared to Attention model



9.9% relative improvement of CER on WSJ1(80hr)



Lower is Better!

λ=0.2 performs best!

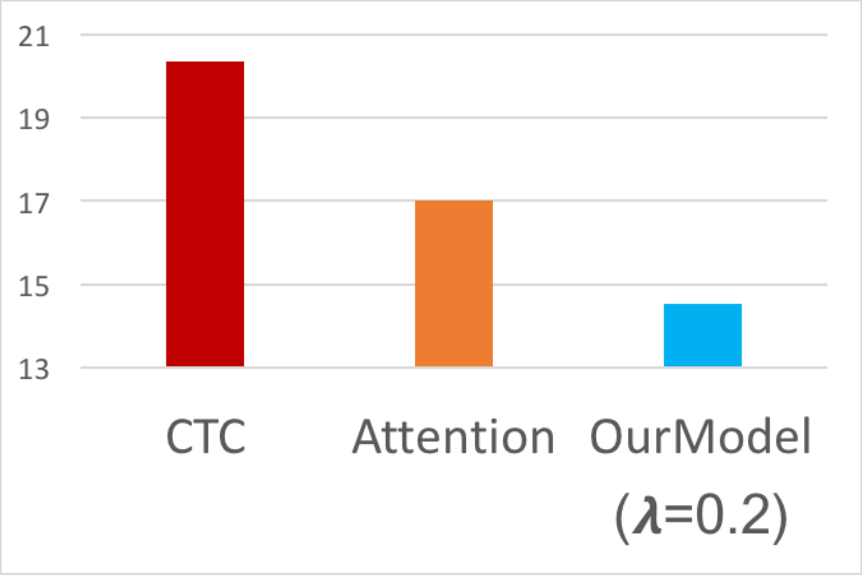
Larger λ gives more weight on CTC

	Dev	Eval
CTC	11.5	9.0
Attention	12.0	8.2
OurModel (λ=0.2)	**11.3**	**7.4**
OurModel (λ=0.5)	12.0	8.3
OurModel (λ=0.8)	11.7	8.5

9.9% improvement

WER of our best system was 18.2%
WER of (Bahdanau, et al. ICASSP 2016) was 18.6%

14.6% relative improvement of CER on WSJ0(15hr)



Lower is Better!

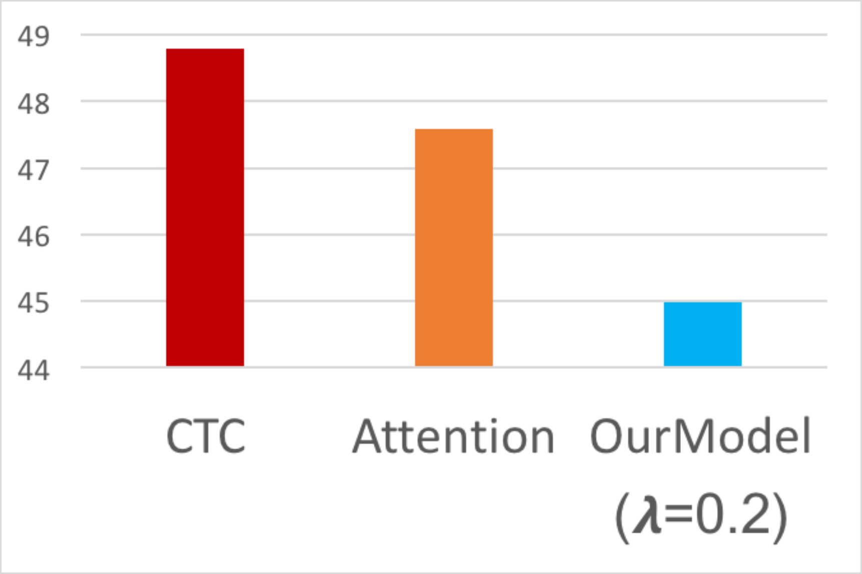
λ=0.2 performs best!

Larger λ gives more weight on CTC

	Dev	Eval
CTC	27.4	20.3
Attention	25.0	17.0
OurModel (λ=0.2)	**23.0**	**14.5**
OurModel (λ=0.5)	26.3	16.2
OurModel (λ=0.8)	32.2	21.3

14.6% improvement

5.4% relative improvement of CER on CHiME4(18hr)



Lower is Better!

λ=0.2 performs best!

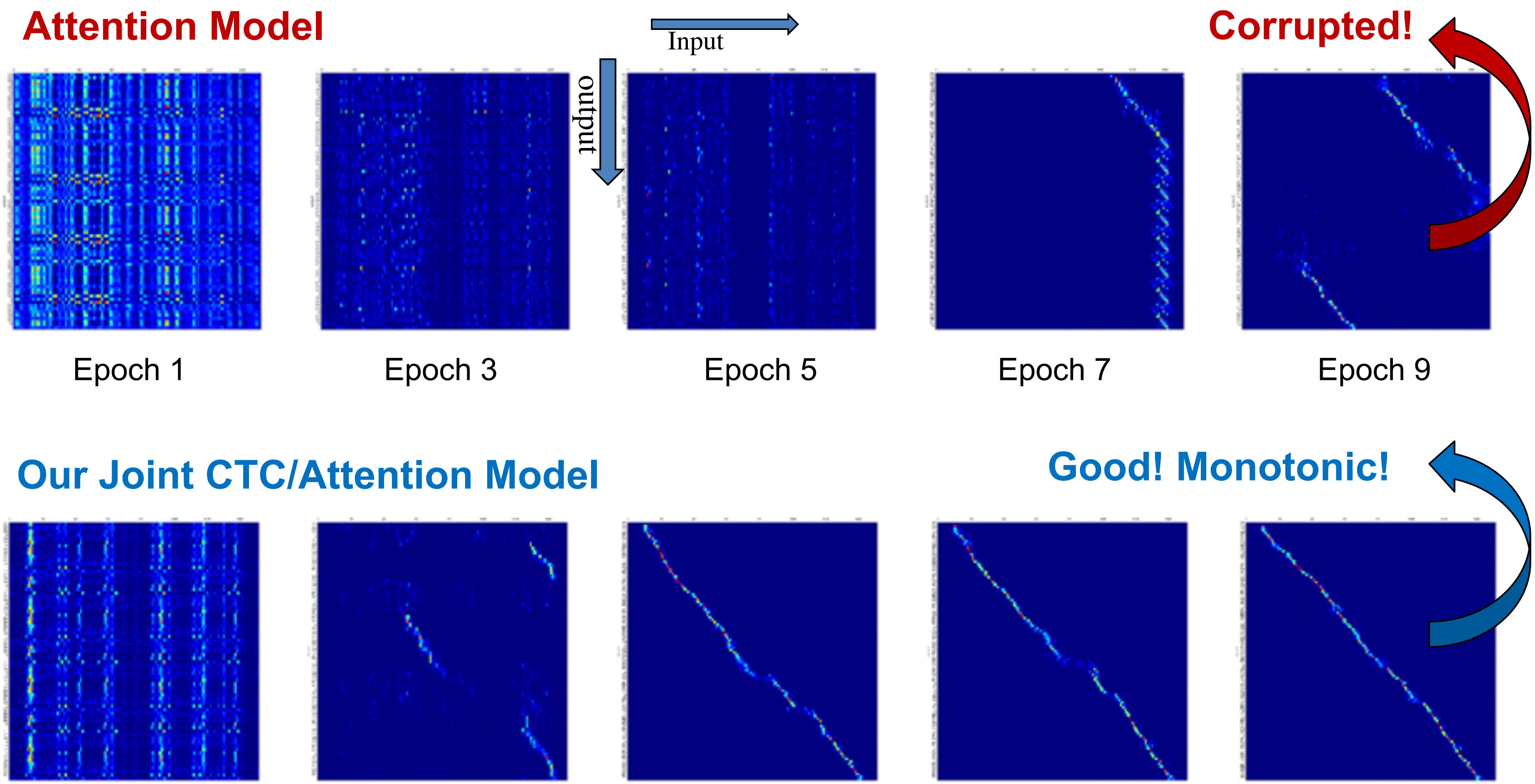
Larger λ gives more weight on CTC

	Dev	Eval
CTC	37.6	48.8
Attention	35.0	47.6
OurModel (λ=0.2)	**32.1**	**45.0**
OurModel (λ=0.5)	34.6	46.5
OurModel (λ=0.8)	35.4	48.3

5.4% improvement

More robust input/output alignment of attention

- Alignment of one selected utterance from CHiME4



Outline

- Introduction and motivation
- Our proposed model: Joint CTC/Attention
- Experiments and results
- Conclusion

Conclusion

- Joint CTC/Attention model
  - does not use any linguistic information
  - shows 5.4 – 14.6 % relative improvements in CER, compared to Attention-based Encoder-Decoder
  - speeds up learning process
  - requires small additional computational cost but only in training mode, not in decoding mode.

- Our framework can be applied to other seq2seq tasks where its alignment is monotonic

Current research

- Further experimental results on Corpus of Spontaneous Japanese (CSJ) – 581hr
  - Achieved comparable performance to **state-of-the-art**

	task1	task2	task3
Attention (581h)	11.5	7.9	9.0
OurModel (581h)	10.9	7.8	8.3
OurModel2 (581h)	**9.5**	**7.0**	**7.8**
DNN/sMBR-hybrid (236h for AM/ 581h for LM)	9.0	7.2	9.6
CTC-syllable (581h)	9.4	7.3	7.5

Thank you!

Questions & Answers