The 33rd Conference on Computational Linguistics and Speech Processing

October 15-16, 2021 • Online and National Central University, Taiwan

ROCLING 2021 NCU

Advancing End-to-End Automatic Speech Recognition

Jinyu Li

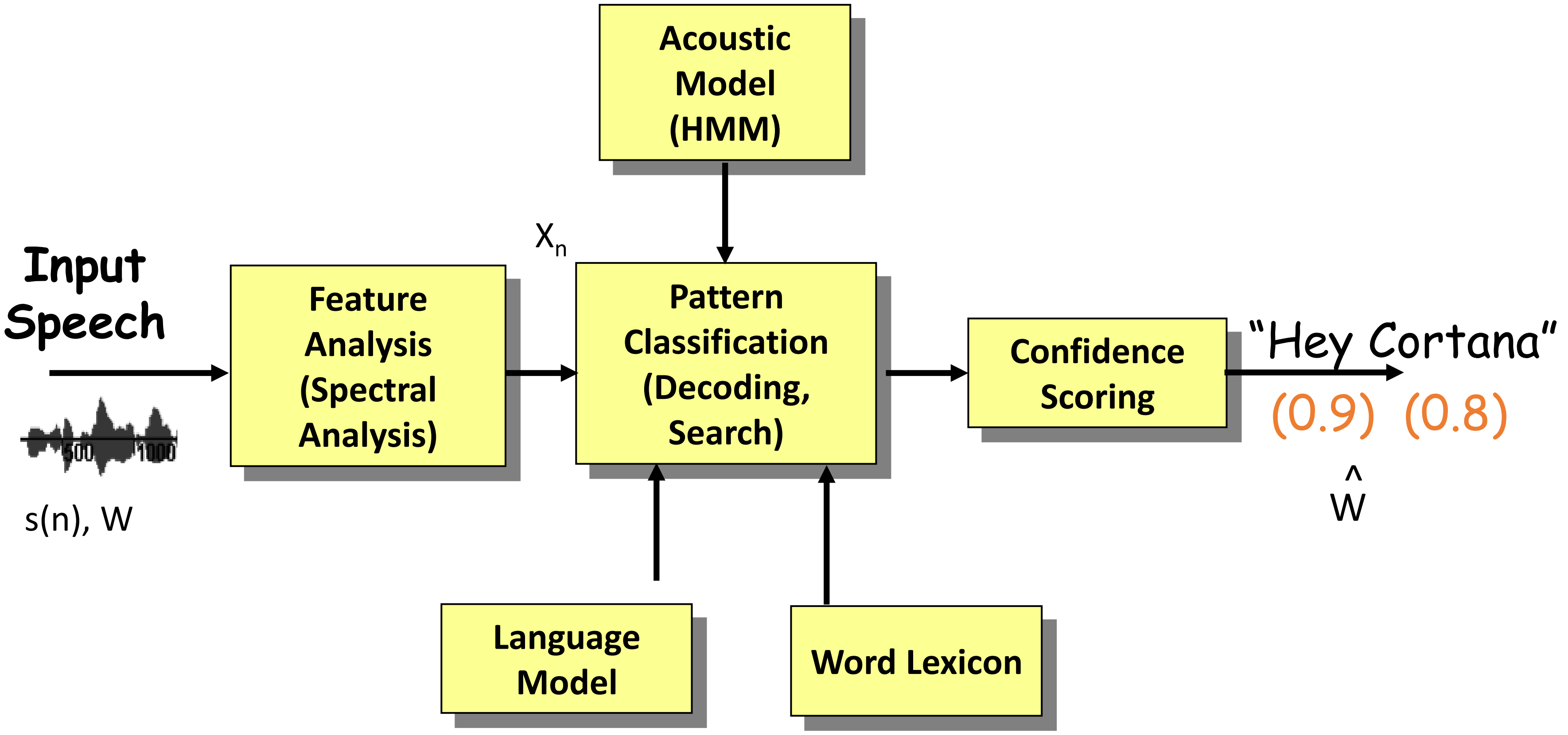
Microsoft

Outline

- End-to-end (E2E) ASR fundamental
  - CTC
  - AED
  - RNN-T
- E2E advances
  - Encoder
  - Multilingual
  - Adaptation
  - Advanced models

E2E Fundamental

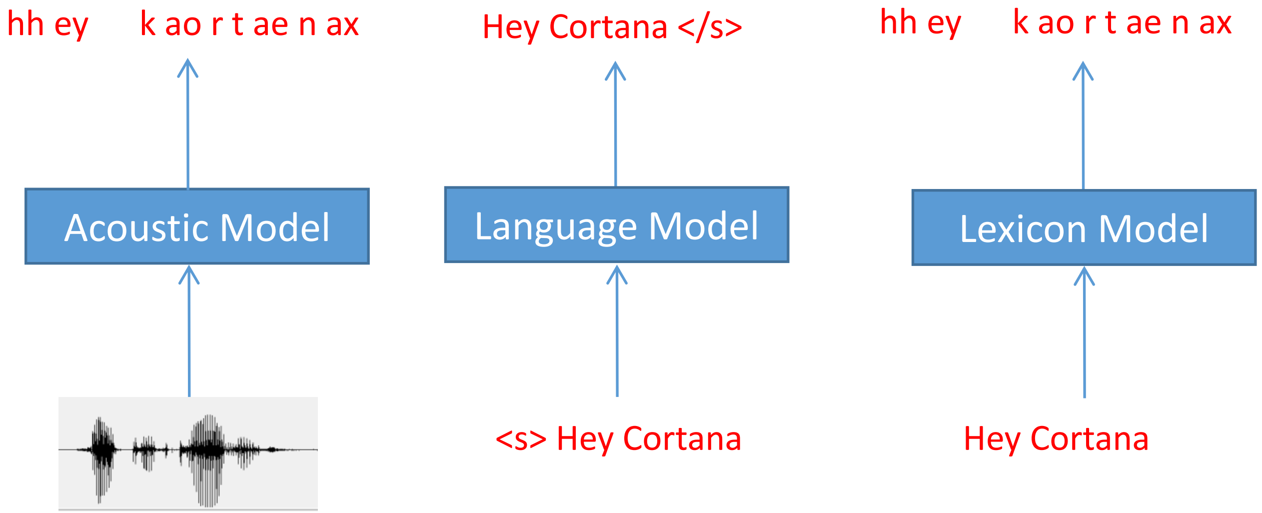
Conventional Automatic Speech Recognition (ASR)



Hybrid vs. End-to-End (E2E) Modeling

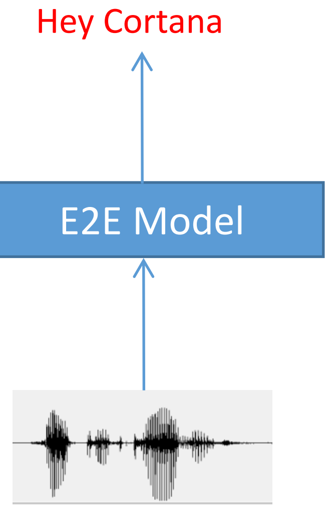
Hybrid

Separate models are trained, and then are used all together during testing in an ad-hoc way.



E2E

A single model is used to directly map the speech waveform into the target word sequence.



Advantages of E2E Models

- E2E models use a single objective function which is consistent with the ASR objective
- E2E models directly output characters or even words, greatly simplifying the ASR pipeline
- E2E models are much more compact than traditional hybrid models -- can be deployed to devices with high accuracy and low latency

Graves and Jaitly, “Towards end-to-end speech recognition with recurrent neural networks” PMLR, 2014.
Hannun et al., “Deep speech: Scaling up end-to-end speech recognition,” in arXiv preprint, 2014.

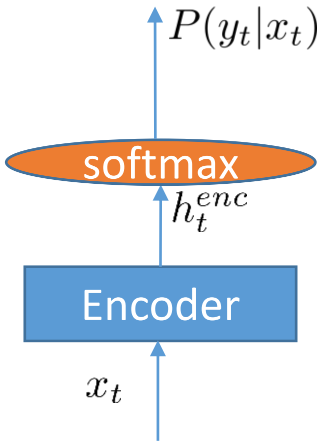
Current Status

- E2E models achieve the state-of-the-art results in most benchmarks in terms of ASR accuracy.

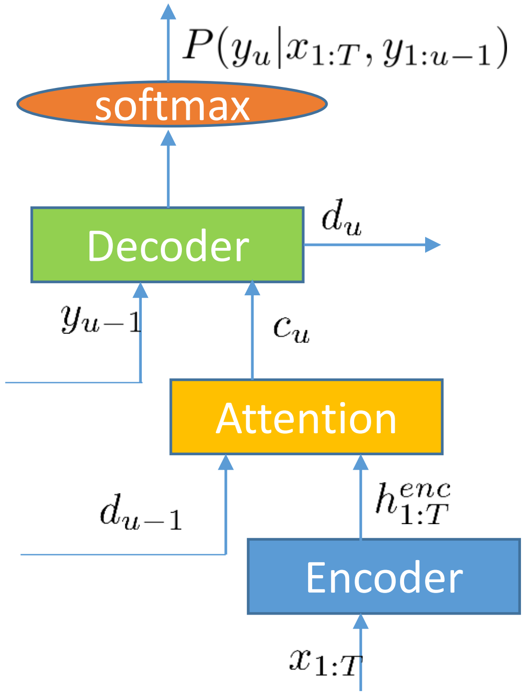
- Hybrid models still dominate commercial ASR systems currently, because they are fully optimized for decades for practical challenges such as streaming, latency, adaptation capability etc.

- In this talk, we overview the popular E2E models with the focus on technologies addressing those challenges *from industry perspective*.

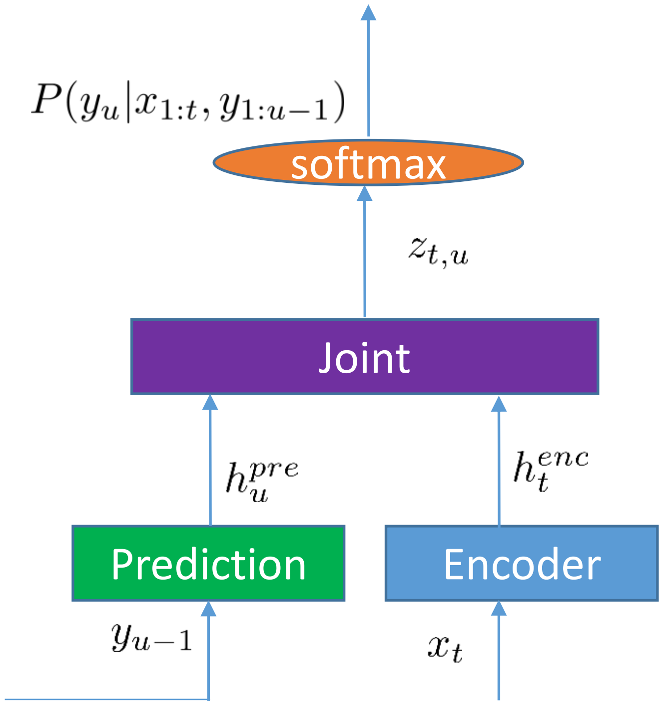
E2E Models



Connectionist Temporal Classification (CTC)



Attention-based encoder decoder (AED)



RNN-Transducer (RNN-T)

CTC

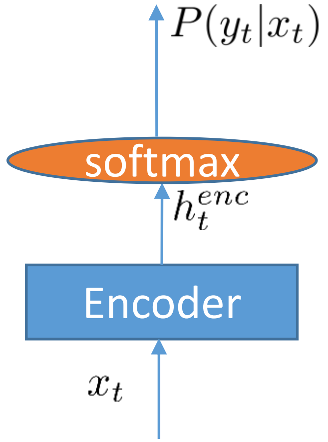
- The first and simplest E2E ASR model.
- To solve the challenge that target label length is smaller than the speech input length:
  - Inserts blank and allows label repetition to have the same length of CTC path and speech input sequence.

$$P(\mathbf{y}|\mathbf{x}) = \sum_{\mathbf{q}\in\mathbf{B}^{-1}(\mathbf{y})} P(\mathbf{q}|\mathbf{x})$$

- Frame independence assumption

$$P(\mathbf{q}|\mathbf{x}) = \prod_{t=1}^{T} P(q_t|\mathbf{x})$$

- Revives with the Transformer encoder and the emerged self-supervised learning technologies



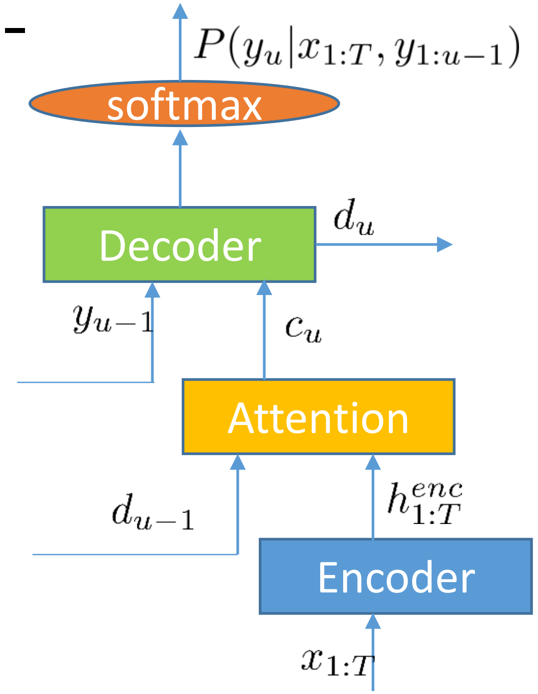
Graves et al., "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in Proc. ICML, 2006.

AED

- The sequence probability is calculated in an auto-regressive way.

$$P(\mathbf{y}|\mathbf{x}) = \prod_u P(y_u|\mathbf{x}, \mathbf{y}_{1:u-1})$$

- Encoder: converts input feature sequences into high-level hidden feature sequences.
- Attention: computes attention weights to generate a context vector as a weighted sum of the encoder output.
- Decoder: takes the previous output label together with the context vector to generate its output $P(y_u|\mathbf{x}, \mathbf{y}_{1:u-1})$



Cho et al., "Learning phrase representations using RNN encoder-decoder for statistical machine translation," in arXiv preprint, 2014.

Streaming

- Most commercial setups need the ASR systems to be streaming with low latency: ASR system produces the recognition results at the same time as the user is speaking.
- Non-streaming ASR is not practical in most ASR scenarios where speech signal comes in a continuous mode without segmentation.
- Full attention in AED may not be ideal to ASR because the speech signal and output label sequence are monotonic.
  - Streaming AED: apply attention on chunks of input speech.

Streaming

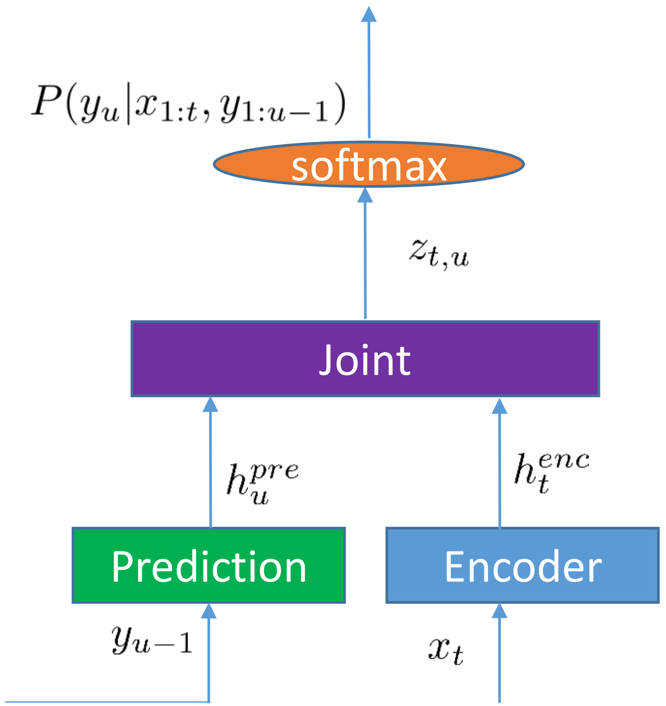
- Most commercial setups need the ASR systems to be streaming with low latency: ASR system produces the recognition results at the same time as the user is speaking.

- Full attention in AED may not be ideal to ASR because the speech signal and output label sequence are monotonic.
  - Streaming AED: apply attention on chunks of input speech.

- RNN-T provides a natural way for streaming ASR because its output conditions on the previous output token and the speech sequence until the current time step.

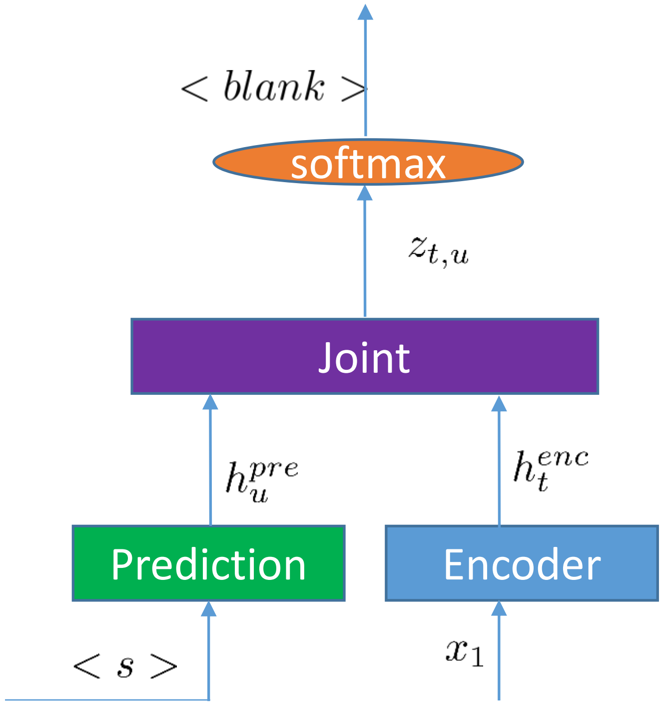
RNN-T

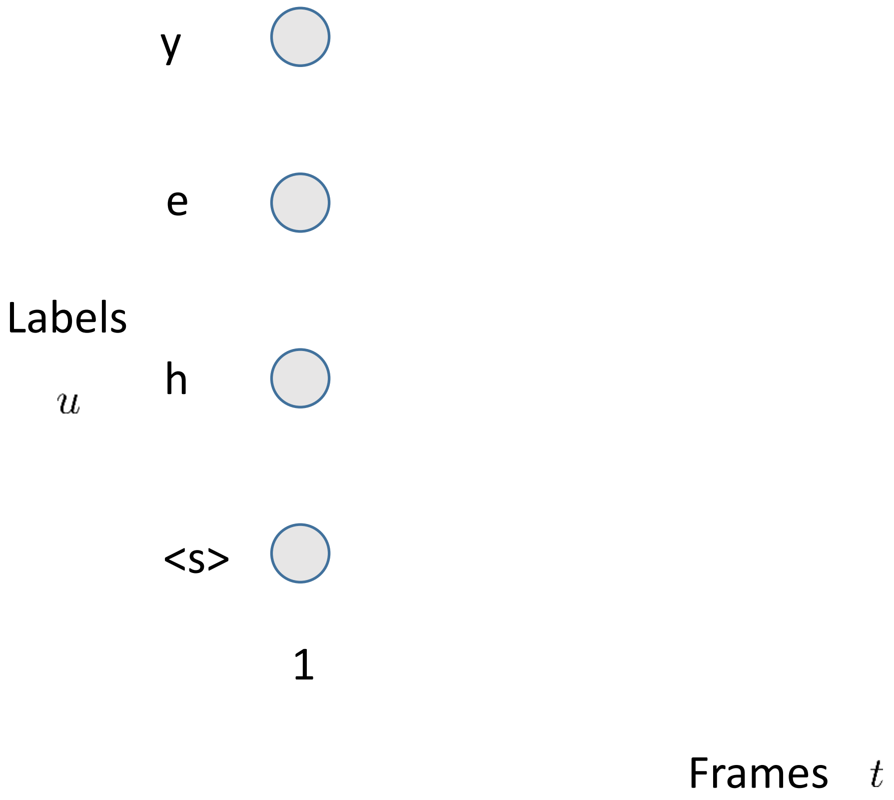
- Encoder: converts input feature sequences into high-level hidden feature sequences.
- Prediction network: producing a high-level representation based on previous label.
- Joint network: combines the outputs from encoder and prediction network.



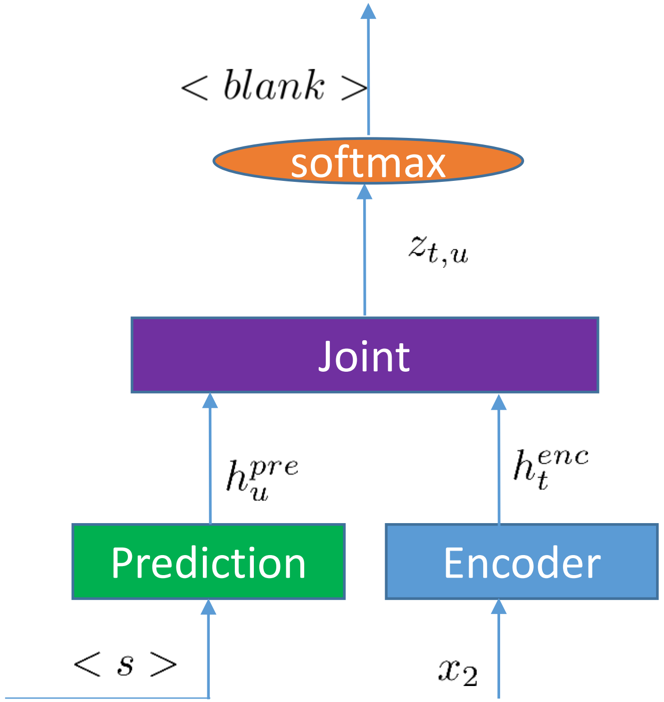
Graves., "Sequence Transduction with Recurrent Neural Networks," in arXiv preprint, 2012.

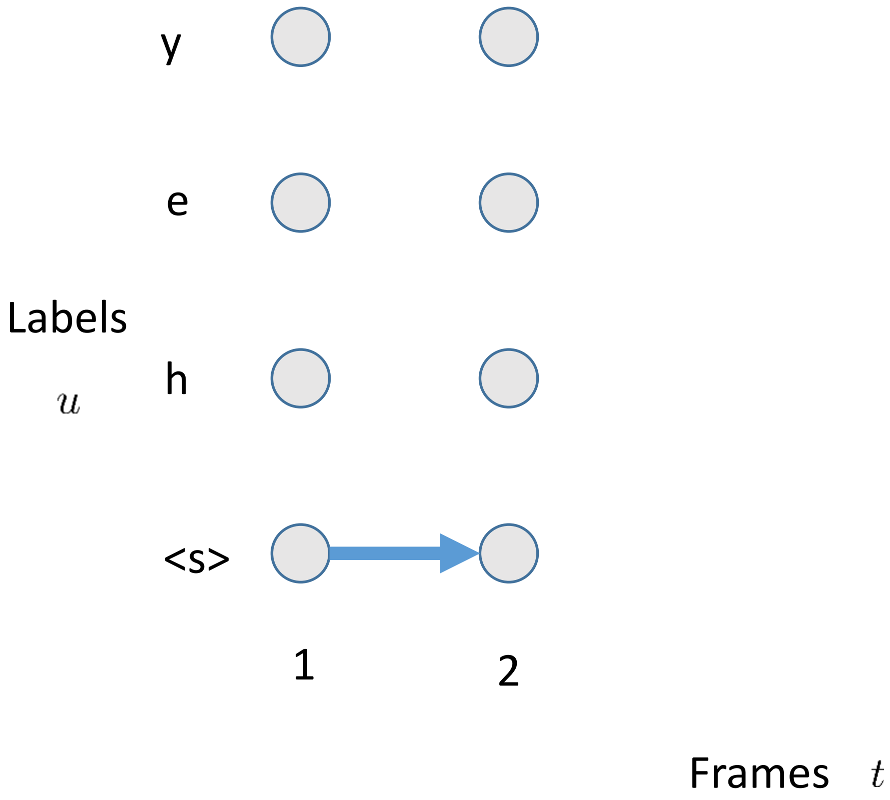
RNN-T Path


$< blank >$
softmax
$z_{t,u}$
Joint
h_u^{pre}
h_t^{enc}
Prediction
Encoder
$< s >$
x_1

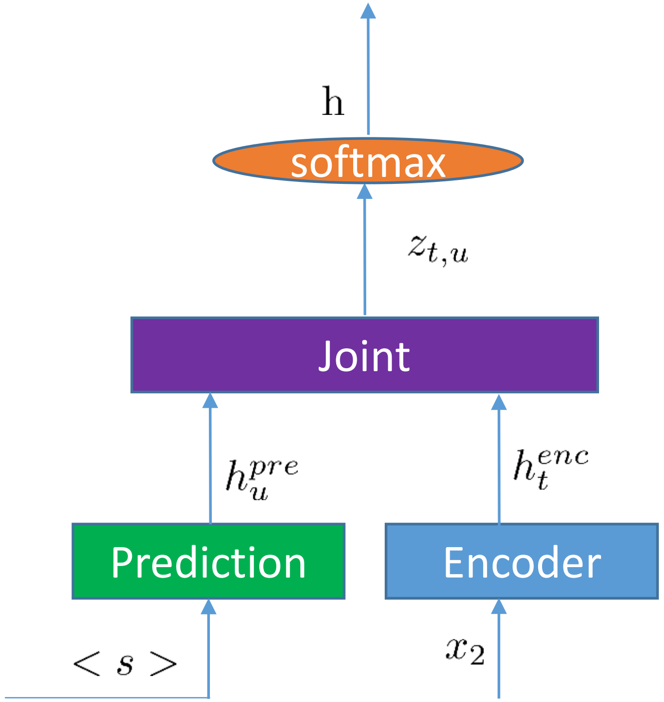

y
e
Labels
u
h
<s>
1
Frames t

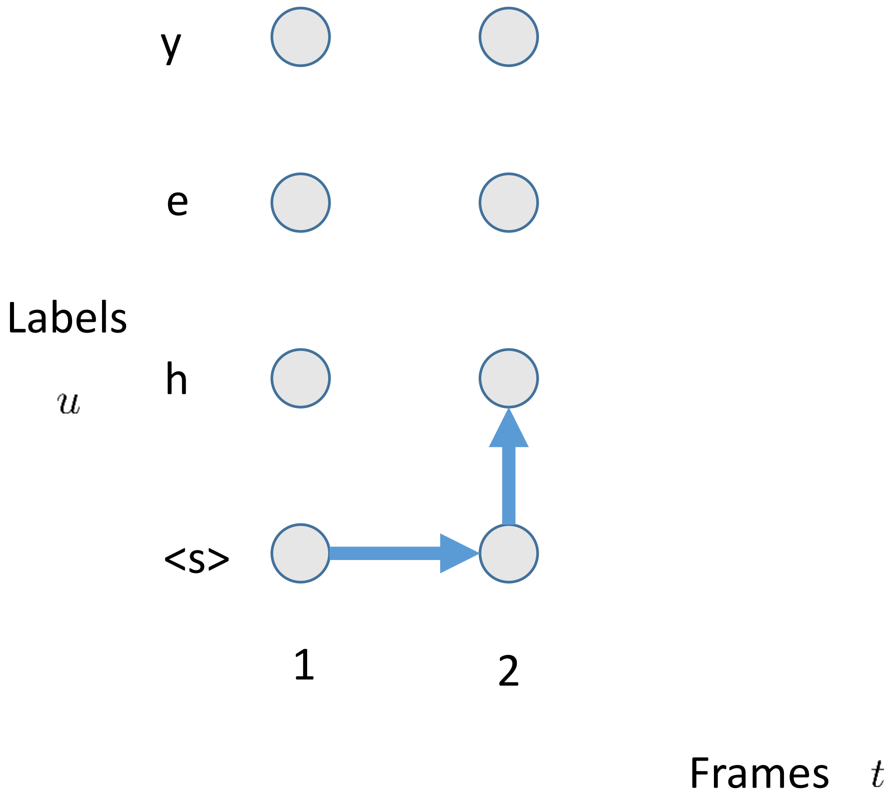
RNN-T Path


$< blank >$
softmax
$z_{t,u}$
Joint
h_u^{pre}
h_t^{enc}
Prediction
Encoder
$< s >$
x_2

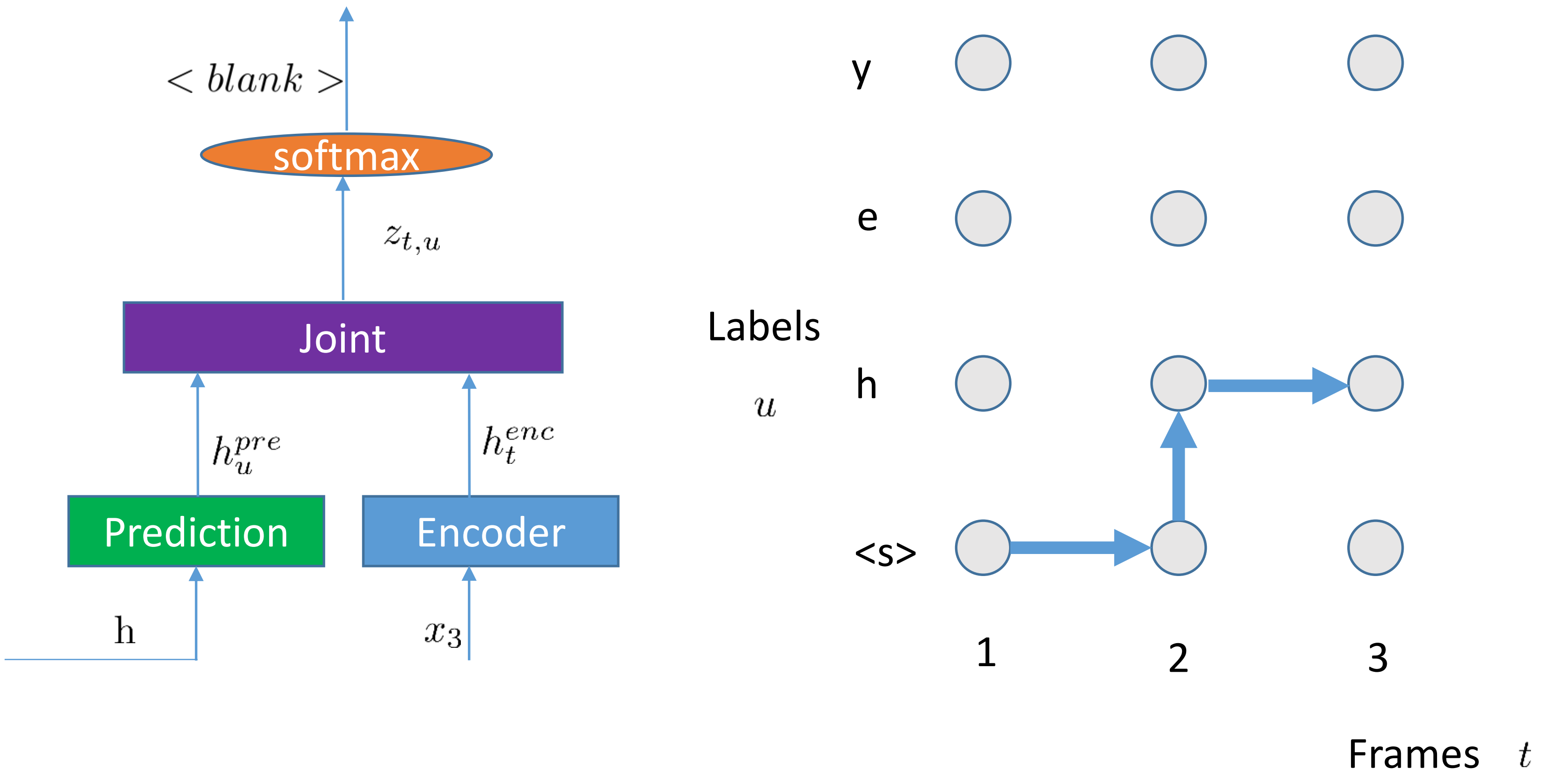

y
e
Labels
u
h
<s>
1
2
Frames t

RNN-T Path

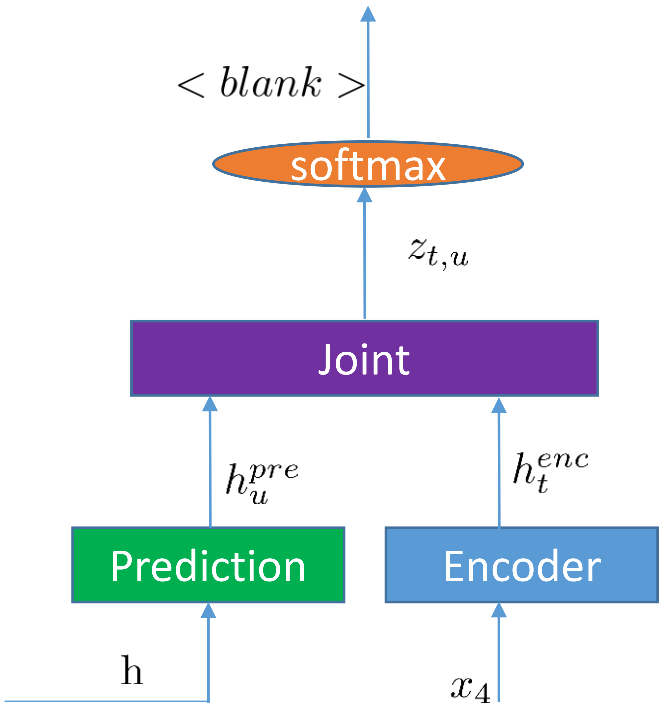

h
softmax
$z_{t,u}$
Joint
h_u^{pre}
h_t^{enc}
Prediction
Encoder
$< s >$
x_2

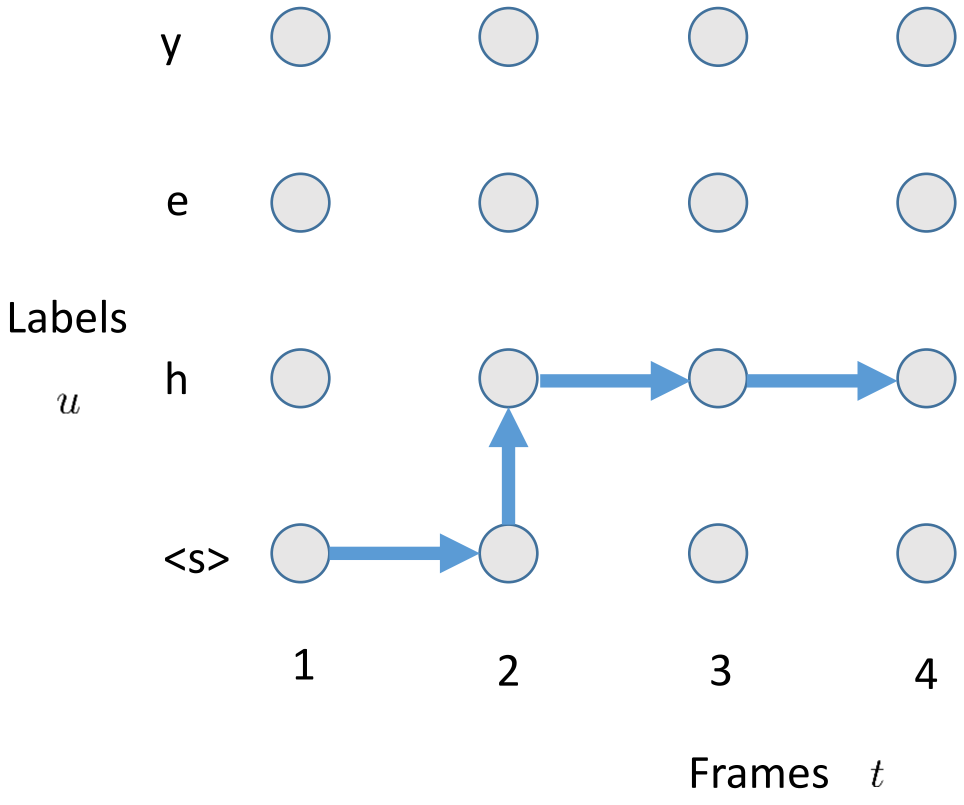

y
e
Labels
u
h
<s>
1
2
Frames t

RNN-T Path

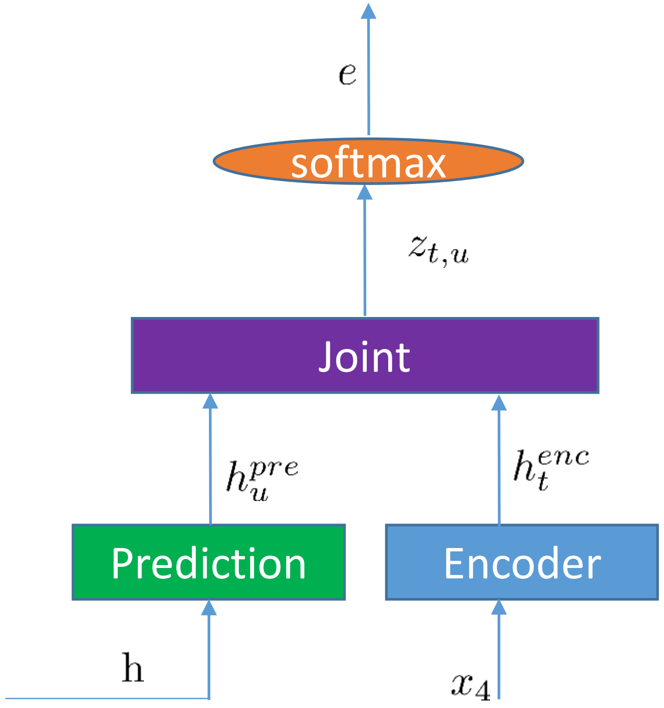

$< blank >$
softmax
$z_{t,u}$
Joint
h_u^{pre}
h_t^{enc}
Prediction
Encoder
h
x_3
y
e
Labels
u
h
<s>
1
2
3
Frames t

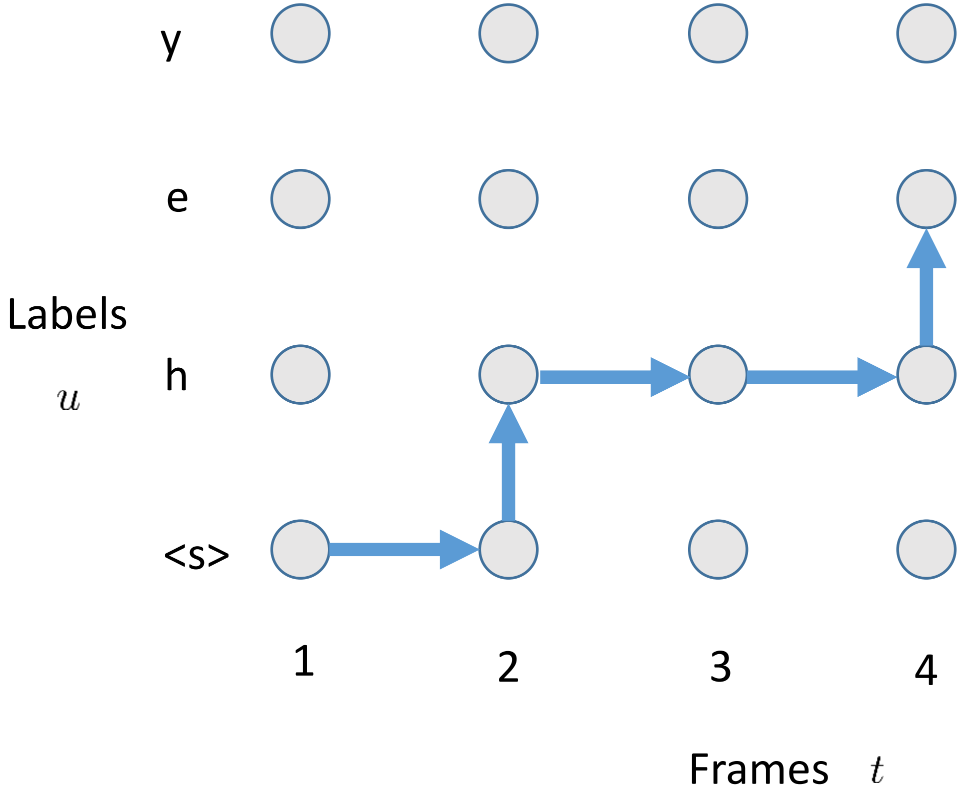
RNN-T Path


$< blank >$
softmax
$z_{t,u}$
Joint
h_u^{pre}
h_t^{enc}
Prediction
Encoder
h
x_4

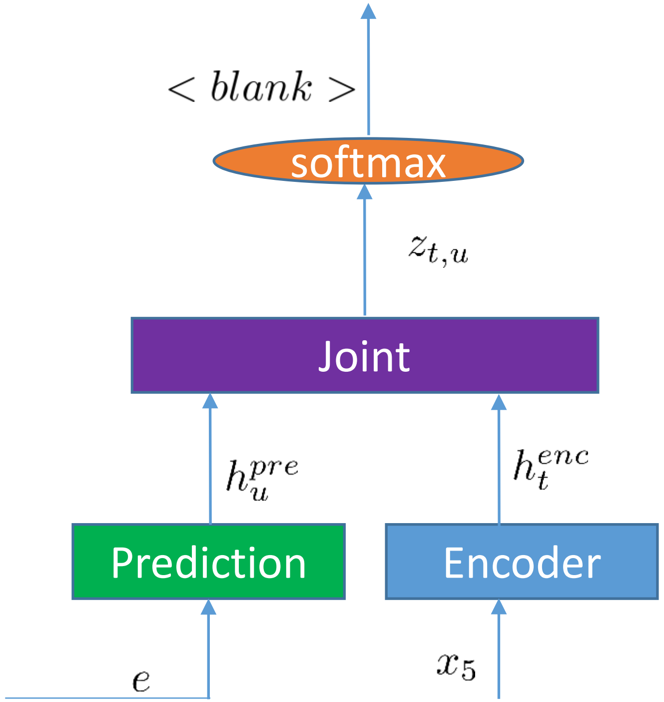

y
e
Labels
u
h
<s>
1
2
3
4
Frames t

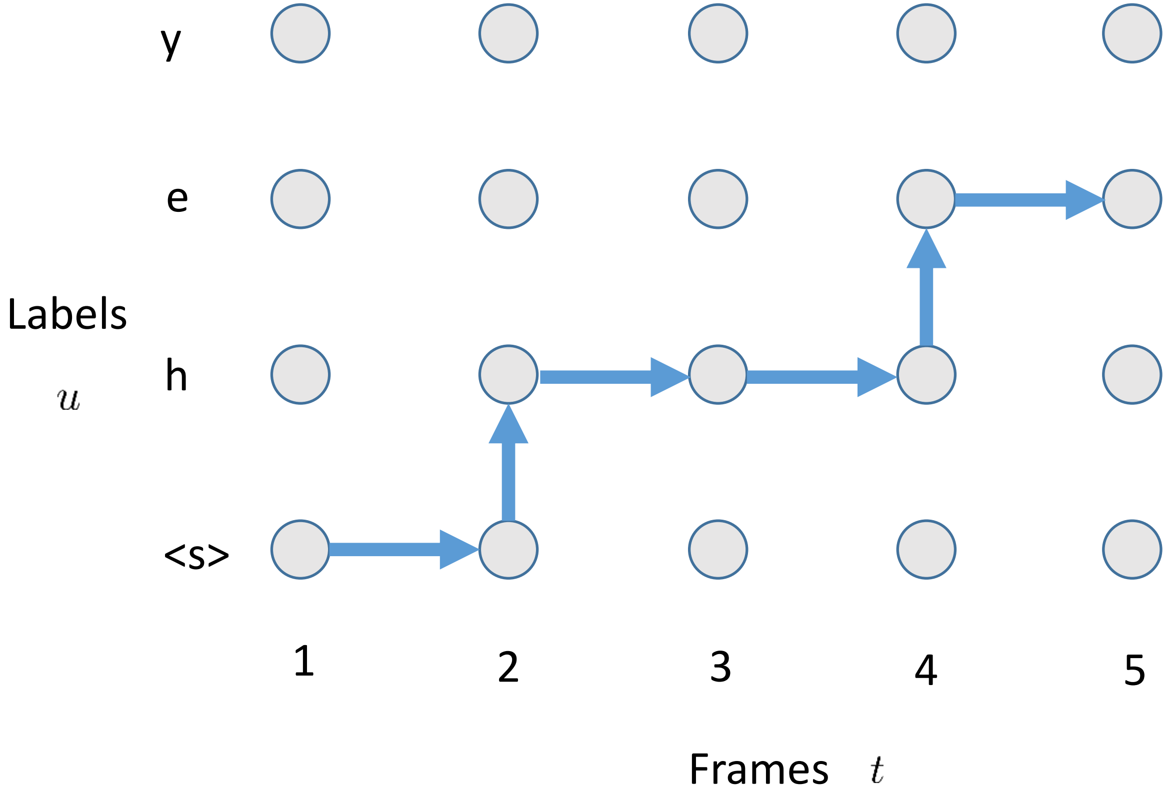
RNN-T Path


e
softmax
$z_{t,u}$
Joint
h_u^{pre}
h_t^{enc}
Prediction
Encoder
h
x_4

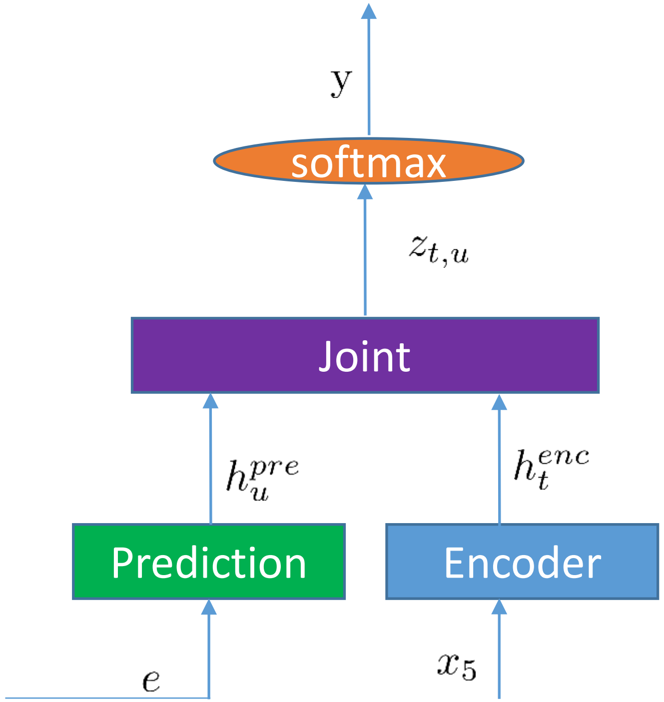

y
e
Labels
u
h
<s>
1
2
3
4
Frames t

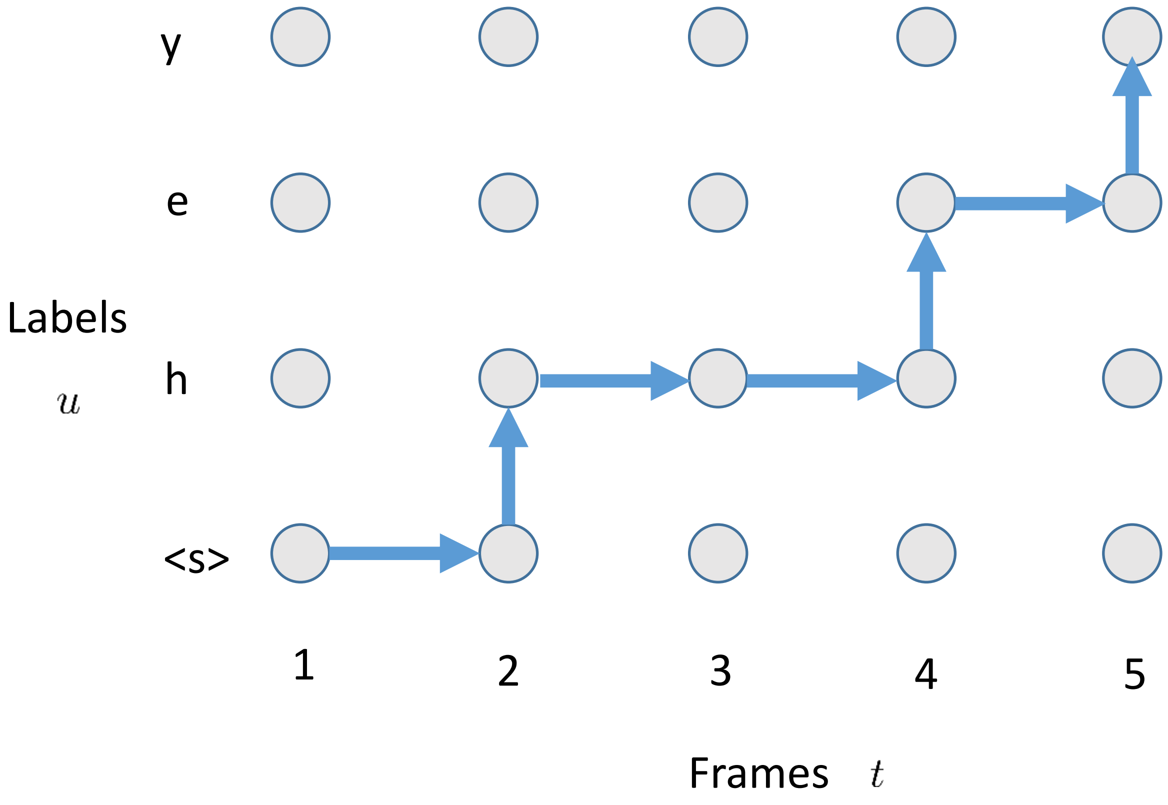
RNN-T Path


$< blank >$
softmax
$z_{t,u}$
Joint
h_u^{pre}
h_t^{enc}
Prediction
Encoder
e
x_5

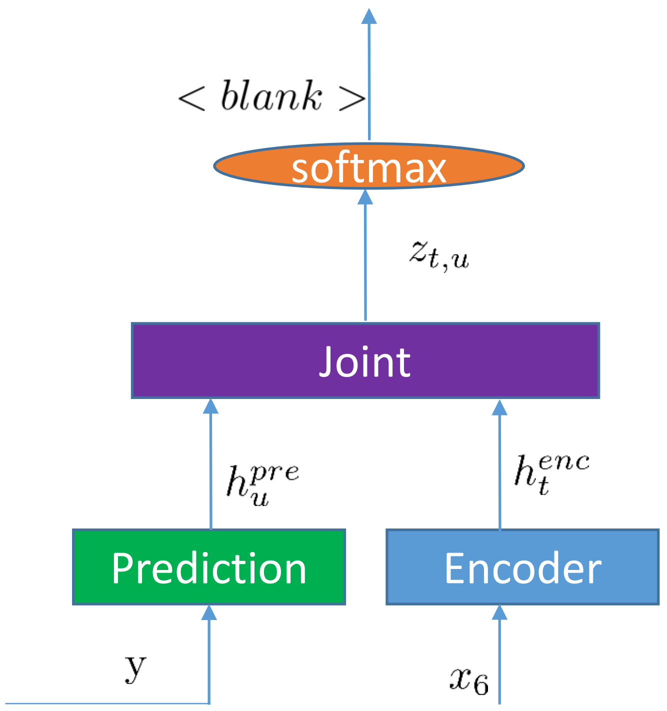

Labels
u
y
e
h
<s>
1
2
3
4
5
Frames t

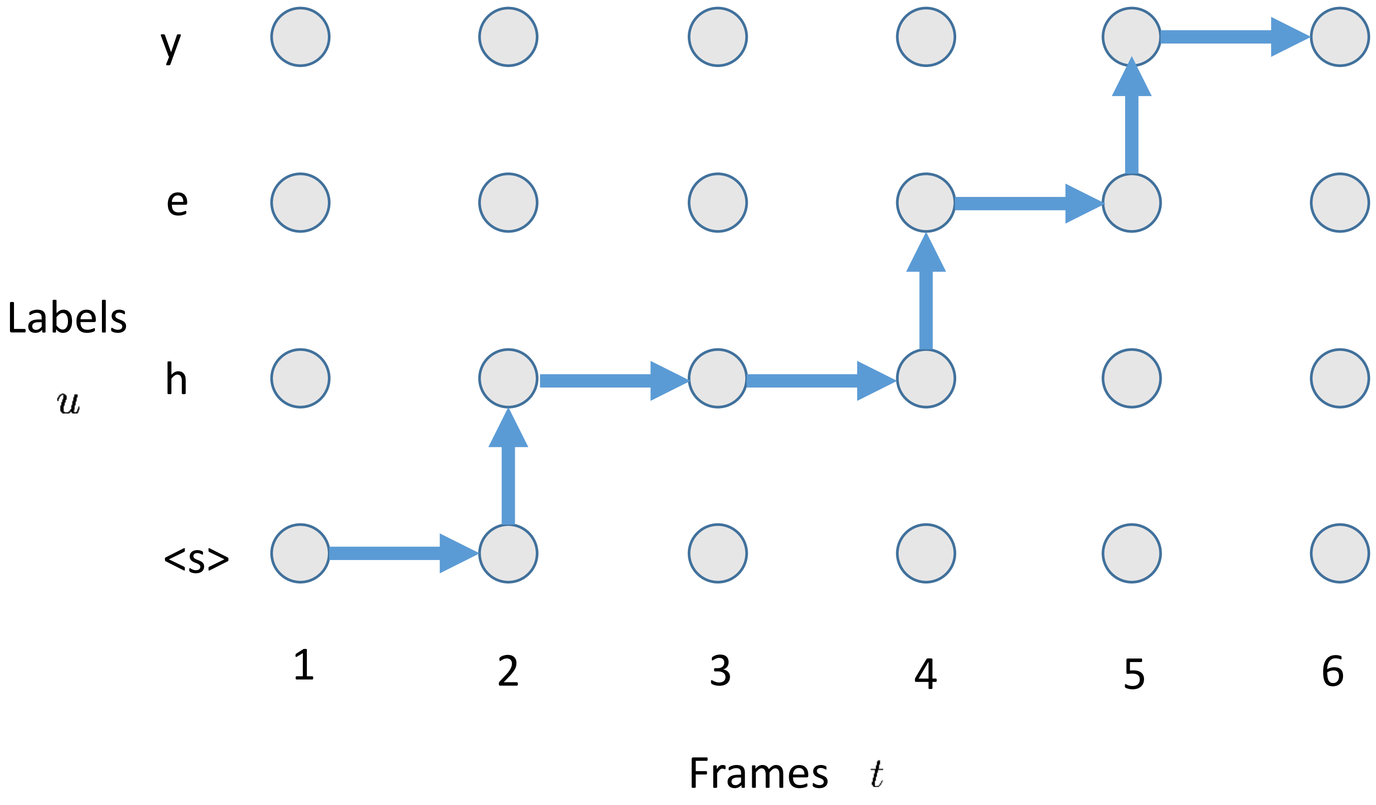
RNN-T Path


y
softmax
$z_{t,u}$
Joint
h_u^{pre}
h_t^{enc}
Prediction
Encoder
e
x_5

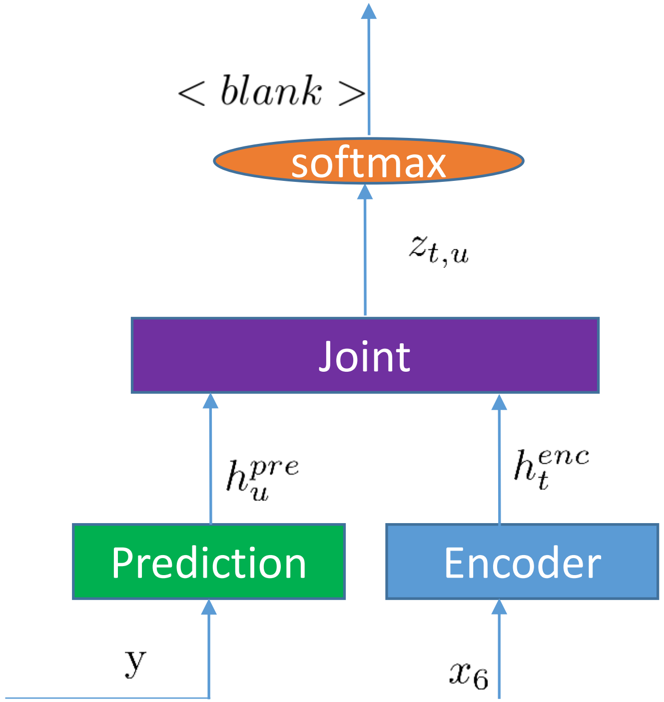

Labels
u
y
e
h
<s>
1
2
3
4
5
Frames t

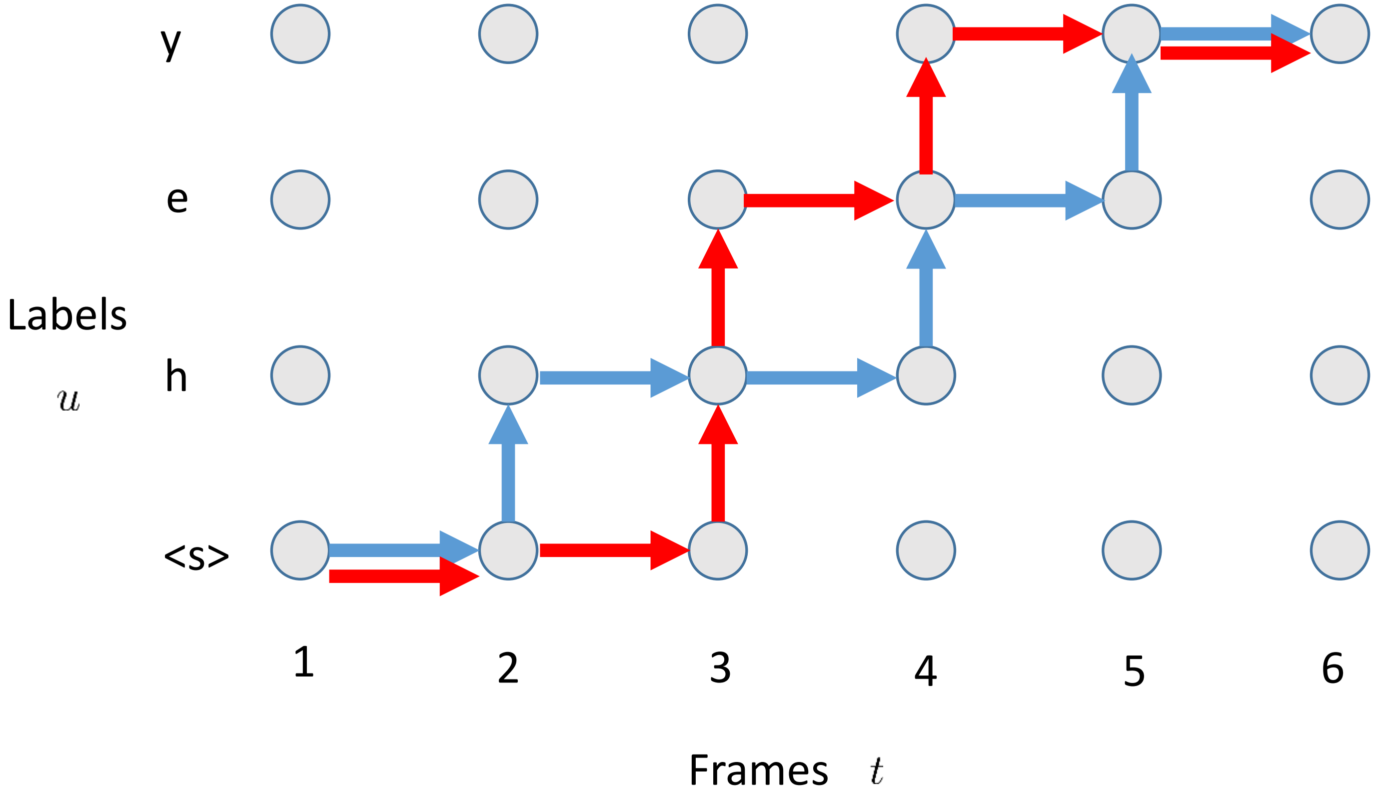
RNN-T Path


$< blank >$
softmax
$z_{t,u}$
Joint
h_u^{pre}
h_t^{enc}
Prediction
Encoder
y
x_6


Labels
u
y
e
h
<s>
1
2
3
4
5
6
Frames t

RNN-T Path

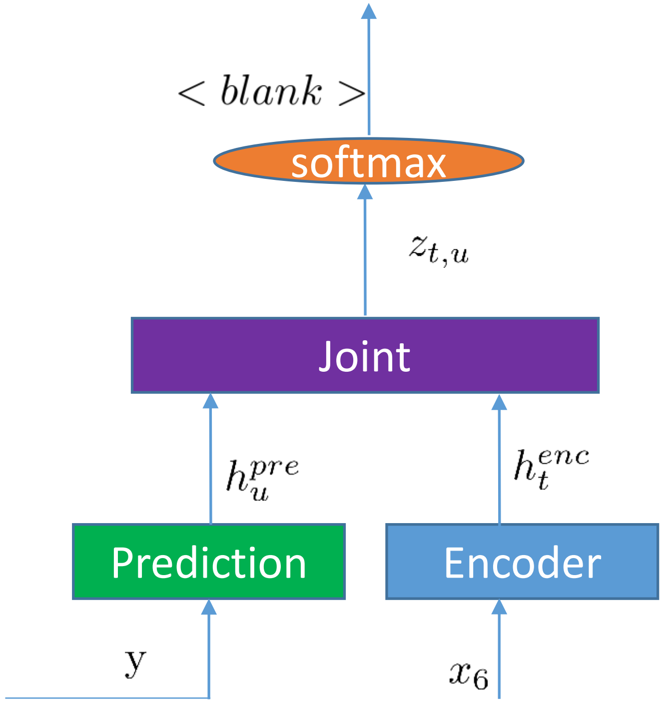


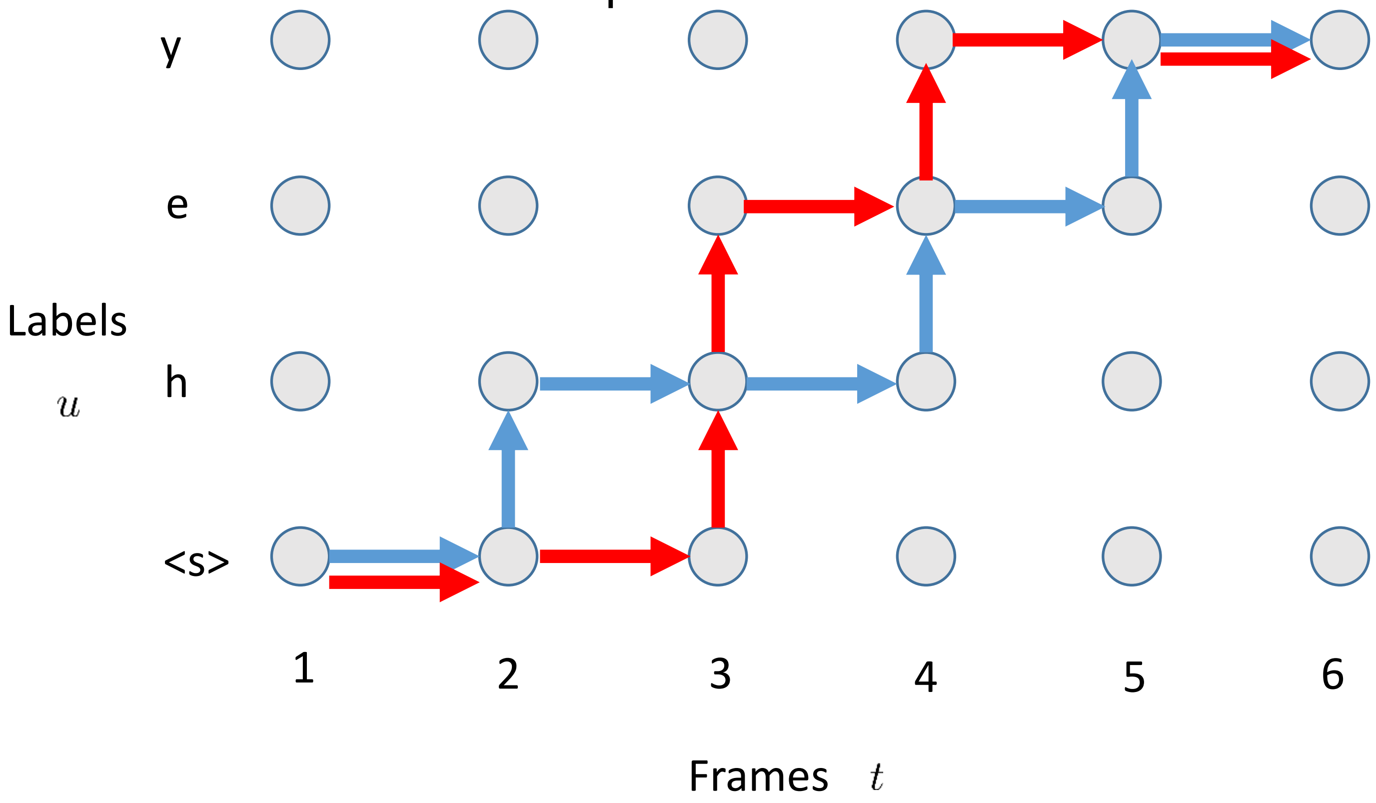


Microsoft

RNN-T Training

Given a label sequence of length U and acoustic frames T, we generate UxT softmax.
The training maximizes the probabilities of all RNN-T paths.





E2E Models

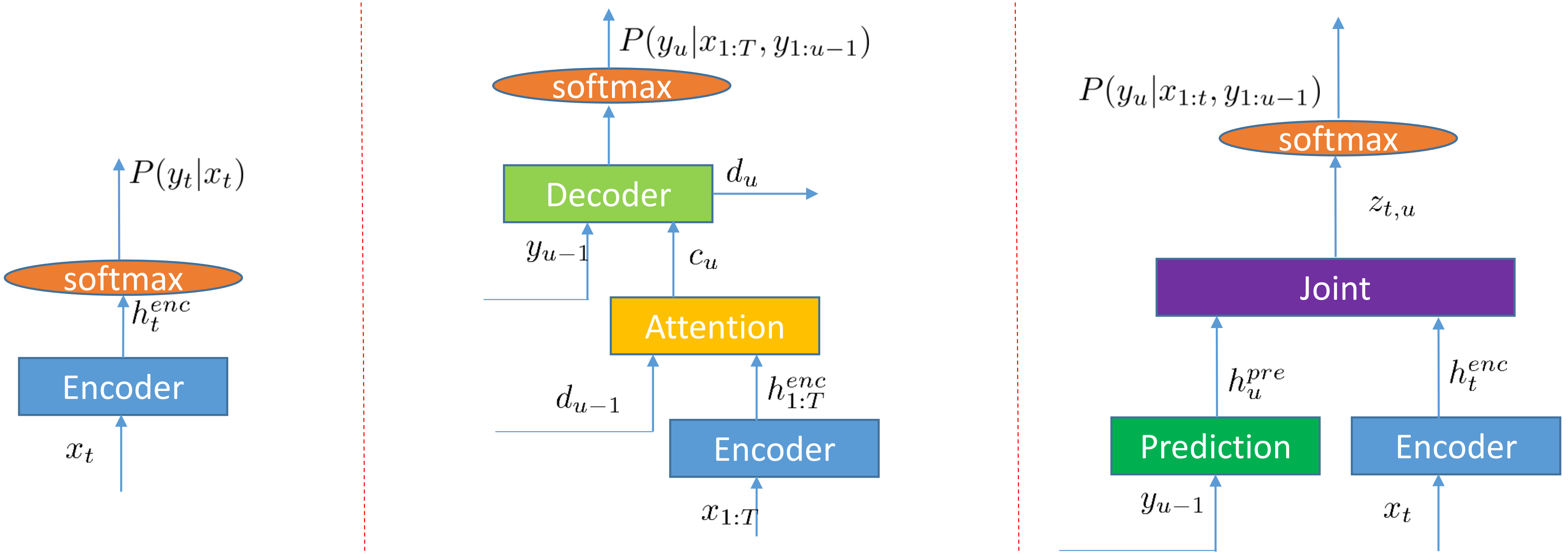
	CTC	AED	RNN-T
Independence assumption	Yes	No	No
Attention mechanism	No	Yes	No
Streaming	Natural	Additional work needed	Natural
Ideal operation scenario	Streaming	Offline	Streaming
Long form capability	Good	Weak	Good

RNN-T is the most popular E2E model in industry which requires streaming ASR.

Sainath et al. "A streaming on-device end-to-end model surpassing server-side conventional model quality and latency" in Proc. ICASSP, 2020
Li et al., "Developing RNN-T Models Surpassing High-Performance Hybrid Models with Customization Capability," *in Proc. Interspeech*, 2020.

E2E Advances -- Encoder

Encoder is the Most Important Component

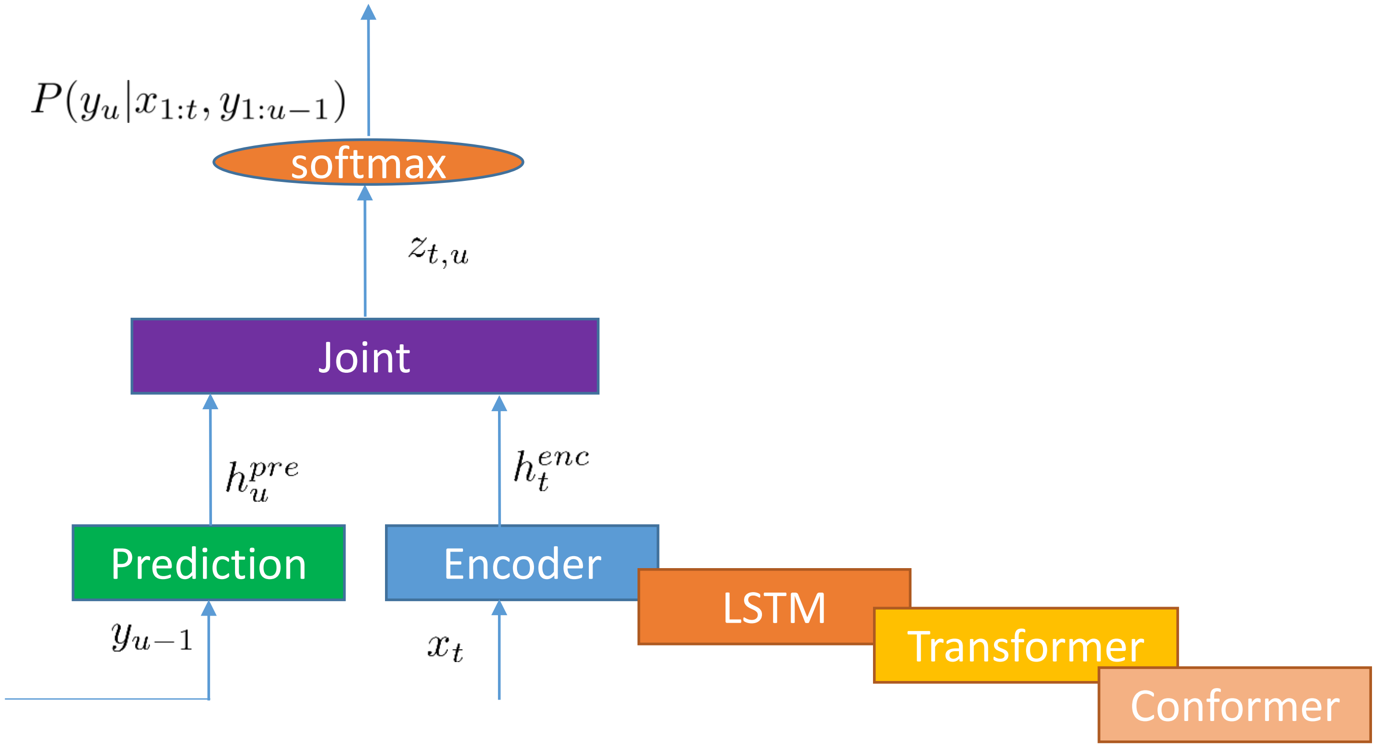


Connectionist Temporal Classification (CTC)

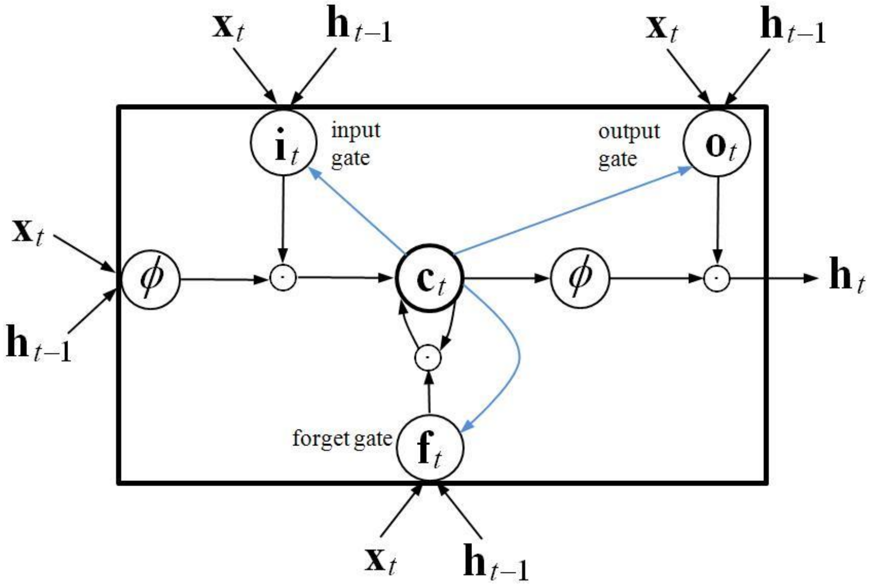
Attention-based encoder decoder (AED)

RNN-Transducer (RNN-T)

Encoder for RNN-T



LSTM Formulations



$$\mathbf{i}_t^l = \sigma(\mathbf{W}_{ix}^l \mathbf{x}_t^l + \mathbf{W}_{ih}^l \mathbf{h}_{t-1}^l + \mathbf{p}_i^l \odot \mathbf{c}_{t-1}^l + \mathbf{b}_i^l)$$

$$\mathbf{f}_t^l = \sigma(\mathbf{W}_{fx}^l \mathbf{x}_t^l + \mathbf{W}_{fh}^l \mathbf{h}_{t-1}^l + \mathbf{p}_f^l \odot \mathbf{c}_{t-1}^l + \mathbf{b}_f^l)$$

$$\mathbf{c}_t^l = \mathbf{f}_t^l \odot \mathbf{c}_{t-1}^l + \mathbf{i}_t^l \odot \phi(\mathbf{W}_{cx}^l \mathbf{x}_t^l + \mathbf{W}_{ch}^l \mathbf{h}_{t-1}^l + \mathbf{b}_c^l)$$

$$\mathbf{o}_t^l = \sigma(\mathbf{W}_{ox}^l \mathbf{x}_t^l + \mathbf{W}_{oh}^l \mathbf{h}_{t-1}^l + \mathbf{p}_o^l \odot \mathbf{c}_t^l + \mathbf{b}_o^l)$$

$$\mathbf{h}_t^l = \mathbf{o}_t^l \odot \phi(\mathbf{c}_t^l)$$

$$\mathbf{x}_t^l = \begin{cases} \mathbf{h}_t^{l-1}, & \text{if } l > 1 \\ \mathbf{s}_t, & \text{if } l = 1 \end{cases}$$

Hochreiter and Schmidhuber. "Long short-term memory," in Neural computation, 1997.

Transformer

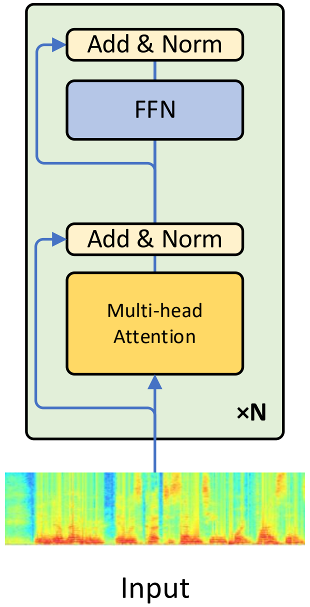
- Self-attention: computes the attention distribution over the input speech sequence

$$\alpha_{t,\tau} = \frac{\exp(\beta(\mathbf{W}_q\mathbf{x}_t)^T(\mathbf{W}_k\mathbf{x}_\tau))}{\sum_{\tau'}\exp(\beta(\mathbf{W}_q\mathbf{x}_t)^T(\mathbf{W}_k\mathbf{x}_{\tau'}))}$$

- Attention weights are used to combine the value vectors to generate the layer output

$$\mathbf{z}_t = \sum_\tau \alpha_{t\tau}\mathbf{W}_v\mathbf{x}_\tau = \sum_\tau \alpha_{t\tau}\mathbf{v}_\tau$$

- Multi-head self-attention: applies multiple parallel self-attentions on the input sequence

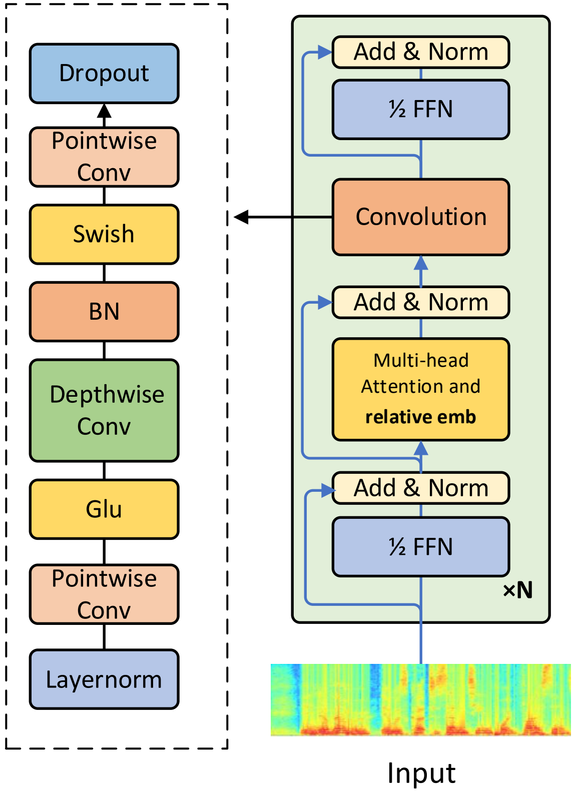


Vaswani et al. "Attention is all you need" NIPS 2017
Zhang et al., "Transformer Transducer: A Streamable Speech Recognition Model with Transformer Encoders and RNN-T Loss," in Proc. ICASSP 2020.

Microsoft

Conformer

- Transformer: good at capturing global context, but less effective in extracting local patterns
- Convolutional neural network (CNN): works on local information
- Conformer: combines Transformer with CNN



Gulati et al. “Conformer: Convolution-augmented Transformer for Speech Recognition,” in Proc. Interspeech, 2020.

Industry Requirement of Transformer Encoder

- Streaming with low latency and low computational cost
- Vanilla Transformer fails so because it attends the full sequence
- Solution: Attention mask is all you need

Attention Mask is All You Need

- Compute attention weight $\{\alpha_{t,\tau}\}$ for time t over input sequence $\{\boldsymbol{x}_\tau\}$, binary attention mask $\{m_{t,\tau}\}$ to control range of input $\{\mathbf{x}_\tau\}$ to use

$$\alpha_{t,\tau} = \frac{\mathrm{m}_{t,\tau}\exp(\beta\left(W_q\boldsymbol{x}_t\right)^T(W_k\boldsymbol{x}_\tau))}{\sum_{\tau'}\mathrm{m}_{t,\tau'}\exp(\beta\left(W_q\boldsymbol{x}_t\right)^T(W_k x_{\tau'}))} = softmax(\beta\boldsymbol{q}_t^T\boldsymbol{k}_\tau, m_{t,\tau})$$

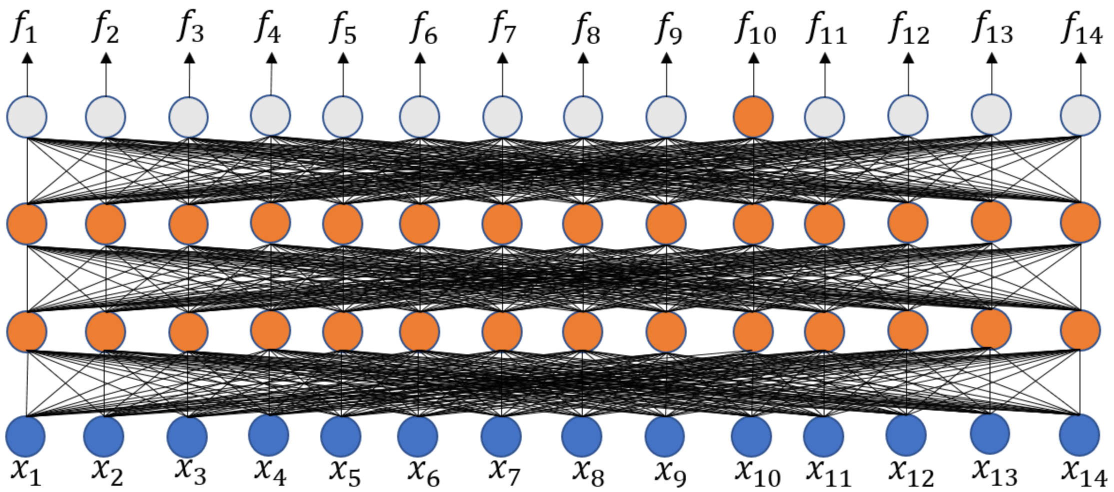
- Apply attention weight over value vector $\{\boldsymbol{v}_\tau\}$

$$z_t = \sum_\tau \alpha_{t,\tau}W_v\boldsymbol{x}_\tau = \sum_\tau \alpha_{t,\tau}\boldsymbol{v}_\tau$$

Chen et al., "Developing real-time streaming transformer transducer for speech recognition on large-scale dataset," in Proc. ICASSP 2021.

Attention Mask is All You Need

- Offline (whole utterance)



Predicting output for x_{10}

Frame Index

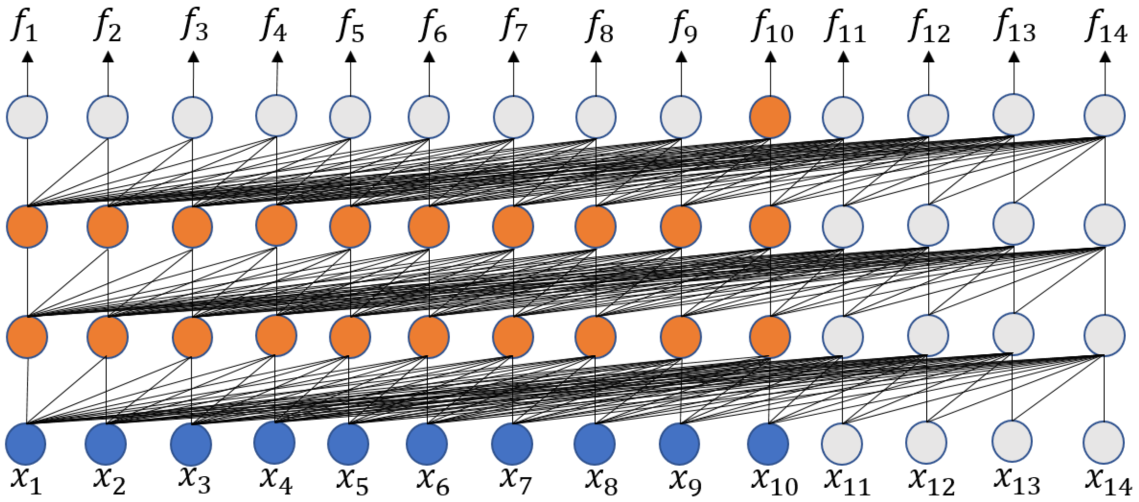
$$
\begin{array}{c}
1\\2\\3\\4\\5\\6\\7\\8\\9\\10\\11\\12\\13\\14
\end{array}
\begin{bmatrix}
1&1&1&1&1&1&1&1&1&1&1&1&1&1\\
1&1&1&1&1&1&1&1&1&1&1&1&1&1\\
1&1&1&1&1&1&1&1&1&1&1&1&1&1\\
1&1&1&1&1&1&1&1&1&1&1&1&1&1\\
1&1&1&1&1&1&1&1&1&1&1&1&1&1\\
1&1&1&1&1&1&1&1&1&1&1&1&1&1\\
1&1&1&1&1&1&1&1&1&1&1&1&1&1\\
1&1&1&1&1&1&1&1&1&1&1&1&1&1\\
1&1&1&1&1&1&1&1&1&1&1&1&1&1\\
1&1&1&1&1&1&1&1&1&1&1&1&1&1\\
1&1&1&1&1&1&1&1&1&1&1&1&1&1\\
1&1&1&1&1&1&1&1&1&1&1&1&1&1\\
1&1&1&1&1&1&1&1&1&1&1&1&1&1\\
1&1&1&1&1&1&1&1&1&1&1&1&1&1
\end{bmatrix}.
$$

Not streamable

Attention Mask

Attention Mask is All You Need

- 0 lookahead, full history



Predicting output for x_{10}

Frame Index

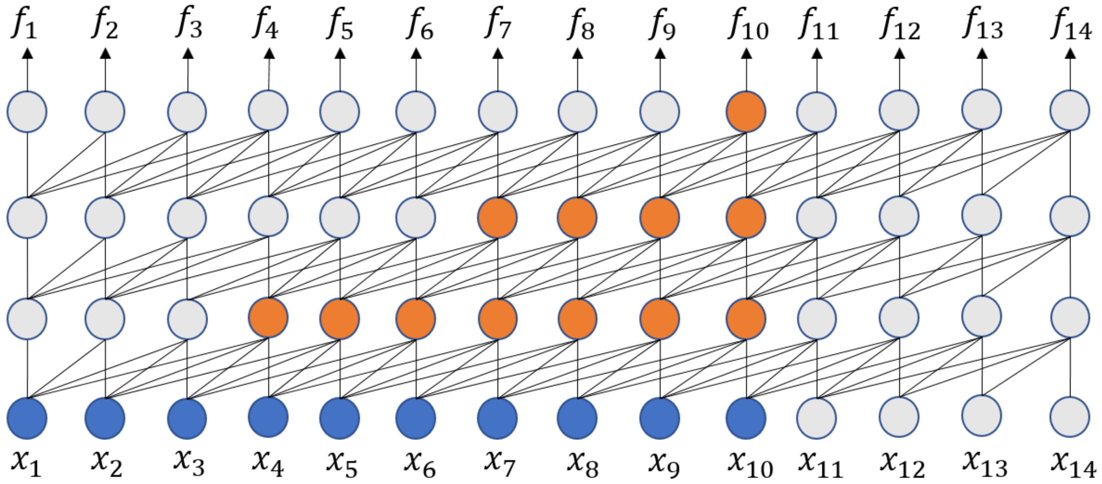
$$\begin{array}{r}1\\2\\3\\4\\5\\6\\7\\8\\9\\10\\11\\12\\13\\14\end{array}\begin{bmatrix}1&0&0&0&0&0&0&0&0&0&0&0&0&0\\1&1&0&0&0&0&0&0&0&0&0&0&0&0\\1&1&1&0&0&0&0&0&0&0&0&0&0&0\\1&1&1&1&0&0&0&0&0&0&0&0&0&0\\1&1&1&1&1&0&0&0&0&0&0&0&0&0\\1&1&1&1&1&1&0&0&0&0&0&0&0&0\\1&1&1&1&1&1&1&0&0&0&0&0&0&0\\1&1&1&1&1&1&1&1&0&0&0&0&0&0\\1&1&1&1&1&1&1&1&1&0&0&0&0&0\\1&1&1&1&1&1&1&1&1&1&0&0&0&0\\1&1&1&1&1&1&1&1&1&1&1&0&0&0\\1&1&1&1&1&1&1&1&1&1&1&1&0&0\\1&1&1&1&1&1&1&1&1&1&1&1&1&0\\1&1&1&1&1&1&1&1&1&1&1&1&1&1\end{bmatrix}.$$

Attention Mask

Memory and runtime cost increase linearly

Attention Mask is All You Need

- 0 lookahead, limited history (3 frames)



Predicting output for x_{10}

Frame Index

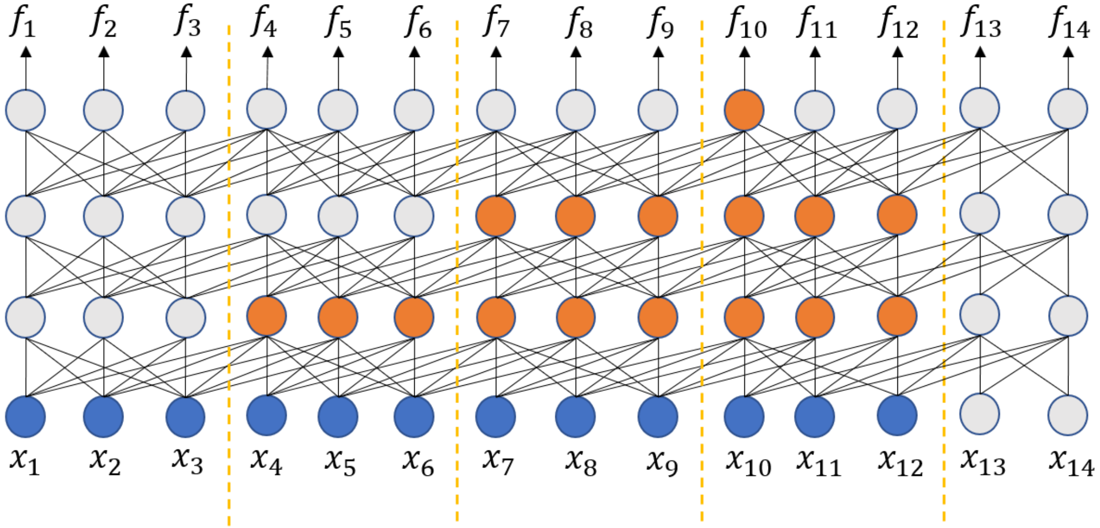
$$
\begin{array}{c}
1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \\ 10 \\ 11 \\ 12 \\ 13 \\ 14
\end{array}
\begin{bmatrix}
1&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&1&0&0&0&0&0&0&0&0&0&0&0&0\\
1&1&1&0&0&0&0&0&0&0&0&0&0&0\\
1&1&1&1&0&0&0&0&0&0&0&0&0&0\\
0&1&1&1&1&0&0&0&0&0&0&0&0&0\\
0&0&1&1&1&1&0&0&0&0&0&0&0&0\\
0&0&0&1&1&1&1&0&0&0&0&0&0&0\\
0&0&0&0&1&1&1&1&0&0&0&0&0&0\\
0&0&0&0&0&1&1&1&1&0&0&0&0&0\\
0&0&0&0&0&0&1&1&1&1&0&0&0&0\\
0&0&0&0&0&0&0&1&1&1&1&0&0&0\\
0&0&0&0&0&0&0&0&1&1&1&1&0&0\\
0&0&0&0&0&0&0&0&0&1&1&1&1&0\\
0&0&0&0&0&0&0&0&0&0&1&1&1&1
\end{bmatrix}.
$$

Attention Mask

In some scenario, small amount of latency is allowed

Attention Mask is All You Need

- Small lookahead (at most 2 frames), limited history (3 frames)



Predicting output for x_{10}

Frame Index

$$
\begin{array}{c}
1\\2\\3\\4\\5\\6\\7\\8\\9\\10\\11\\12\\13\\14
\end{array}
\begin{bmatrix}
1&1&1&0&0&0&0&0&0&0&0&0&0&0\\
1&1&1&0&0&0&0&0&0&0&0&0&0&0\\
1&1&1&0&0&0&0&0&0&0&0&0&0&0\\
1&1&1&1&1&1&0&0&0&0&0&0&0&0\\
1&1&1&1&1&1&0&0&0&0&0&0&0&0\\
1&1&1&1&1&1&0&0&0&0&0&0&0&0\\
0&0&0&1&1&1&1&1&1&0&0&0&0&0\\
0&0&0&1&1&1&1&1&1&0&0&0&0&0\\
0&0&0&1&1&1&1&1&1&0&0&0&0&0\\
0&0&0&0&0&0&1&1&1&1&1&1&0&0\\
0&0&0&0&0&0&1&1&1&1&1&1&0&0\\
0&0&0&0&0&0&1&1&1&1&1&1&0&0\\
0&0&0&0&0&0&0&0&0&1&1&1&1&1\\
0&0&0&0&0&0&0&0&0&1&1&1&1&1
\end{bmatrix}.
$$

Look-ahead window [0, 2]

Attention Mask

E2E Advances -- Multilingual

Multilingual

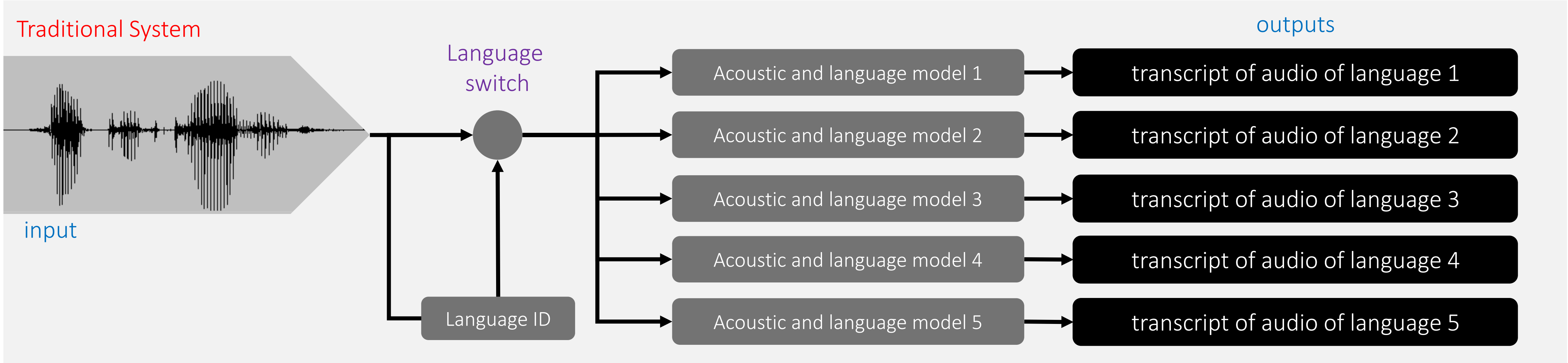
- % people can speak only 1 language fluently.
- % people can speak only 2 languages fluently.
- % people can speak only 3 languages fluently.
- % people can speak only 4 languages fluently.
- % people can speak 5+ languages fluently.

Multilingual

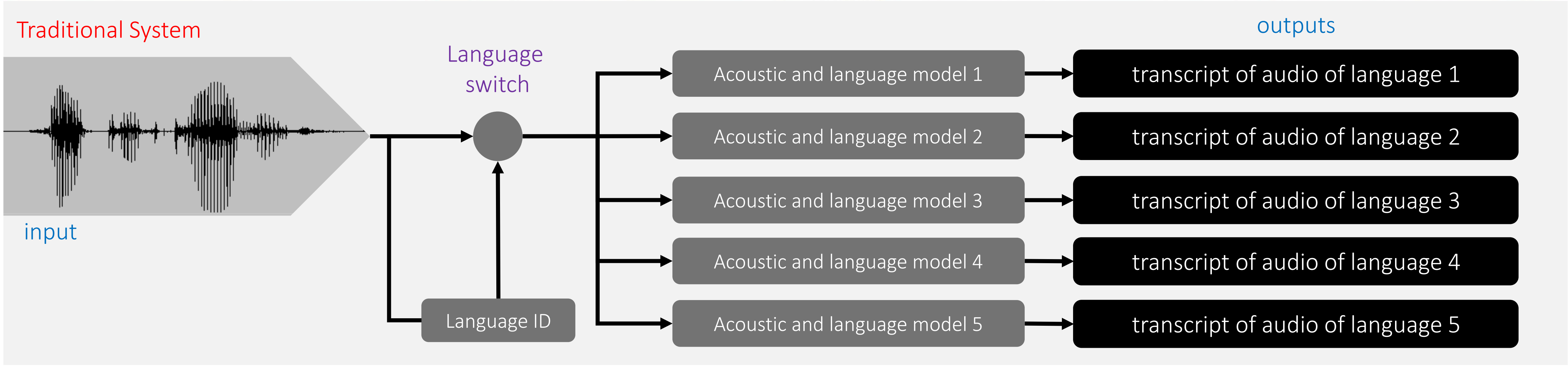
- 40% people can speak only 1 language fluently.
- 43% people can speak only 2 languages fluently.
- 13% people can speak only 3 languages fluently.
- 3% people can speak only 4 languages fluently.
- <0.1% people can speak 5+ languages fluently.

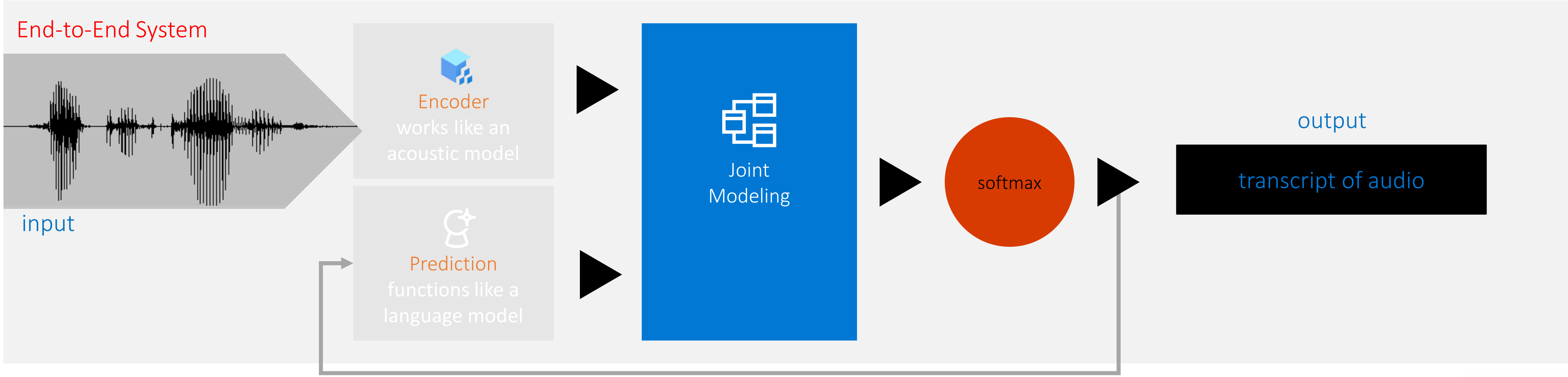
- Human cannot recognize all languages. Can we build a *single high quality multilingual model on device* to serve *all users*?

Traditional System



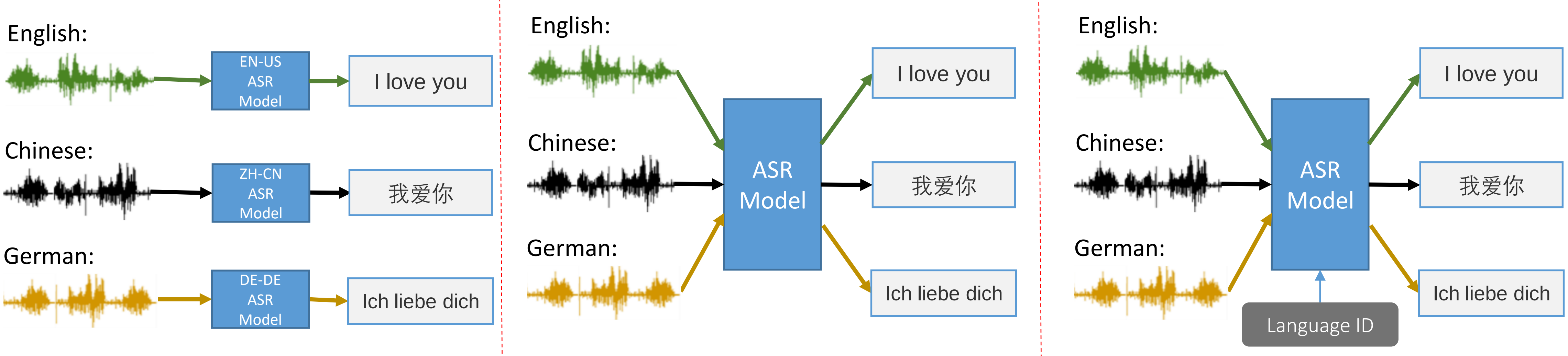
- System size scales up linearly with the number of languages
- Heavily depends on LID, which introduces obvious latency


Traditional System
outputs
Language switch
input
Acoustic and language model 1
Acoustic and language model 2
Acoustic and language model 3
Acoustic and language model 4
Acoustic and language model 5
transcript of audio of language 1
transcript of audio of language 2
transcript of audio of language 3
transcript of audio of language 4
transcript of audio of language 5
Language ID


End-to-End System
input
Encoder
works like an
acoustic model
Prediction
functions like a
language model
Joint
Modeling
softmax
output
transcript of audio

Multilingual E2E Models

- Double-edged sword of pooling all language data
  - Maximum sharing between languages; One model for all languages
  - Confusion between languages, which can be addressed with a one-hot LID input.
    - Multilingual w/ LID is more like a monolingual model with the requirement of prior knowledge of language to speak.



Multiple monolingual ASR | Universal multilingual ASR w/o LID | Universal multilingual ASR w/ LID

Watanabe et al., "Language independent end-to-end architecture for joint language identification and speech recognition," *in Proc. ASRU*, 2017.
Kim and Seltzer, "Towards language-universal end-to-end speech recognition," *in Proc. ICASSP*, 2018.
Toshniwal et al., "Multilingual speech recognition with a single end-to-end model," *in Proc. ICASSP*, 2018.

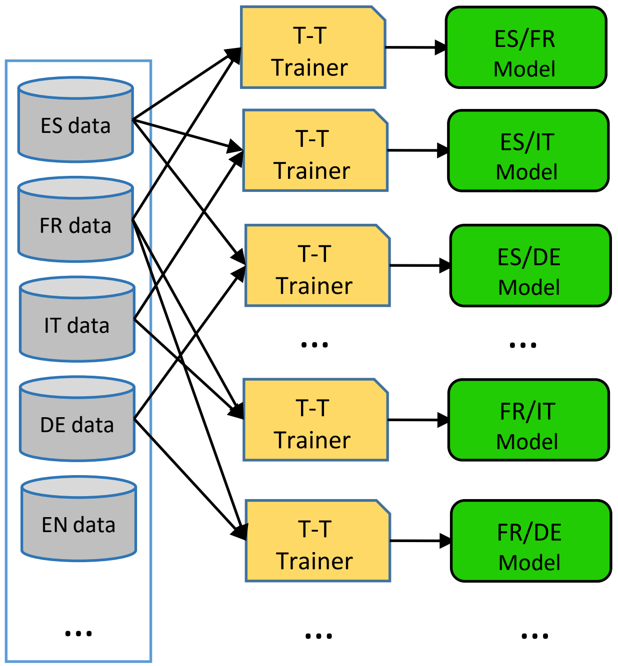
Multilingual E2E Models

- Large gap between models w/o LID and w/ LID.
- The gap keeps increasing when building the model with more languages.

Language	Monolingual Baseline	Multilingual w/o 1-hot LID	Multilingual w 1-hot LID
EN	9.52	10.72	10.50
ES	19.98	19.83	16.07
FR	21.58	27.02	17.43
IT	19.67	21.59	15.30
PL	19.39	23.99	13.69
PT	14.58	14.14	13.01
NL	20.74	24.41	17.70
DE	16.26	18.16	16.24
RO	14.91	15.56	14.62
EL	17.63	17.83	17.43
AVE	17.22	19.32	15.20

Specific Model for Every Combination of Languages?

- Development cost is formidable

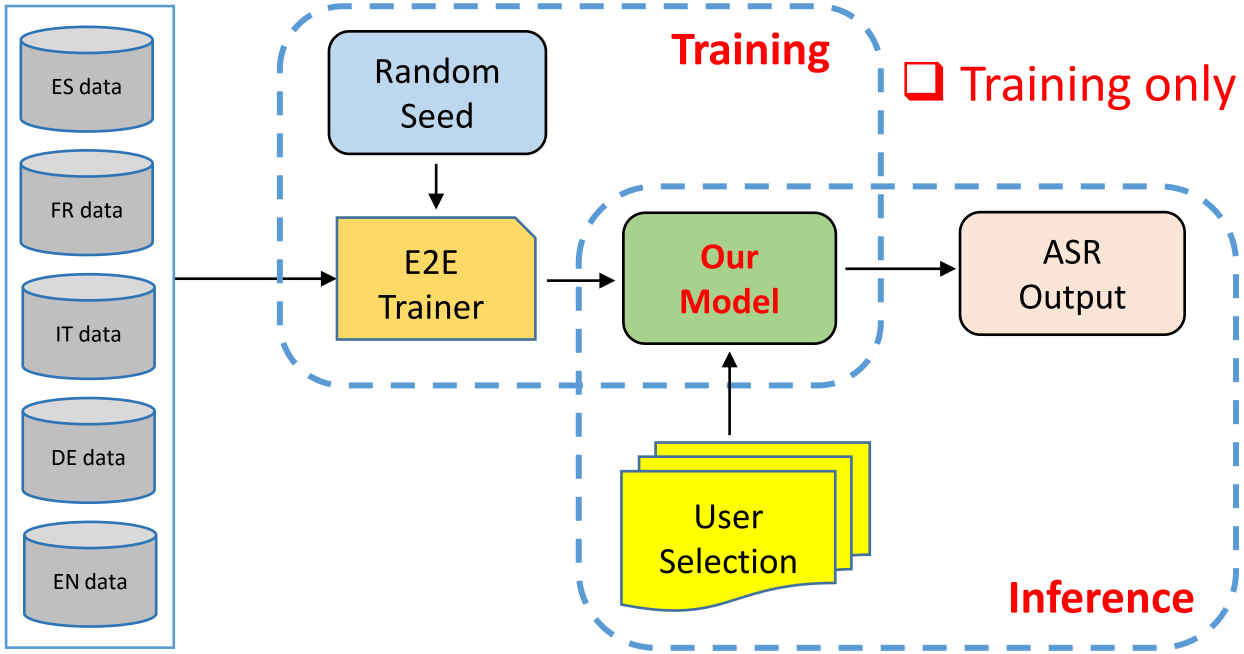


$\sum_{1 \le m \le n} C_n^m$ for n languages

C_{10}^1=10, C_{10}^2=45, C_{10}^3=120

How to Deal with Multilingual Speakers?

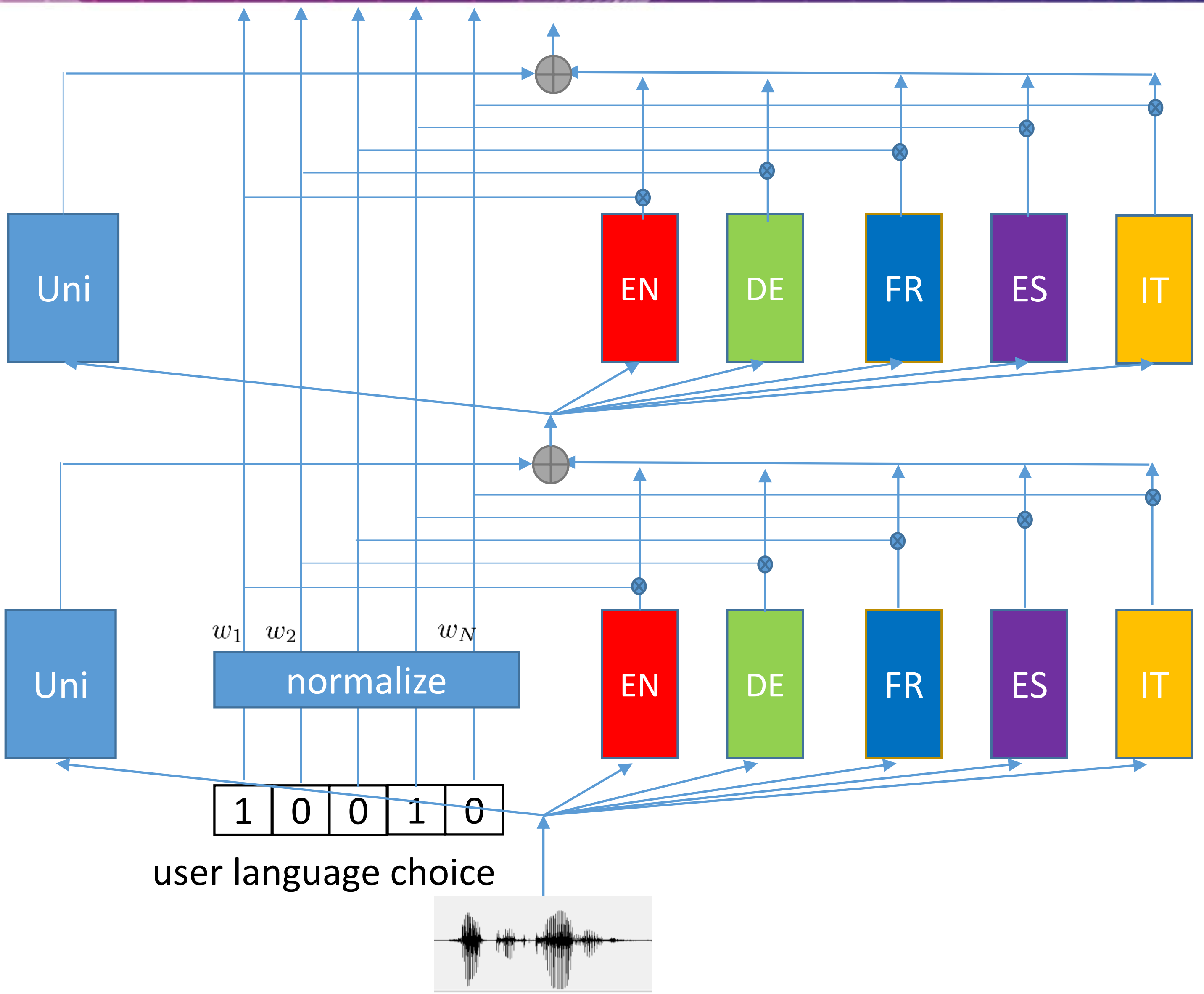
Configurable Multilingual ASR



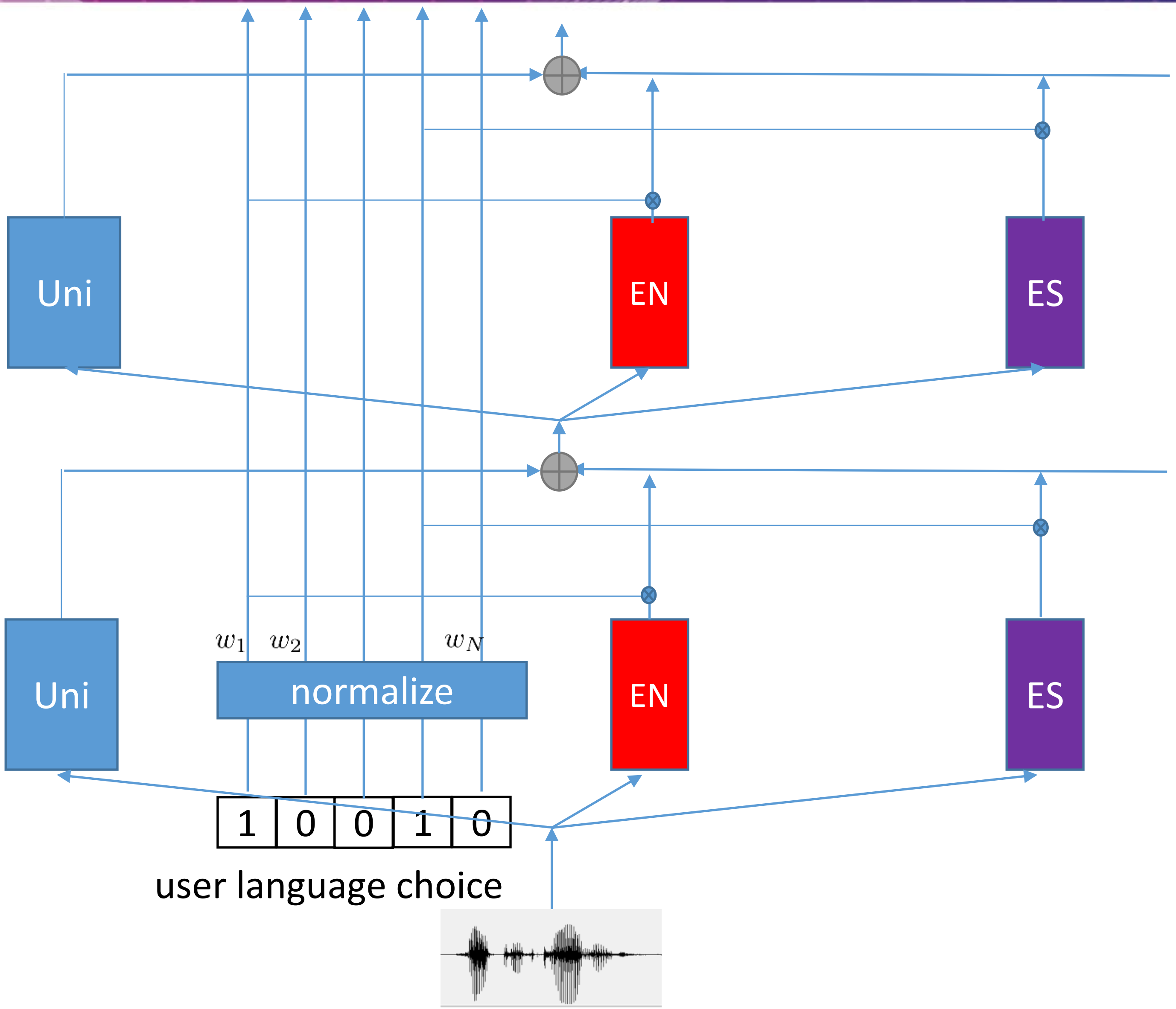
- ❑ Training only once
- ❑ Configured as different models

Zhou et al., "A Configurable Multilingual Model is All You Need to Recognize All Languages," *arXiv preprint* 2021.

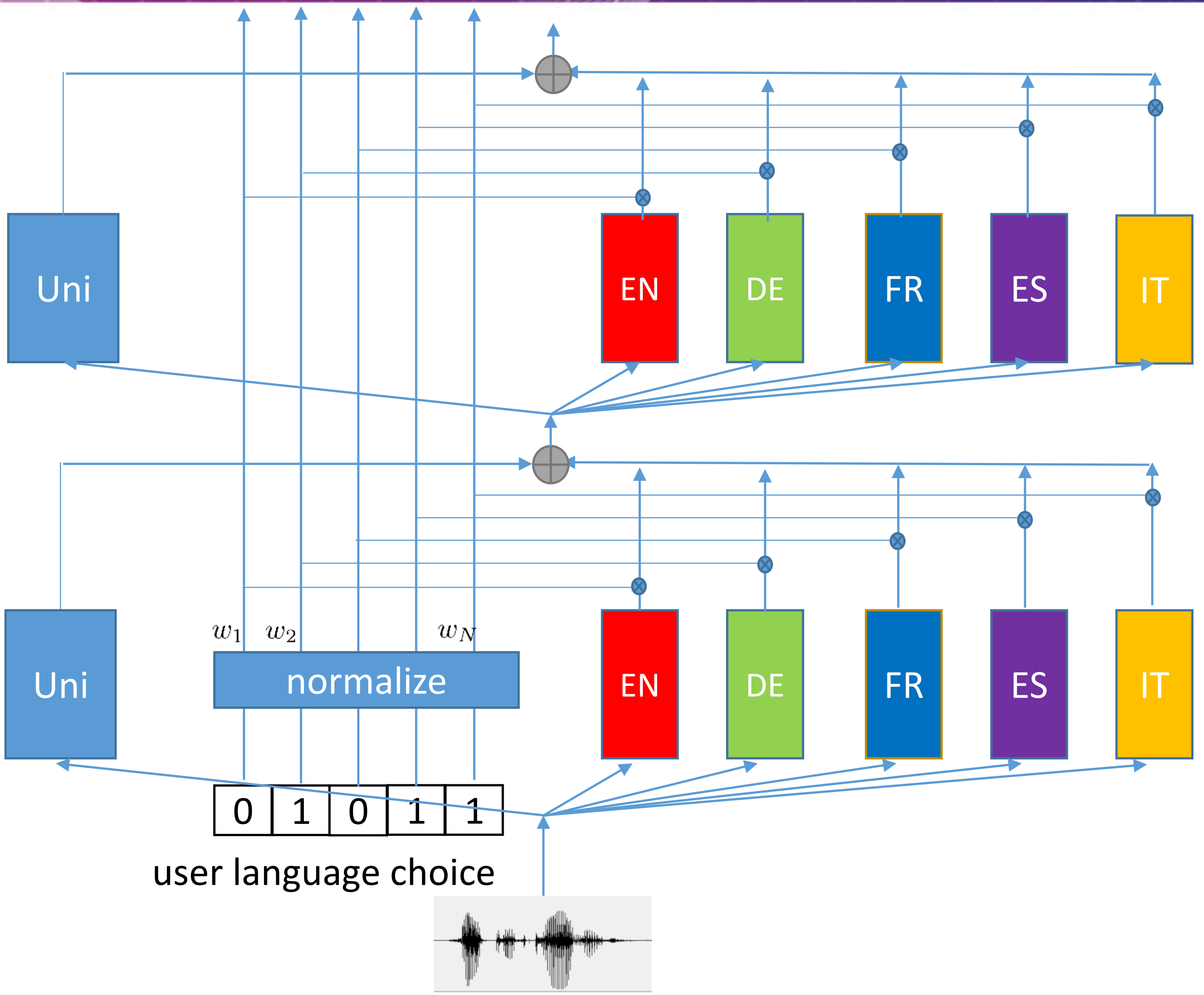
- **Universal module**: modeling the sharing across languages
- **Expert module**: modeling the residual from universal module for each language



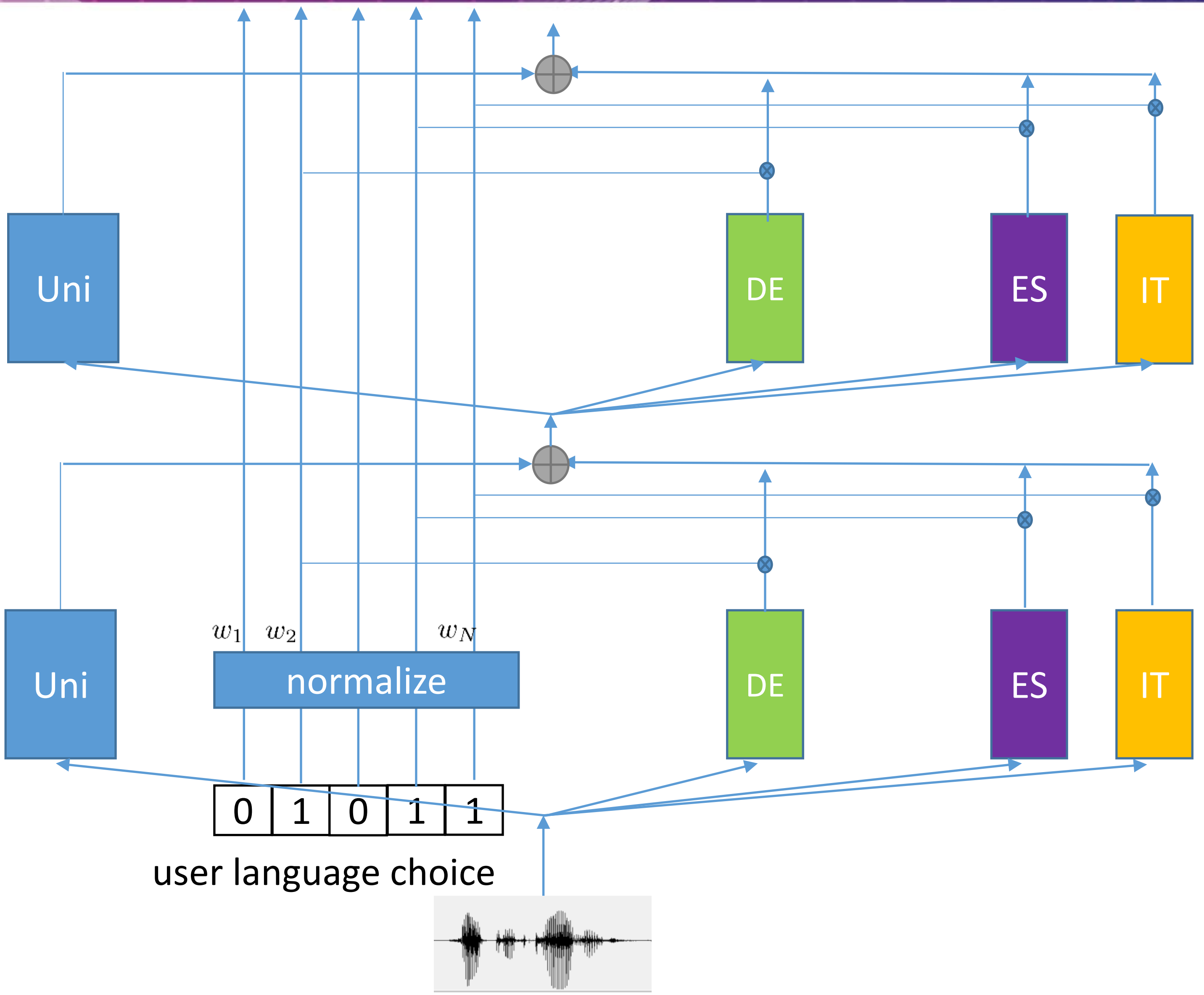
- **Universal module**: modeling the sharing across languages
- **Expert module**: modeling the residual from universal module for each language



- **Universal module**: modeling the sharing across languages
- **Expert module**: modeling the residual from universal module for each language



- **Universal module**: modeling the sharing across languages
- **Expert module**: modeling the residual from universal module for each language



E2E Advances -- Adaptation

Adaptation

- **Speaker adaptation**: adapts ASR models to better recognize a target speaker's speech.
- **Domain adaptation**: adapts ASR models to the target domain which has content mismatch from the source domain.
- **Customization**: leverages context such as contacts, location, music play list etc., of a specific user to significantly boost the ASR accuracy for this user.

Speaker Adaptation

- The biggest challenge: the adaptation data amount from the target speaker is usually very small.

- Solutions:
  - regularization techniques such as Kullback-Leibler (KL) divergence, maximum a posteriori adaptation, or elastic weight consolidation
  - multi-task learning: auxiliary task with a small number of output tokens
  - multi-speaker text-to-speech (TTS) to expand the adaption set for the speaker

Li et al., "Speaker adaptation for end-to-end CTC models," *in Proc. SLT,* 2018.
Sim et al., "Personalization of end-to-end speech recognition on mobile devices for named entities," *in Proc. ASRU,* 2019.
Huang et al., "Rapid RNN-T Adaptation Using Personalized Speech Synthesis and Neural Language Generator," *in Proc. Interspeech*, 2020.

Domain Adaptation

- The biggest challenge: not easy to get enough paired speech-text data in the new domain.

- Solution: utilize the new domain text only.
  - LM fusion: fusing E2E models with an external LM trained with the new domain text data.
  - Bayesian methods: remove internal LM contribution when fusing with an external LM.

Toshniwal et al., "A comparison of techniques for language model integration in encoder-decoder speech recognition" *in Proc. SLT*, 2018.
Kanda et al., "Maximum a posteriori Based Decoding for CTC Acoustic Models," *in Proc. Interspeech*, 2016.
McDermott et al., "A density ratio approach to language model fusion in end-to-end automatic speech recognition," *in Proc. ASRU*, 2019.
Meng et al., "Internal language model estimation for domain-adaptive end-to-end speech recognition," in Proc. SLT, 2021.

Microsoft

TTS for Domain Adaptation

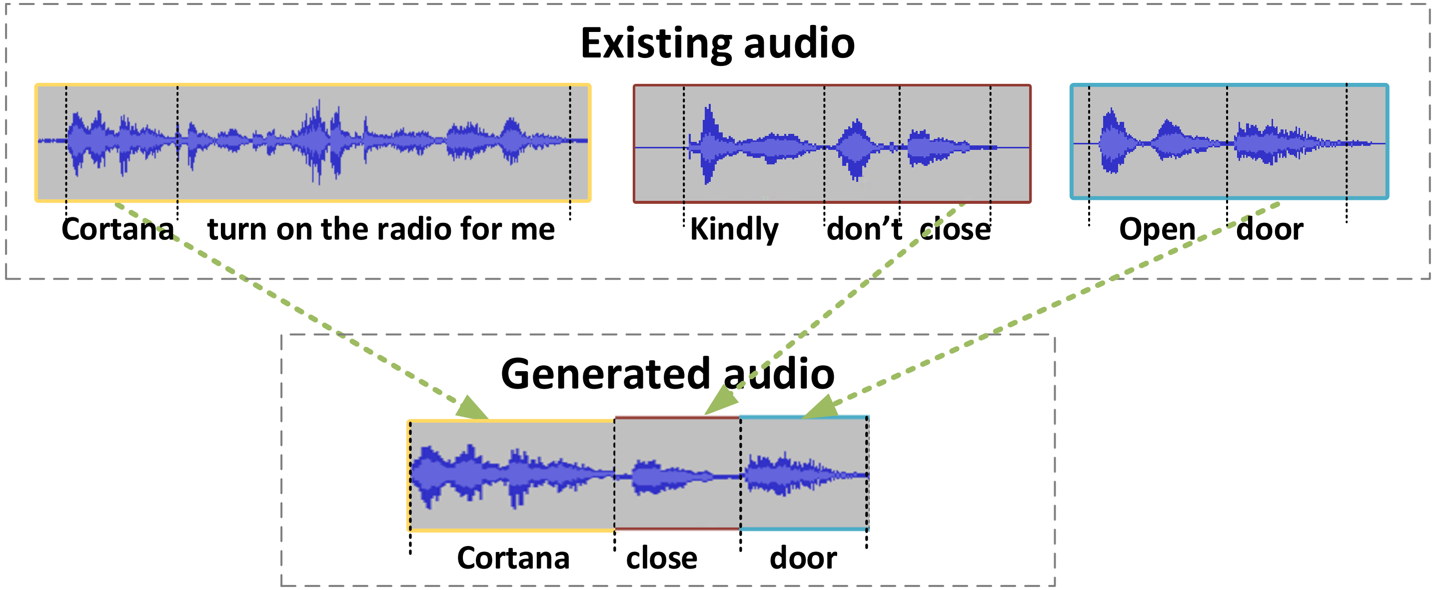
- Adapt E2E models with the synthesized speech generated from the new domain text without the need of LM fusion.

- Drawbacks:
  - TTS speech is different from the real speech. It sometimes also degrades the recognition accuracy on real speech.
  - The speaker variation in the TTS data is far less than that in the large-scale ASR training data.
  - The cost of training a multi-speaker TTS model and the generation of synthesized speech from the model is large.

Sim et al., “Personalization of end-to-end speech recognition on mobile devices for named entities,” *in Proc. ASRU*, 2019.
Li et al., “Developing RNN-T Models Surpassing High-Performance Hybrid Models with Customization Capability,” *in Proc. Interspeech*, 2020.
Zheng et al., “Using synthetic audio to improve the recognition of out-of-vocabulary words in end-to-end ASR systems,” *in Proc. ICASSP*, 2021.

Data Splicing for Domain Adaptation

- Generate new audio from original ASR training data.

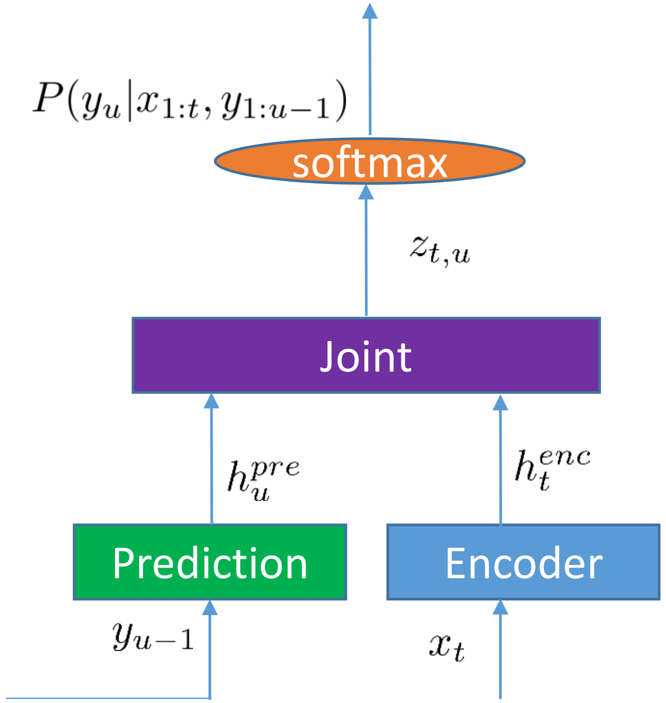


Zhao et al., "On Addressing Practical Challenges for RNN-Transducer," *in Proc. ASRU*, 2021.

Is the Prediction Network a LM?

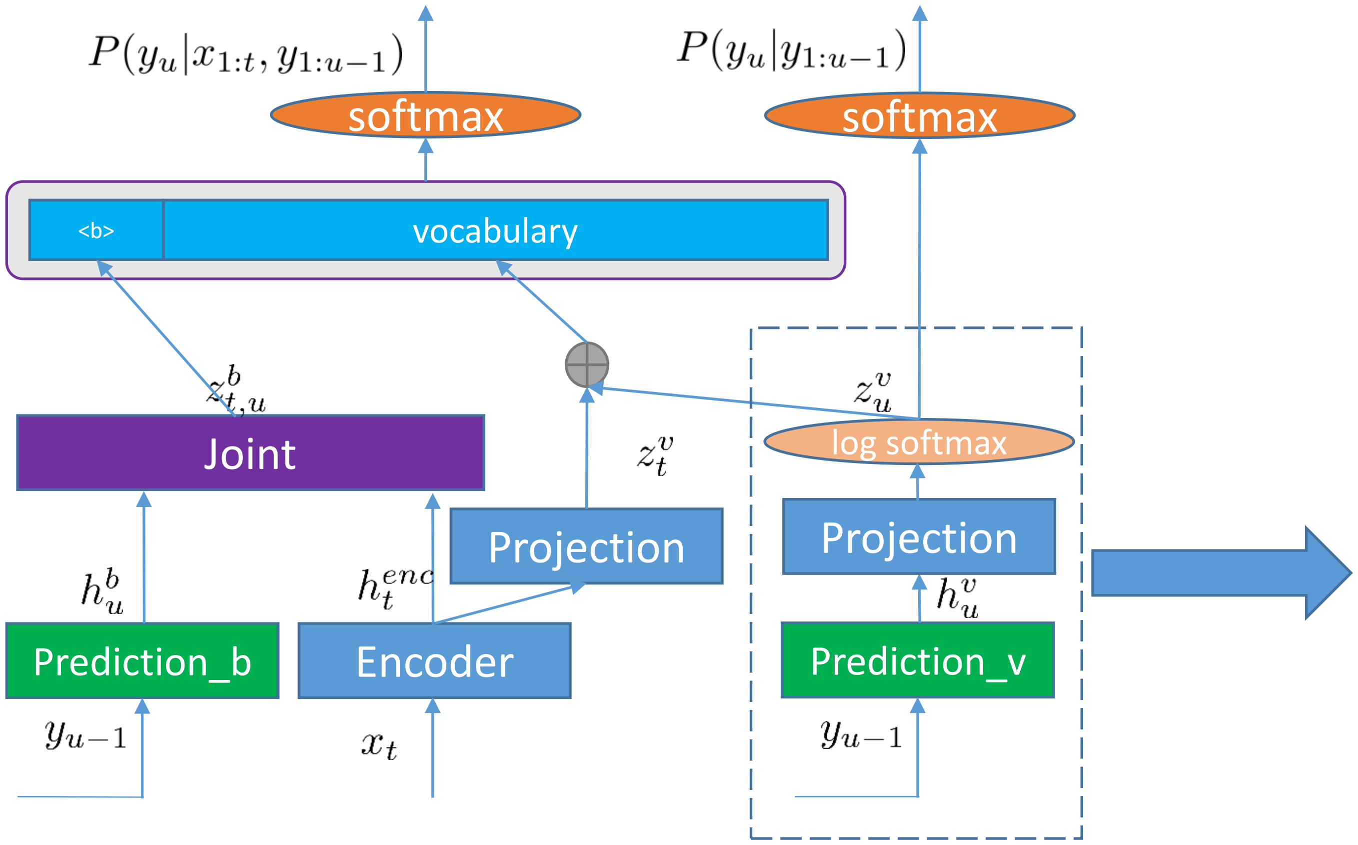
- If the prediction network in RNN-T is a LM, we can use new-domain text to adapt it without even bothering audio data generation.

- However, it does not fully function as a LM because it needs to predict both normal tokens and blank.



Ghodsi et al., "RNN-transducer with stateless prediction network," *in Proc. ICASSP*, 2020.
Chen et al., "Factorized Neural Transducer for Efficient Language Model Adaptation," *in arXiv preprint*, 2021.

Factorized Neural Transducer



Functions as a neural LM. Can be adapted with text only data!

Chen et al., "A Configurable Multilingual Model is All You Need to Recognize All Languages," *arXiv preprint* 2021.

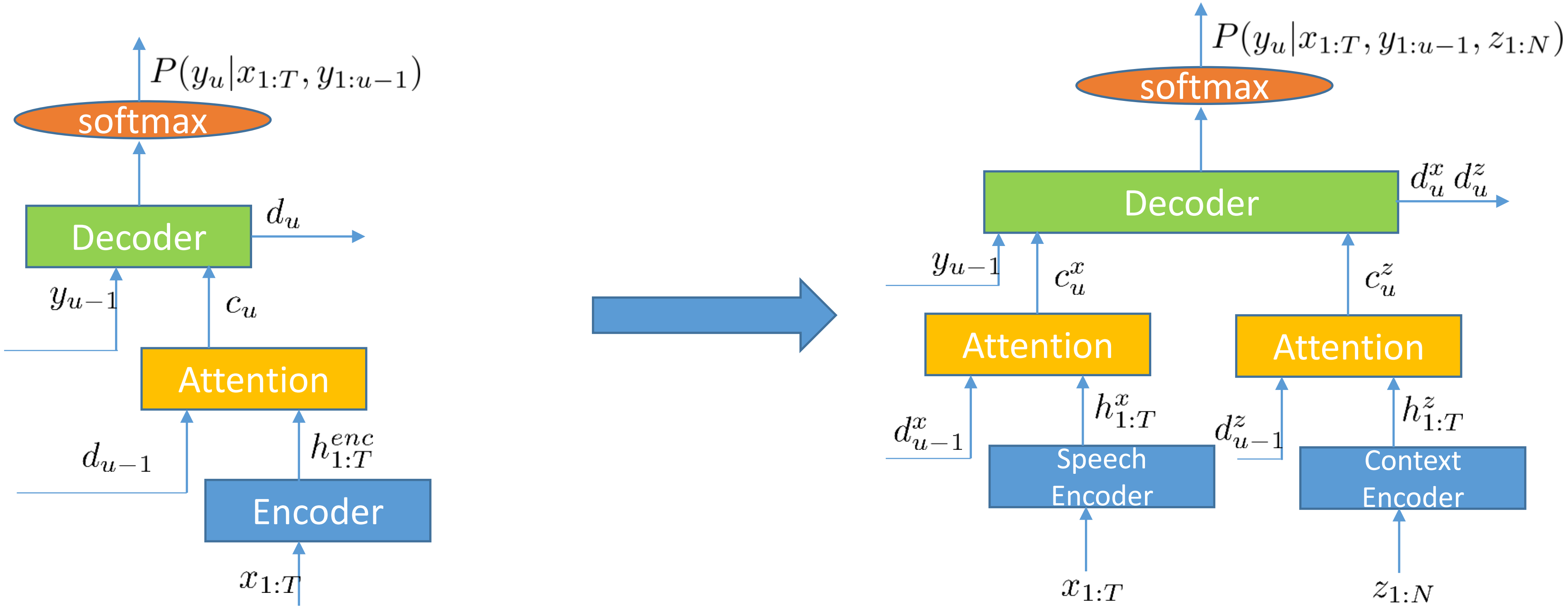
Customization – an Example

- An English ASR system usually cannot recognize the contact names of a Chinese person well ☹

- If the English ASR system is presented with the contact list of this Chinese person, the ASR output can be biased toward the contact names ☺

- Such biasing is even more effective when designed with context activation phrases such as "call", "email", "text" etc. ☺

Contextual Biasing Models



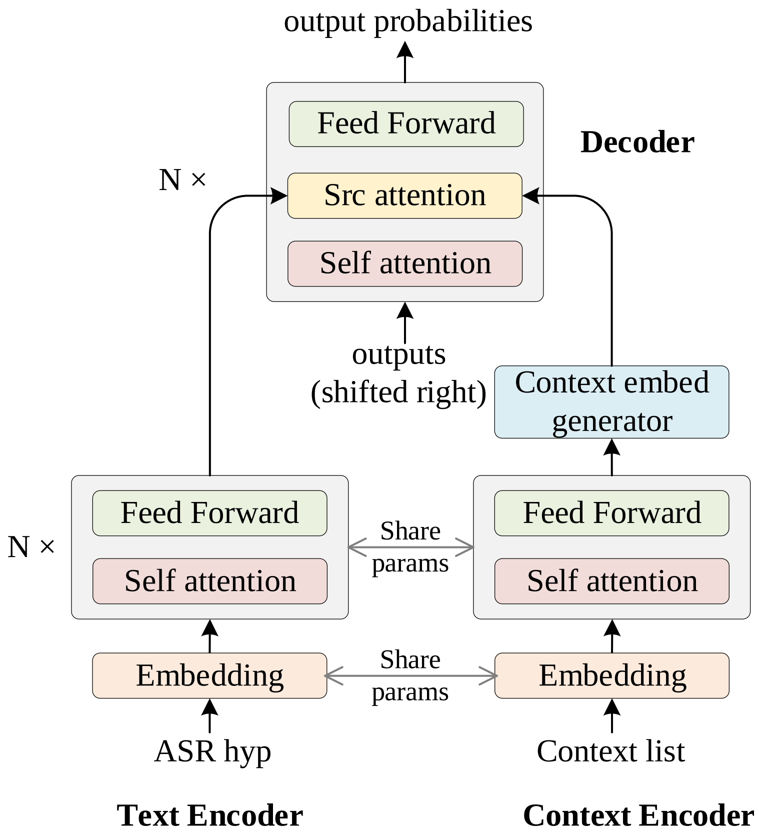
- Effective for small phrase list
- Challenging for the bias attention module to focus if the biasing list is too large

Pundak et al., “Deep context: end-to-end contextual speech recognition,” *in Proc. SLT*, 2018.
Pundak et al., “Phoebe: Pronunciation-aware contextualization for end-to-end speech recognition,” *in Proc. ICASSP*, 2019.
Jain et al., “Contextual RNN-T for Open Domain ASR,” *in Proc. Interspeech*, 2020.

Contextual Spelling Correction

- Both the embeddings of ASR hypothesis and contextual phrase list are used as the input to the decoder

- Key to success: A filtering mechanism is used to trim the very large phrase list to a relatively small one so that the attention can perform well



Wang et al., "A Light-weight contextual spelling correction model for customizing transducer-based speech recognition systems," *in Proc. Interspeech*, 2021.

Microsoft

E2E Advances – Advanced Models

Non-autoregressive Models

- Autoregressive models: predict target tokens in a left-to-right manner – slow decoding speed with high accuracy.

$$P(\mathbf{y}|\mathbf{x}) = \prod_u P(y_u|\mathbf{x}, \mathbf{y}_{1:u-1})$$

- Non-autoregressive models: generates all target tokens simultaneously – fast decoding speed with slightly low accuracy.

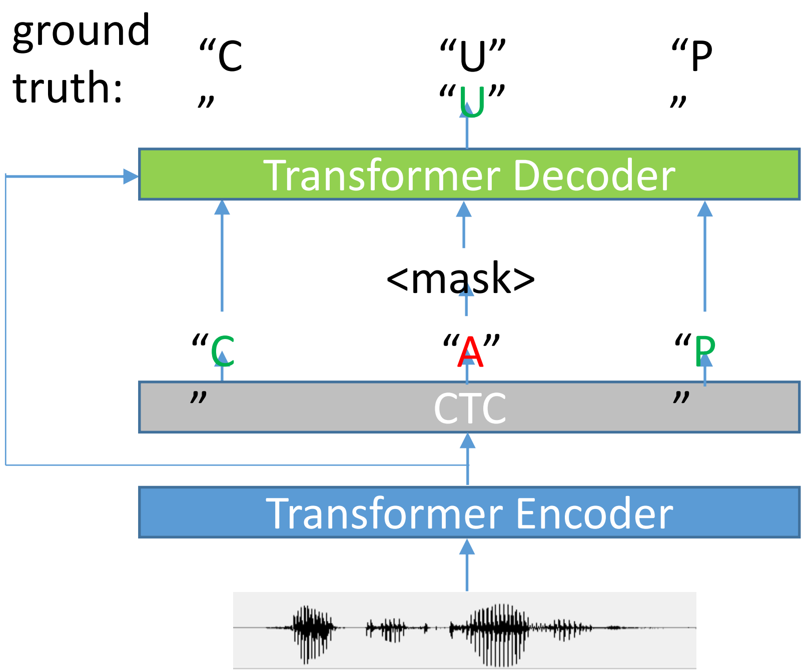
$$P(\mathbf{y}|\mathbf{x}) = \prod_{u=1}^{L} P(y_u|\mathbf{x})$$

Bai et al., "Listen Attentively, and Spell Once: Whole Sentence Generation via a Non-Autoregressive Architecture for Low-Latency Speech Recognition," in Proc. Interspeech, 2020.
Chen et al., "Non-autoregressive transformer for speech recognition," in IEEE Signal Processing Letters, 2020.

Mask CTC

- Mask CTC: predicts a set of masked tokens conditioning on the observed tokens

$$P(\mathbf{y}_{mask}|\mathbf{y}_{obs},\mathbf{x}) = \prod_{y\in\mathbf{y}_{mask}} P(y|\mathbf{y}_{obs},\mathbf{x})$$



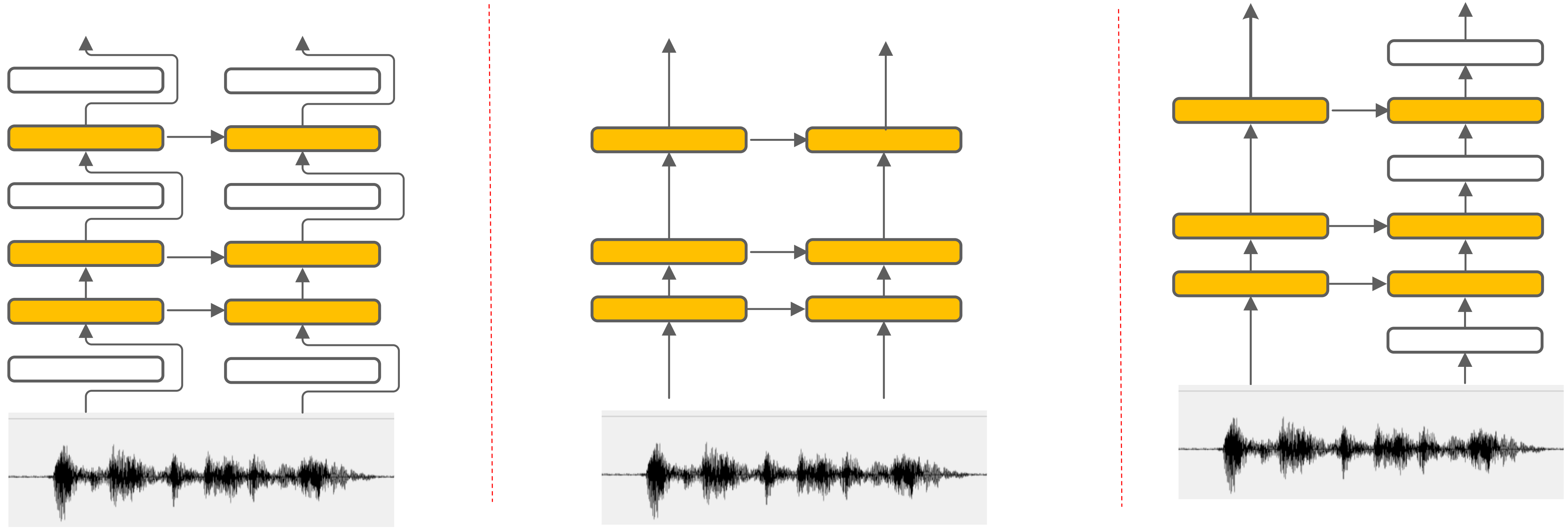
Higuchi et al., "Mask CTC: Non-autoregressive end-to-end ASR with CTC and mask predict," in Proc. Interspeech, 2020.

Unified Models – Trained Once, Deployed in Multiple Scenarios

- Dual model: unifies streaming and non-streaming modes
- Dynamic encoder: dynamic computational cost during inference
- Variable context encoder: configured for different latency requirement at runtime

Yu et al., "Dual-mode ASR: Unify and Improve Streaming ASR with Full-context Modeling," in Proc. ICLR, 2021.
Wu et al., "Dynamic sparsity neural networks for automatic speech recognition," in Proc. ICASSP, 2021.
Shi et al., "Dynamic Encoder Transducer: A Flexible Solution for Trading Off Accuracy for Latency," in Proc. Interspeech, 2021.
Tripathi et al., "Transformer transducer: One model unifying streaming and non-streaming speech recognition," in arXiv preprint, 2021.

Dynamic Encoder



training with layer dropout

pruned encoder in decoding

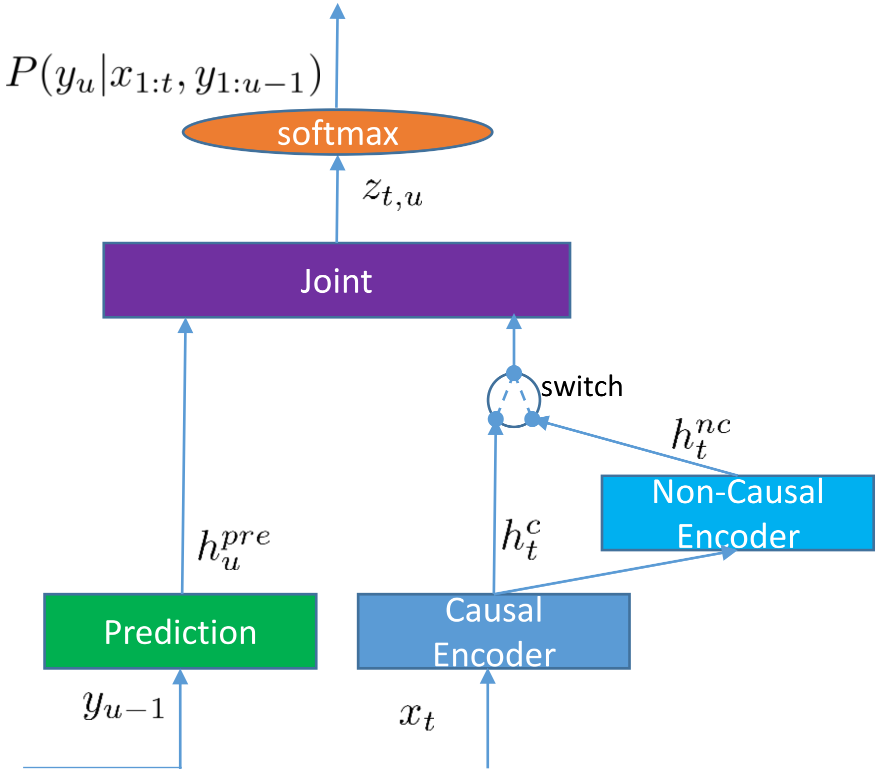
dynamic encoder in decoding

Shi et al., “Dynamic Encoder Transducer: A Flexible Solution for Trading Off Accuracy for Latency,” in Proc. Interspeech, 2021.

Two-pass Models

- The first-pass RNN-T provides immediate ASR results while the second-pass AED can provide better accuracy.

- Cascade model: first-pass causal RNN-T + second-pass non-causal RNN-T.



Sainath et al., “Two-pass end-to-end speech recognition,” in Proc. Interspeech, 2019.
Hu et al., “Deliberation model based two-pass end-to-end speech recognition,” in Proc. ICASSP, 2020.
Narayanan et al., “Cascaded encoders for unifying streaming and non-streaming ASR,” in Proc. ICASSP, 2021.

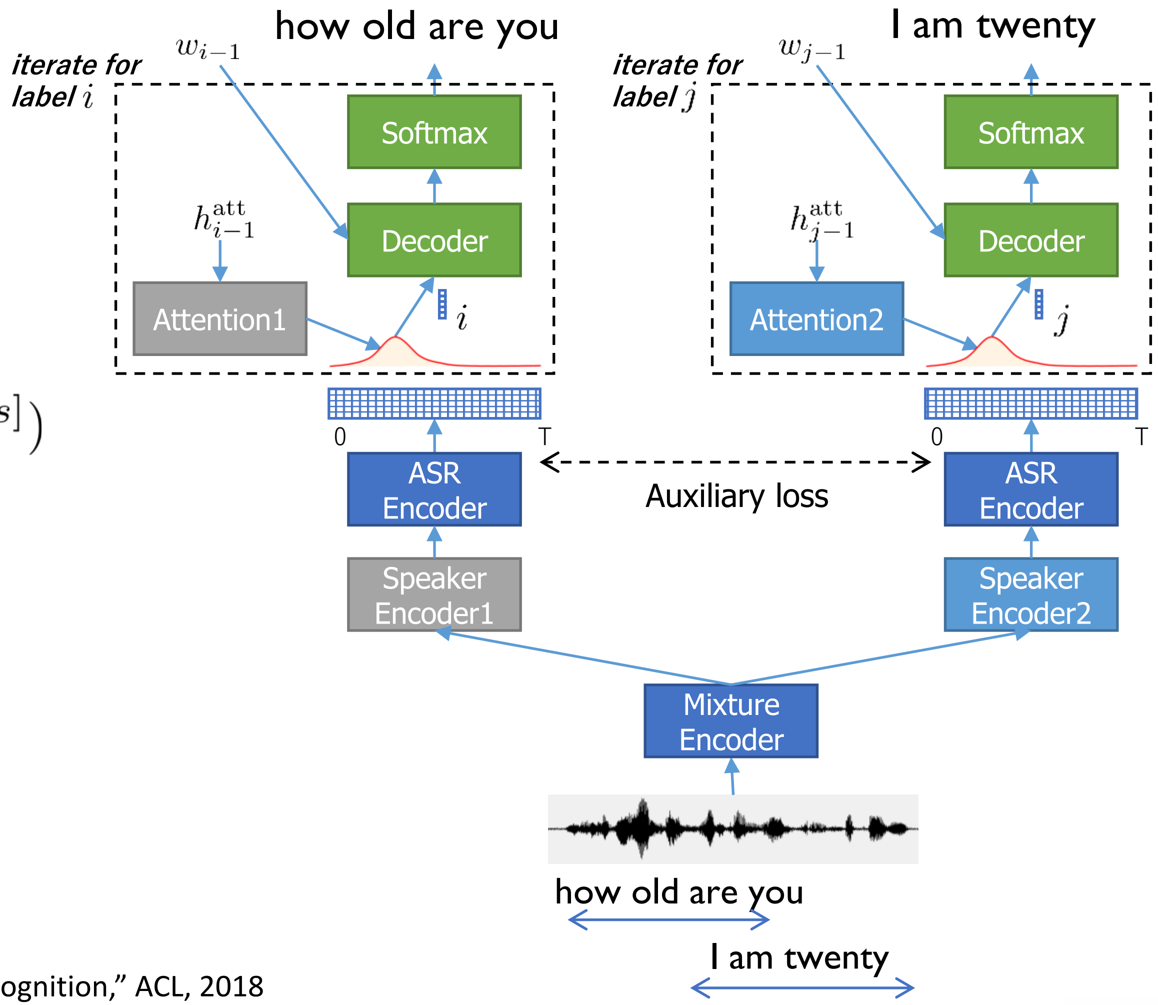
Multi-talker Models

- E2E ASR systems have high accuracy in single-speaker applications ☺
- Very difficult to achieve satisfactory accuracy in scenarios with multiple speakers talk at the same time ☹
- Solutions: E2E multi-talker models

Multi-talker AED Model with PIT

- No need for noisy-clean audio pair for training.

$$L^{PIT} = \min_{\phi \in \Phi(1,\ldots,S)} \sum_{s=1}^{S} CE(\mathbf{y}^s, \mathbf{r}^{\phi[s]})$$

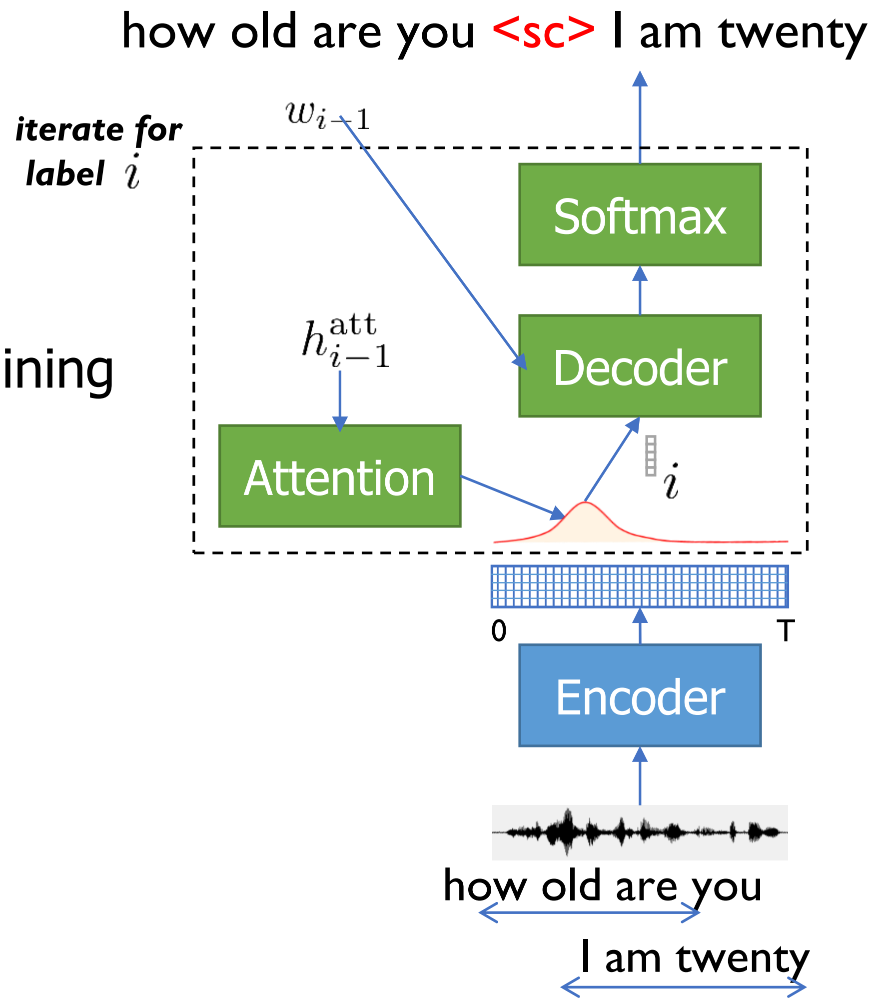


Seki et al., "A Purely End-to-end System for Multi-speaker Speech Recognition," ACL, 2018
Chang et al., End-to-End Monaural Multi-speaker ASR System without pretraining, Proc. ICASSP 2019.

Multi-talker AED Model with SOT

- Can recognize any number of speakers
- Can count the number of speakers
- Serialized output training: first in first out training for O(S) training

$$L^{SOT} = CE(\mathbf{y}, \Psi(1, ..., S))$$

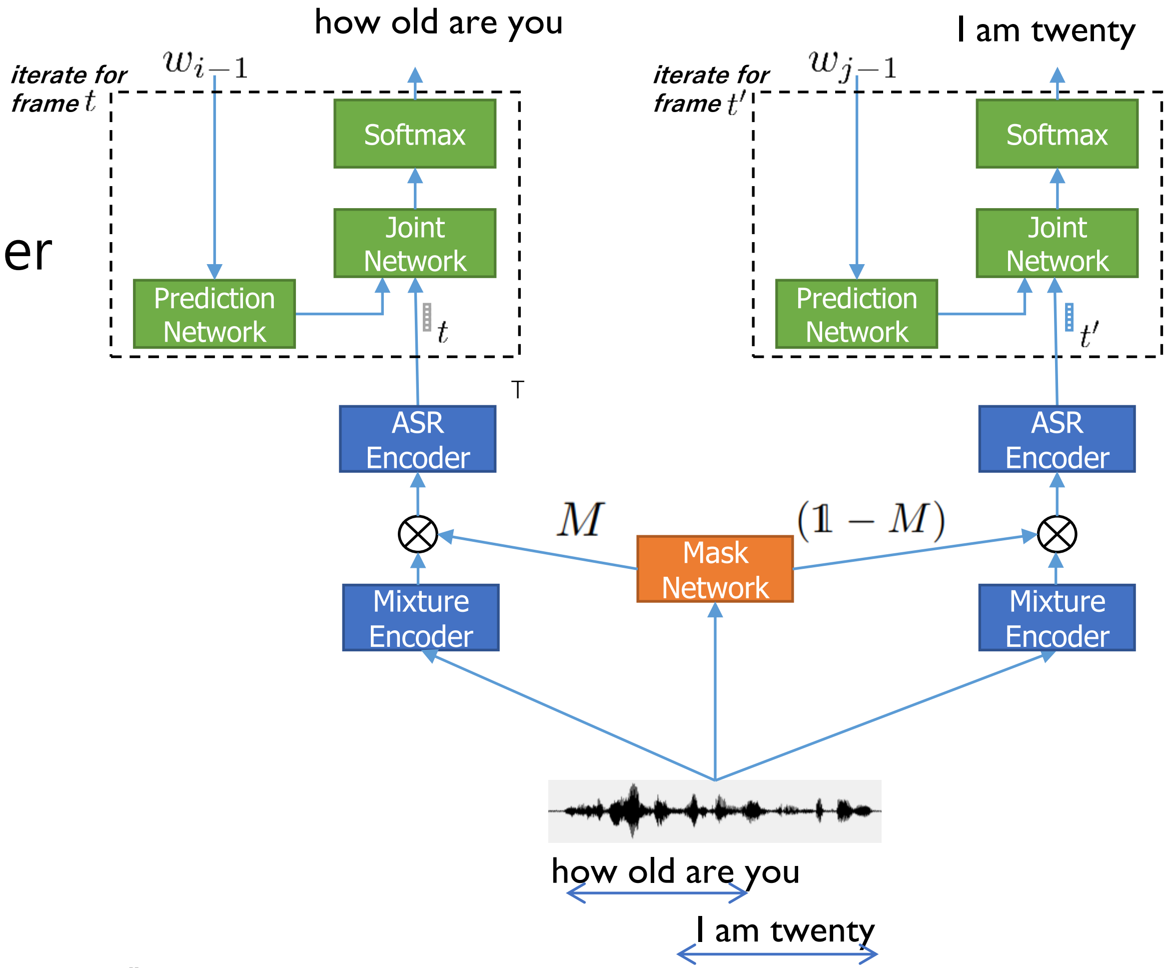


Kanda et al., "Serialized Output Training for End-to-End Overlapped Speech Recognition," in *Proc. Interspeech*, 2020.

Streaming Multi-talker RNN-T Model with HEAT

- Streaming
- Heuristic Error Assignment Training: order the label sequences based on the utterance start time

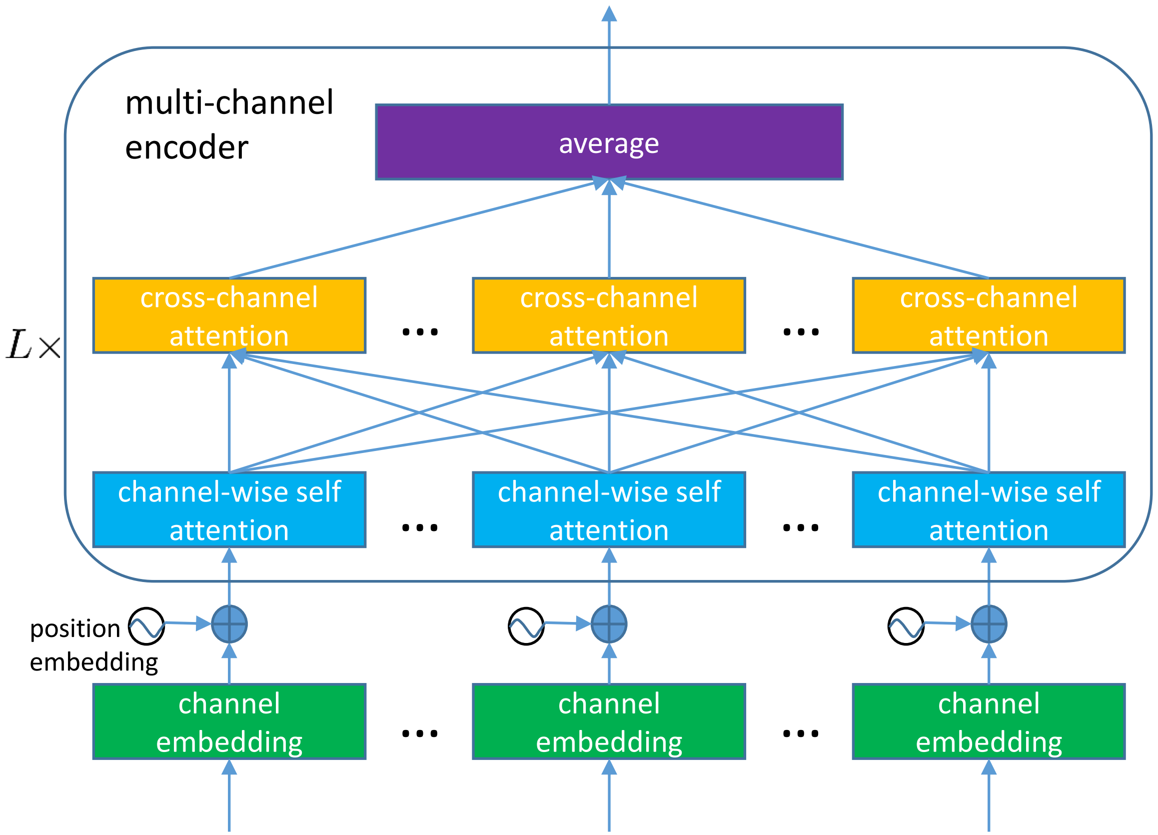
$$L^{HEAT} = \sum_{s=1}^{S} CE(\mathbf{y}^s, \mathbf{r}^{\omega[s]})$$



Tripathi et al., "End-to-end multi-talker overlapping speech recognition," in: Proc. *ICASSP*. 2020.
Lu et al., "Streaming end-to-end multi-talker speech recognition," *IEEE Signal Processing Letters, 2021.*
Sklyar et al., "Streaming Multi-speaker ASR with RNN-T," in Proc. *ICASSP*, 2021.

Multi-channel Models

- Channel-wise self attention: models the correlation across time within a channel
- Cross-channel attention: learns the relationship across channels



Chang et al., "Multi-Channel Transformer Transducer for Speech Recognition," in Proc. Interspeech, 2021.

Conclusions

- We overview E2E models and practical technologies that enable E2E models to potentially replace hybrid models.
  - Encoder: Transformer – attention mask
  - Multilingual: configurable multilingual model
  - Adaptation: LM fusion, TTS adaptation, splicing data, factorized neural transducer, contextual biasing model, and contextual spelling correction
  - Advanced models: Non-autoregressive Models, Unified Models, Two-pass Models, Multi-talker Models, and Multi-channel Models

Challenges

- How to leverage LM training text data
  - E2E models mainly use paired speech-text data
- How to integrate knowledge
  - E.g., it is hard for E2E models to directly output "5:45" when the user says "a quarter to six"
- How to add new words without biasing
  - There are trending words everyday

Thank You!