Carnegie Mellon University
Language Technologies Institute

Controllable and Explainable End-to-End Speech Translation

Shinji Watanabe and Brian Yan

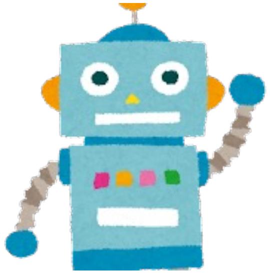
Language Technologies Institute
Carnegie Mellon University

Watanabe's Audio and Voice Lab

SIG SLT Seminar, November 18, 2022

Breaking the End-to-End Black Box

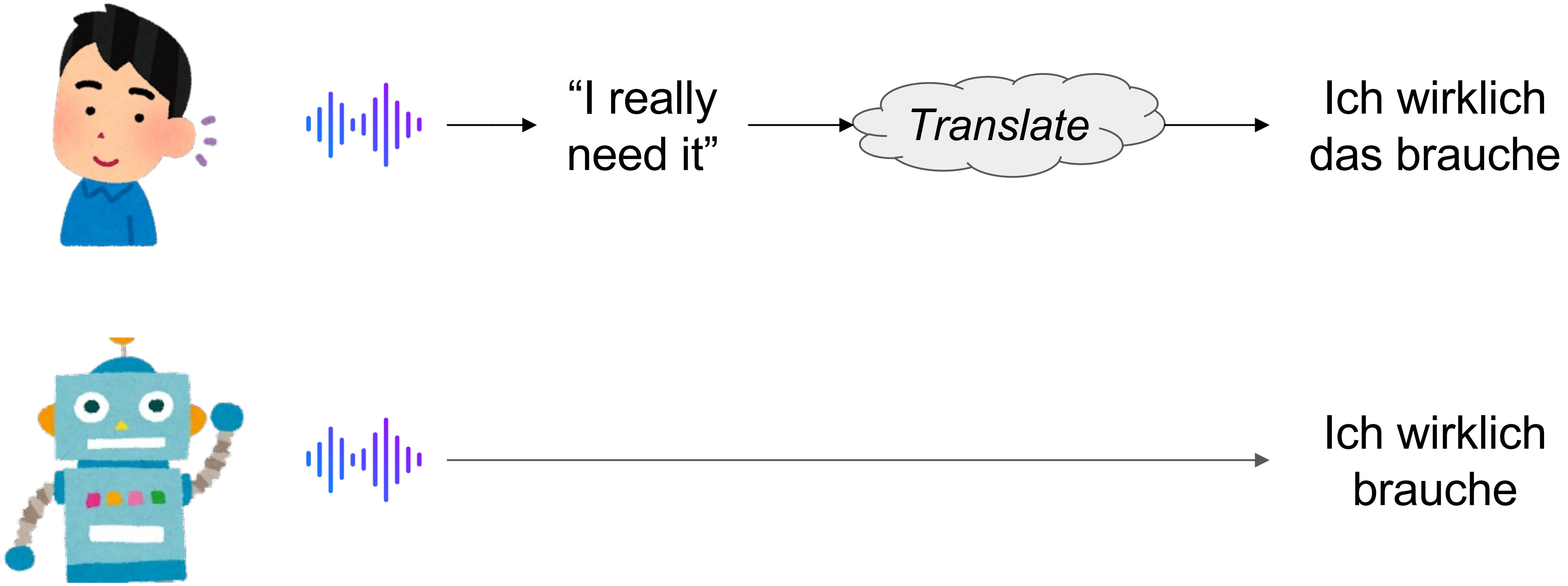
"I really need it" → *Translate* → Ich wirklich das brauche



→ Ich wirklich brauche

What went wrong and how can it be fixed?

Breaking the End-to-End Black Box

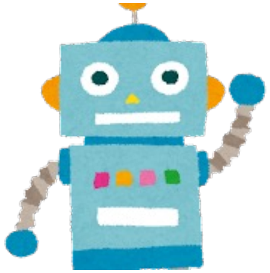


What went wrong and how can it be fixed?

- Generating translations that are too short? **Part 1**

Breaking the End-to-End Black Box

"I really need it" → *Translate* → Ich wirklich das brauche



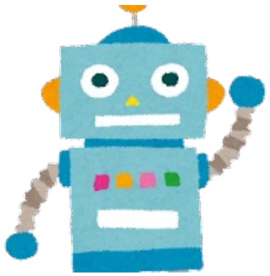
"I really need" → Ich wirklich brauche

What went wrong and how can it be fixed?

- Generating translations that are too short? **Part 1**
- Recognizing what was said? **Part 2**

Breaking the End-to-End Black Box

"I really need it" → *Translate* → Ich wirklich das brauche



"I really need" → Ich wirklich brauche

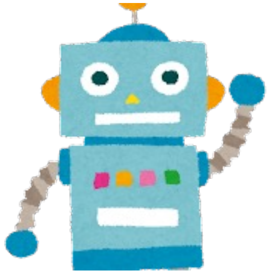
Still, I don't know why this error happens

What went wrong and how can it be fixed?

- Generating translations that are too short? **Part 1**
- Recognizing what was said? **Part 2**

Breaking the End-to-End Black Box

"I really need it" → *Translate* → Ich wirklich das brauche



"I really need **[it]**" → Ich wirklich **[das]** brauche

Still, I don't know why this error happens **because I don't know that "it" corresponds to "das"**

What went wrong and how can it be fixed?

- Generating translations that are too short? **Part 1**
- Recognizing what was said? **Part 2**
- Explaining why this is mistranslated? **Part 3**

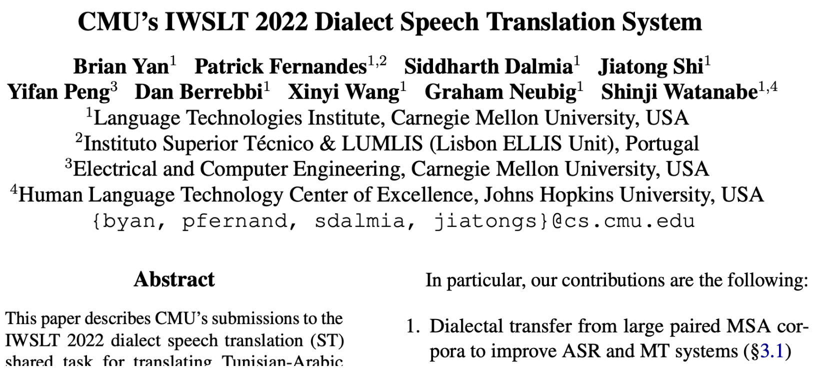
Today's Talk

- CMU's IWSLT 2022 Dialect Speech Translation System
  - **Part 1:** Controlling ST output lengths via joint CTC/attention
  - **Part 2:** Controlling/explaining ST via searchable ASR intermediates
- Explainable E2E Speech Translation via Operation Sequence Generation
  - **Part 3:** Explaining ST via word-level ASR alignments

Today’s Talk

- CMU’s IWSLT 2022 Dialect Speech Translation System
  - **Part 1:** Controlling ST output lengths via joint CTC/attention
  - **Part 2:** Controlling/explaining ST via searchable ASR intermediates

- Explainable E2E Speech Translation via Operation Sequence Generation
  - **Part 3:** Explaining ST via word-level ASR alignments



CMU’s IWSLT 2022 Dialect Speech Translation System

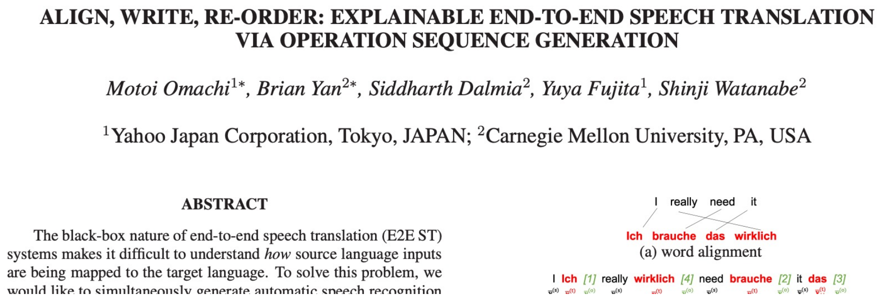
Brian Yan[1] **Patrick Fernandes**[1,2] **Siddharth Dalmia**[1] **Jiatong Shi**[1]
Yifan Peng[3] **Dan Berrebbi**[1] **Xinyi Wang**[1] **Graham Neubig**[1] **Shinji Watanabe**[1,4]
[1]Language Technologies Institute, Carnegie Mellon University, USA
[2]Instituto Superior Técnico & LUMLIS (Lisbon ELLIS Unit), Portugal
[3]Electrical and Computer Engineering, Carnegie Mellon University, USA
[4]Human Language Technology Center of Excellence, Johns Hopkins University, USA
{byan, pfernand, sdalmia, jiatongs}@cs.cmu.edu

Abstract

This paper describes CMU’s submissions to the IWSLT 2022 dialect speech translation (ST) shared task for translating Tunisian-Arabic

In particular, our contributions are the following:

1. Dialectal transfer from large paired MSA corpora to improve ASR and MT systems (§3.1)



ALIGN, WRITE, RE-ORDER: EXPLAINABLE END-TO-END SPEECH TRANSLATION VIA OPERATION SEQUENCE GENERATION

Motoi Omachi[1*], *Brian Yan*[2*], *Siddharth Dalmia*[2], *Yuya Fujita*[1], *Shinji Watanabe*[2]

[1]Yahoo Japan Corporation, Tokyo, JAPAN; [2]Carnegie Mellon University, PA, USA

ABSTRACT

The black-box nature of end-to-end speech translation (E2E ST) systems makes it difficult to understand *how* source language inputs are being mapped to the target language. To solve this problem, we would like to simultaneously generate automatic speech recognition

(a) word alignment

Length Control in Speech Translation

What is a good translation?

- Correct meaning
- Correct **length**
    - e.g. isometric ST for subtitling

Source	*It is actually the true integration of the man and the machine.*
Baseline MT	Es ist tatsächlich die wahre Integration von Mensch und Maschine.
Isometric MT	Es ist die wirkliche Integration von Mensch und Maschine.

Example from IWSLT 2022 Isometric ST Track

Length Control in Speech Translation

What is a good translation?

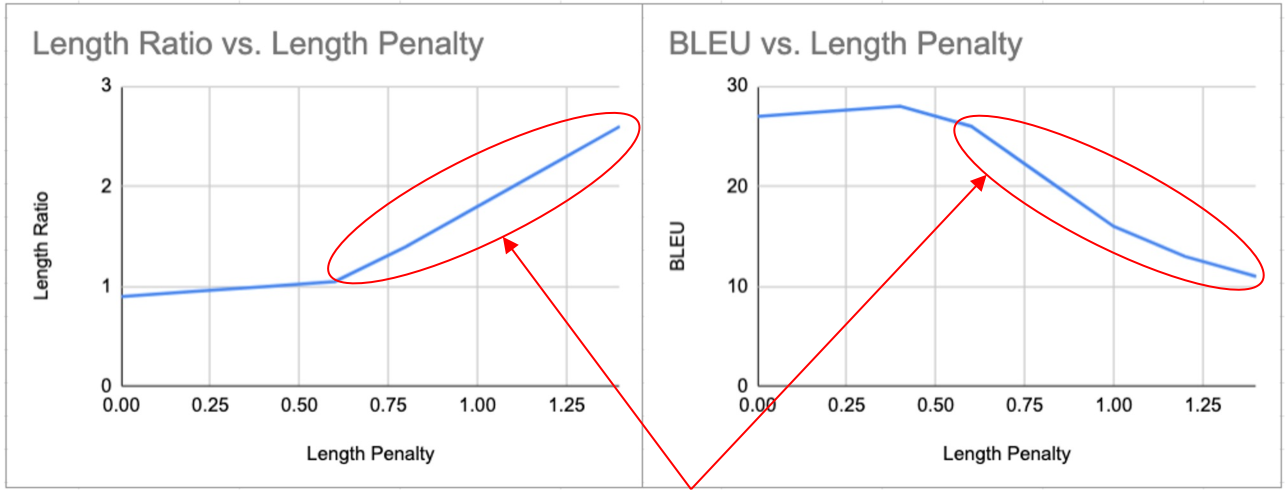
- Correct meaning
- Correct **length**
    - e.g. isometric ST for subtitling

Source	*It is actually the true integration of the man and the machine.*
Baseline MT	Es ist tatsächlich die wahre Integration von Mensch und Maschine.
Isometric MT	Es ist die wirkliche Integration von Mensch und Maschine.

Example from IWSLT 2022 Isometric ST Track

Problem: autoregressive decoders do not have robust end-detection

- Reliant on length penalty/bonus hyperparameter; not robust across domains/datasets



Degenerating quality due to incorrect length penalty leading to overly long outputs

Length Control in Speech Translation

Problem: autoregressive decoders do not have robust end-detection

- Reliant on length penalty/bonus hyperparameter; not robust across domains/datasets

Over-tuning easily happens! This was our experience in IWSLT 2021 😢

System	segm.	data condition	BLEU_TEDRef
ESPNET-ST	Own	Constrained	26.0
HW-TSC	Own	Constrained	25.4
KIT	Own	Constrained	25.4
ESPNET-ST	Own	Constrained	24.7
FBK	Own	Constrained	24.7
UPC†	Own	**Unconstrained**	24.6
APPTEK	Own	Constrained	24.5
VOLCTRANS	**Given**	Constrained	24.3
KIT	Own	Constrained	23.2
APPTEK	Own	Constrained	23.1
NIUTRANS	Own	Constrained	22.8
OPPO	**Given**	Constrained	22.6
VOLCTRANS	**Given**	Constrained	22.2
VUS	**Given**	Constrained	13.7
BUT	**Given**	**Unconstrained**	11.4
LI	**Given**	Constrained	0.2

Results on "original" blind test set; similar lengths to dev data

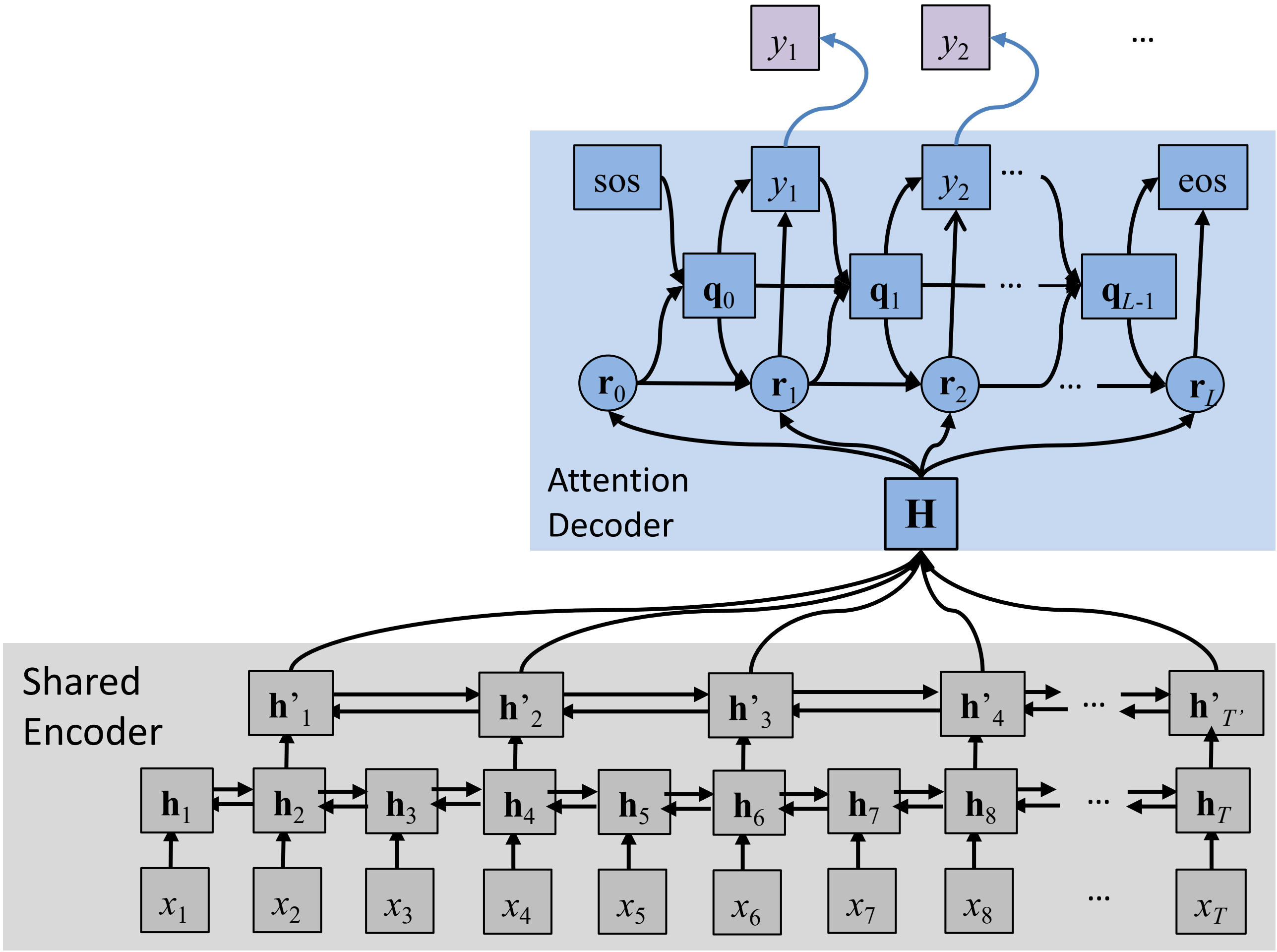
System	segm.	data condition	BLEU_NewRef	BLEU_TEDRef	BLEU_MultiRef
HW-TSC	Own	Constrained	24.6	20.3	34.0
KIT	Own	Constrained	23.4	19.0	32.0
APPTEK	Own	Constrained	22.6	18.3	31.0
KIT	Own	Constrained	22.0	18.1	30.3
APPTEK	Own	Constrained	21.9	18.1	30.4
VOLCTRANS	**Given**	Constrained	21.8	17.1	29.5
UPC†	Own	**Unconstrained**	21.8	18.3	30.6
VOLCTRANS	**Given**	Constrained	21.7	18.7	31.3
ESPNET-ST	Own	Constrained	21.7	18.2	30.6
FBK	Own	Constrained	21.6	18.4	30.6
OPPO	**Given**	Constrained	21.5	17.8	30.2
ESPNET-ST	Own	Constrained	21.2	19.3	31.4
NIUTRANS	Own	Constrained	20.6	19.6	30.3
VUS	**Given**	Constrained	15.3	12.4	20.9
BUT	**Given**	**Unconstrained**	11.7	9.8	16.1
LI	**Given**	Constrained	3.6	2.7	4.8

Results on new blind test set w/ **shorter references** (different annotation guidelines)

Hmm.
We had the same experience before…

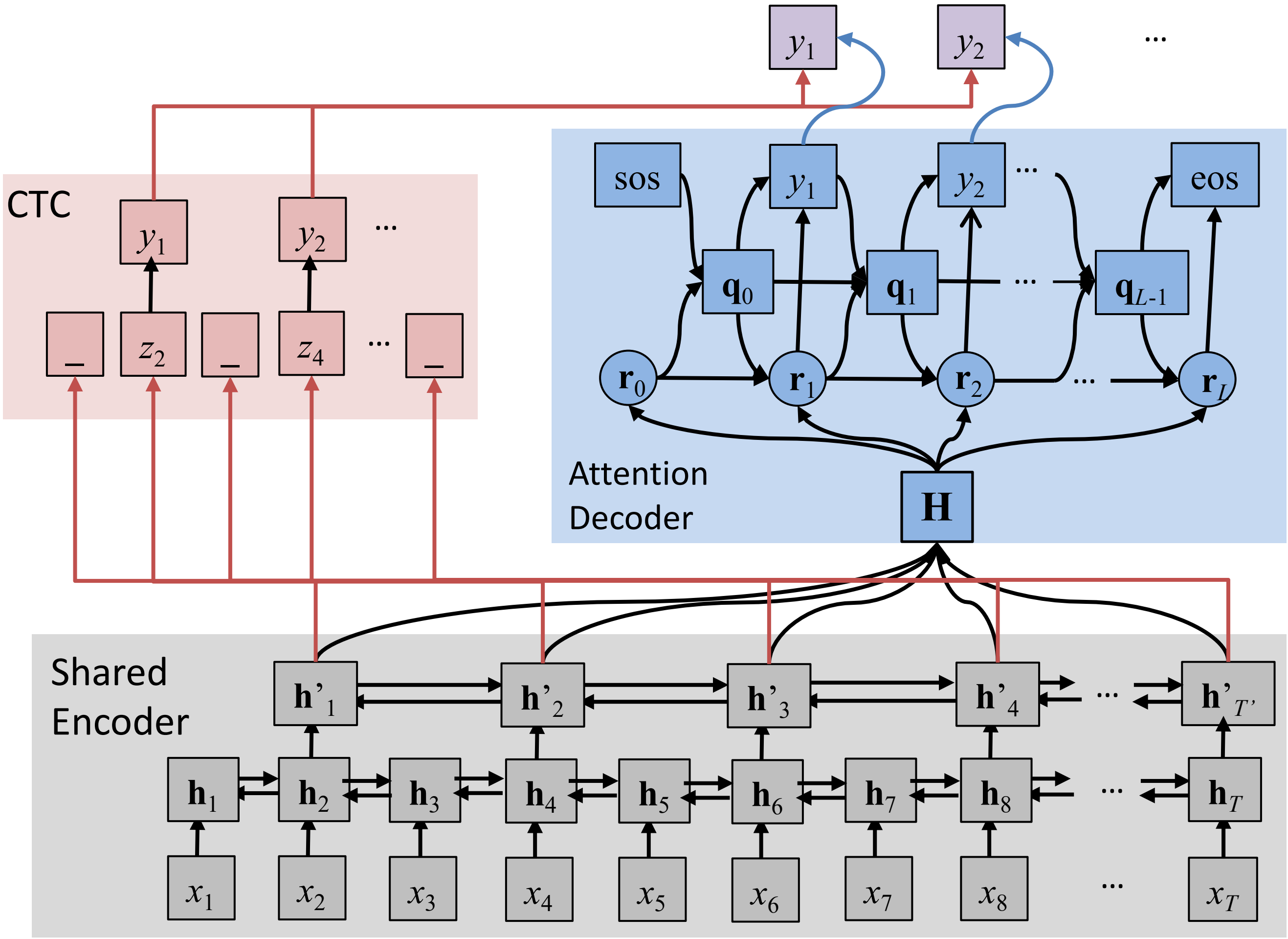
Joint CTC/Attention for ***ASR*** [Kim+ (2017), Hori+ (2017), Watanabe+ (2017)]

Joint CTC/Attention for ***ASR*** [Kim+ (2017), Hori+ (2017), Watanabe+ (2017)]



Joint CTC/Attention for *ASR* [Kim+ (2017), Hori+ (2017), Watanabe+ (2017)]

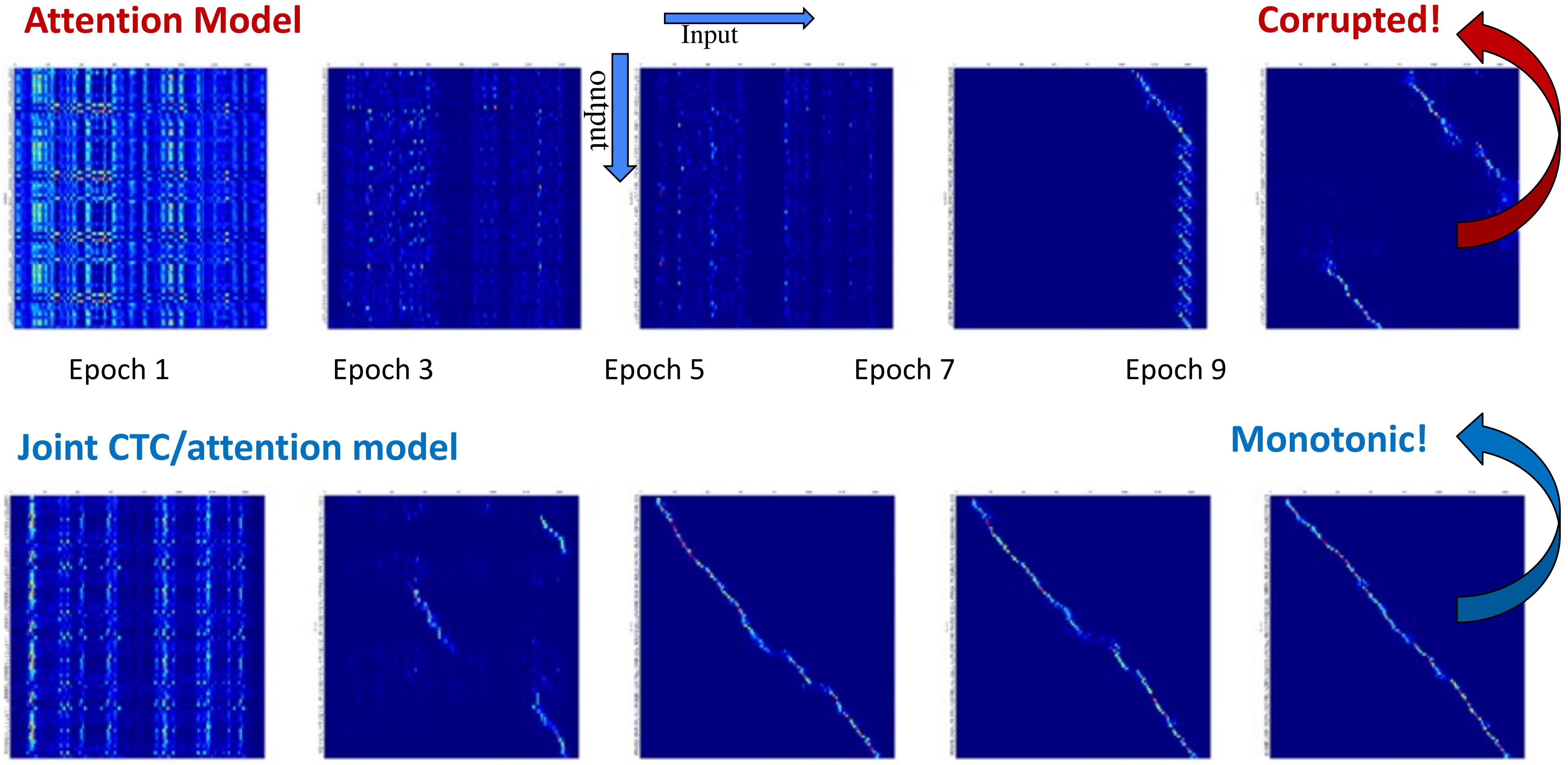
Use ***CTC*** for decoding together with the attention decoder



CTC explicitly eliminates non-monotonic alignment

More robust input/output alignment of attention

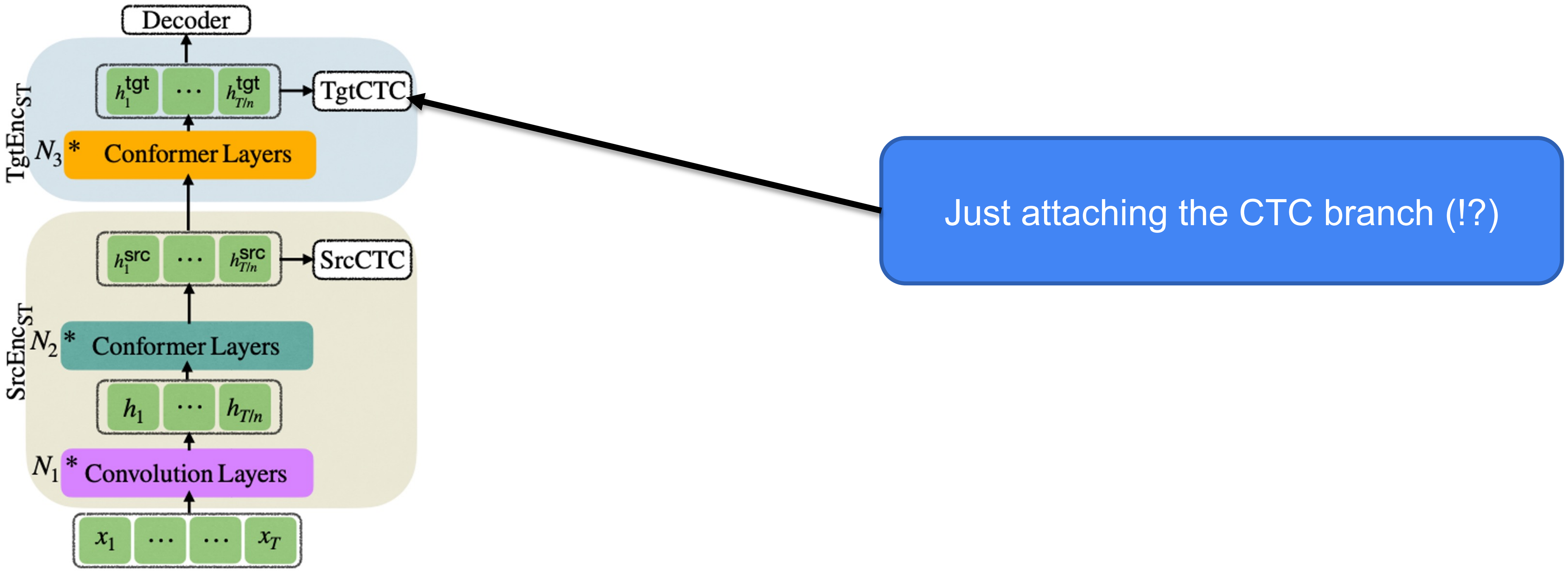
- Alignment of one selected utterance from CHiME4 ASR task



Correctly control the length!

Let's Apply Joint CTC/Attention Architecture to Speech Translation

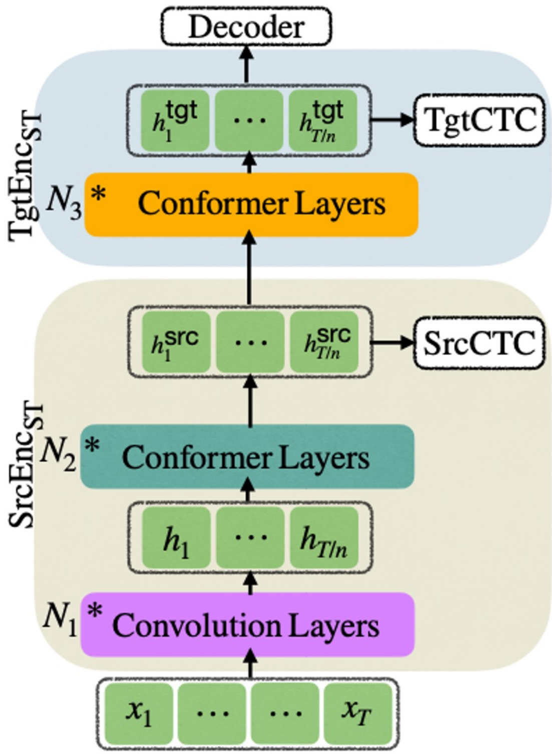
Hierarchical Encoding (ASR→ST)



$$\mathcal{L} = \mathcal{L}_{\text{SRCCTC}} + \lambda_1 \mathcal{L}_{\text{TGTCTC}} + \lambda_2 \mathcal{L}_{\text{ATTN}}$$

Let's Apply Joint CTC/Attention Architecture to Speech Translation

Hierarchical Encoding (ASR→ST)

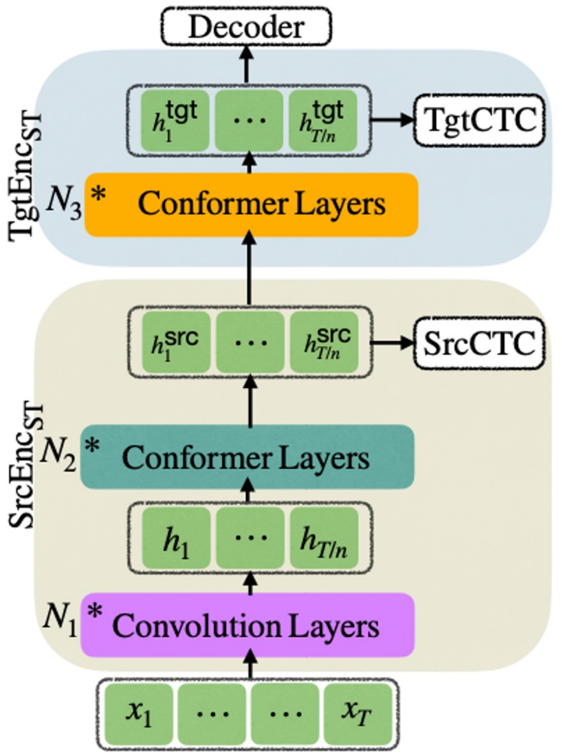


$$\mathcal{L} = \mathcal{L}_{\text{SRCCTC}} + \lambda_1 \mathcal{L}_{\text{TGTCTC}} + \lambda_2 \mathcal{L}_{\text{ATTN}}$$

*But wait … CTC is **monotonic** and ST requires re-ordering*

Joint CTC/Attention Architecture

Hierarchical Encoding (ASR→ST)

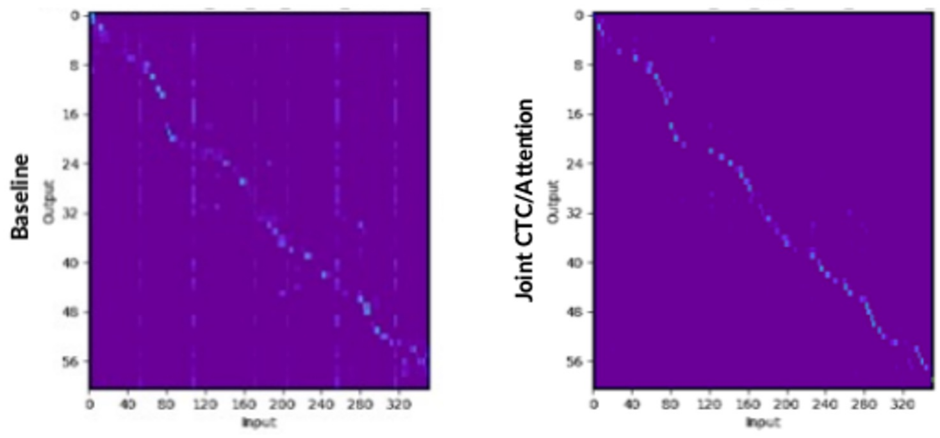


$$\mathcal{L} = \mathcal{L}_{\text{SRCCTC}} + \lambda_1 \mathcal{L}_{\text{TGTCTC}} + \lambda_2 \mathcal{L}_{\text{ATTN}}$$

*But wait … CTC is **monotonic** and ST requires re-ordering*

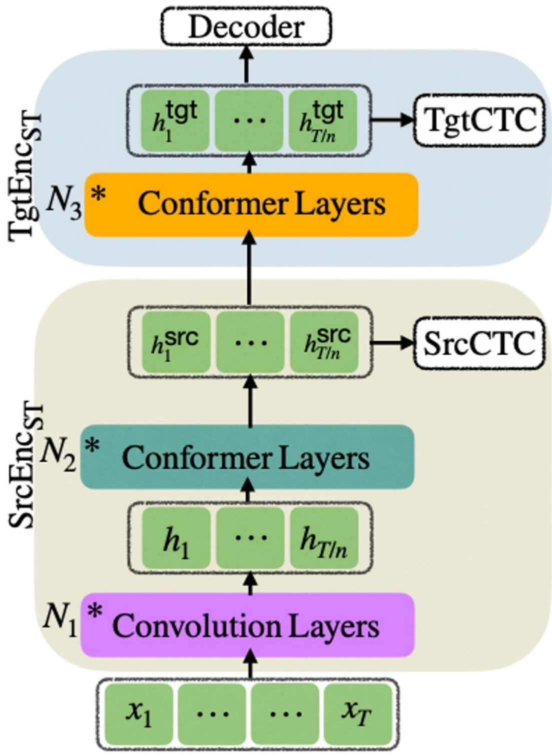
Self-attentional encoder learn to re-order

- Final encoder representations become **monotonic** w.r.t. target translations
- Decoder source attention patterns:



Joint CTC/Attention Architecture

Hierarchical Encoding (ASR→ST)

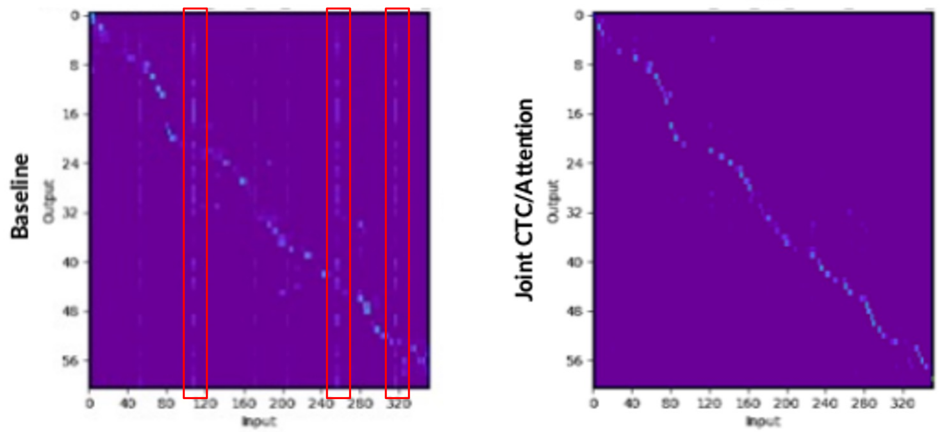


$$\mathcal{L} = \mathcal{L}_{\text{SRCCTC}} + \lambda_1 \mathcal{L}_{\text{TGTCTC}} + \lambda_2 \mathcal{L}_{\text{ATTN}}$$

*But wait … CTC is **monotonic** and ST requires re-ordering*

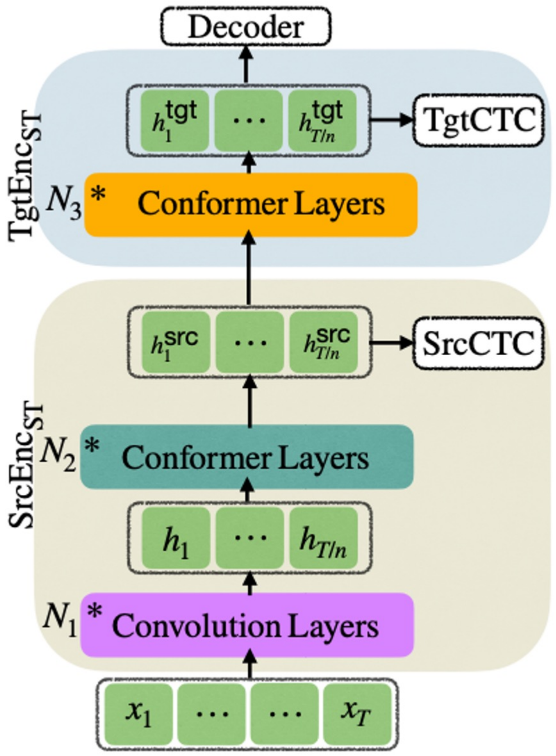
Self-attentional encoder learn to re-order

- Final encoder representations become **monotonic** w.r.t. target translations
- Decoder source attention patterns:



Joint CTC/Attention Architecture

Hierarchical Encoding (ASR→ST)

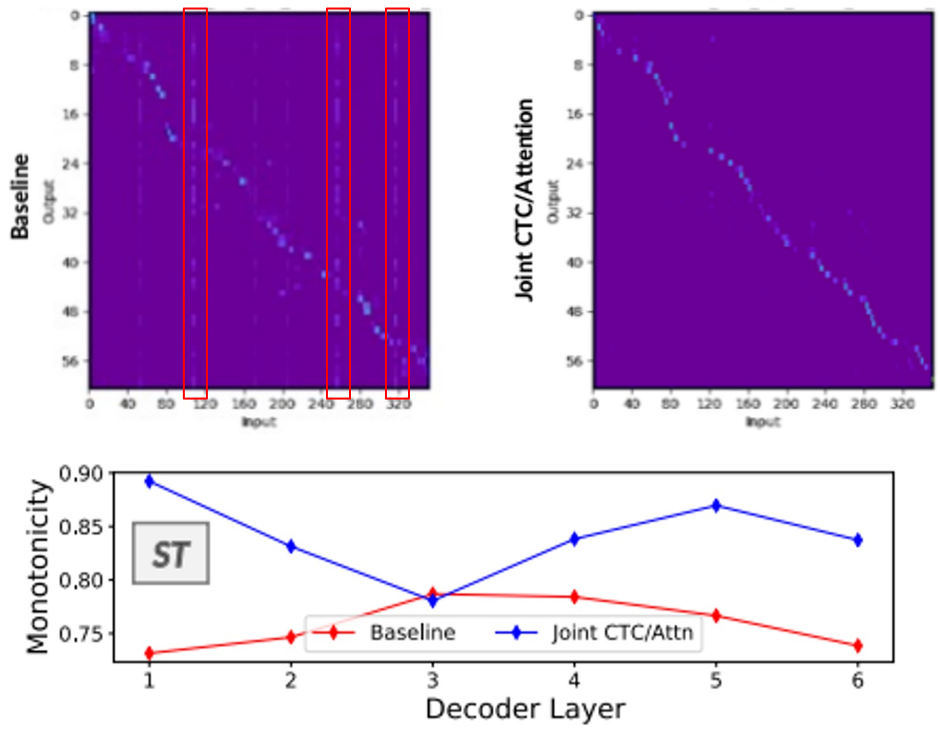


$$\mathcal{L} = \mathcal{L}_{\text{SrcCTC}} + \lambda_1 \mathcal{L}_{\text{TgtCTC}} + \lambda_2 \mathcal{L}_{\text{Attn}}$$

*But wait … CTC is **monotonic** and ST requires re-ordering*

Self-attentional encoder learn to re-order

- Final encoder representations become **monotonic** w.r.t. target translations
- Decoder source attention patterns:



Joint CTC/Attention Decoding: 2 Synchronous Methods

Algorithm 2 *Output*-Synchronous Step Function: attentional decoder proposes candidates to expand hypotheses which are all of l-length at step l.

```
procedure OUTPUTSTEP(prtHs, X, l, p, maxL)
    newPrtHs = {}; endHs = {}
    for y_{1:l-1} ∈ prtHs do
        attnCnds = top-k(P_Attn(y_l | X, y_{1:l-1}), k = p)
        for c ∈ attnCnds do
            y_{1:l} = y_{1:l-1} ⊕ c                          ← Hypothesis Expansion
            α_CTC = CTCScore(y_{1:l}, X_{1:T})
            α_Attn = AttnScore(y_{1:l}, X_{1:T})               ← Joint Scoring
            β = LengthPen(y_{1:l})
            P_Beam(y_{1:l} | X) = α_CTC + α_Attn + β
            if (c is <eos>) or (l is maxL) then
                endHs[y_{1:l}] = P_Beam(·)                    ← End Detection
            else
                newPrtHs[y_{1:l}] = P_Beam(·)
            end if
        end for
    end for
    return newPrtHs, endHs
end procedure
```

CTC prefix scores **indirectly** help end-detection by penalizing hypotheses of incorrect length

Joint CTC/Attention Decoding: 2 Synchronous Methods

Algorithm 2 *Output*-Synchronous Step Function: attentional decoder proposes candidates to expand hypotheses which are all of l-length at step l.

procedure OUTPUTSTEP(prtHs, X, l, p, max$L$)
  newPrtHs = {}; endHs = {}
  for $y_{1:l-1} \in$ prtHs **do**
    attnCnds = top-k$(P_{\text{Attn}}(y_l|X, y_{1:l-1}), \text{k} = p)$
    for $c \in$ attnCnds **do**
      $y_{1:l} = y_{1:l-1} \oplus \text{c}$
      $\boxed{\alpha_{\text{CTC}} = \text{CTCScore}(y_{1:l}, X_{1:T})}$
      $\alpha_{\text{Attn}} = \text{AttnScore}(y_{1:l}, X_{1:T})$
      $\beta = \text{LengthPen}(y_{1:l})$
      $P_{\text{Beam}}(y_{1:l}|X) = \alpha_{\text{CTC}} + \alpha_{\text{Attn}} + \beta$
      if (c is <eos>) or (l is max$L$) **then**
        endHs$[y_{1:l}] = P_{\text{Beam}}(\cdot)$
      else
        newPrtHs$[y_{1:l}] = P_{\text{Beam}}(\cdot)$
      end if
    end for
  end for
  return newPrtHs, endHs
end procedure

Hypothesis Expansion

Joint Scoring

End Detection

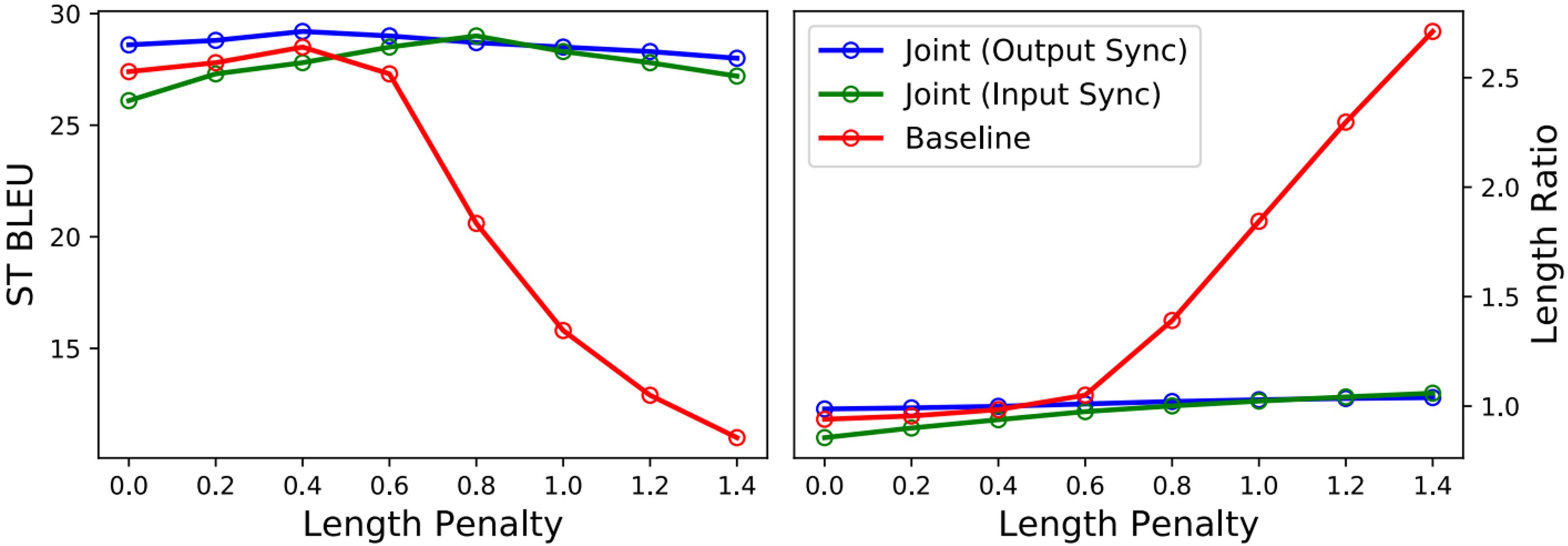
Algorithm 3 *Input*-Synchronous Step Function: CTC proposes candidates to expand hypotheses which are all produced from t input units at step t.

procedure INPUTSTEP(prtHs, X, t, p, T)
  newPrtHs = {}; endHs = {}
  CTCCnds = top-k$(P_{\text{CTC}}(z_t|X), \text{k} = p)$
  for $y \in$ prtHs **do**
    for $c \in$ CTCCnds **do**
      if (c is $\varnothing$) or (c is y[-1]) **then**
        $\tilde{y} = y$
      else
        $\tilde{y} = y \oplus \text{c}$
      end if
      $\alpha_{\text{CTC}} = \text{CTCScore}(\tilde{y}, X_{1:t})$
      $\alpha_{\text{Attn}} = \text{AttnScore}(\tilde{y}, X_{1:T})$
      $\beta = \text{LengthPen}(\tilde{y})$
      $P_{\text{Beam}}(\tilde{y}|X) = \alpha_{\text{CTC}} + \alpha_{\text{ATTN}} + \beta$
      $\boxed{\textbf{if } t \text{ is } T \textbf{ then}}$
        endHs$[\tilde{y}] = P_{\text{Beam}}(\cdot)$
      else
        newPrtHs$[\tilde{y}] = P_{\text{Beam}}(\cdot)$
      end if
    end for
  end for
  return newPrtHs, endHs
end procedure

CTC prefix scores **indirectly** help end-detection by penalizing hypotheses of incorrect length

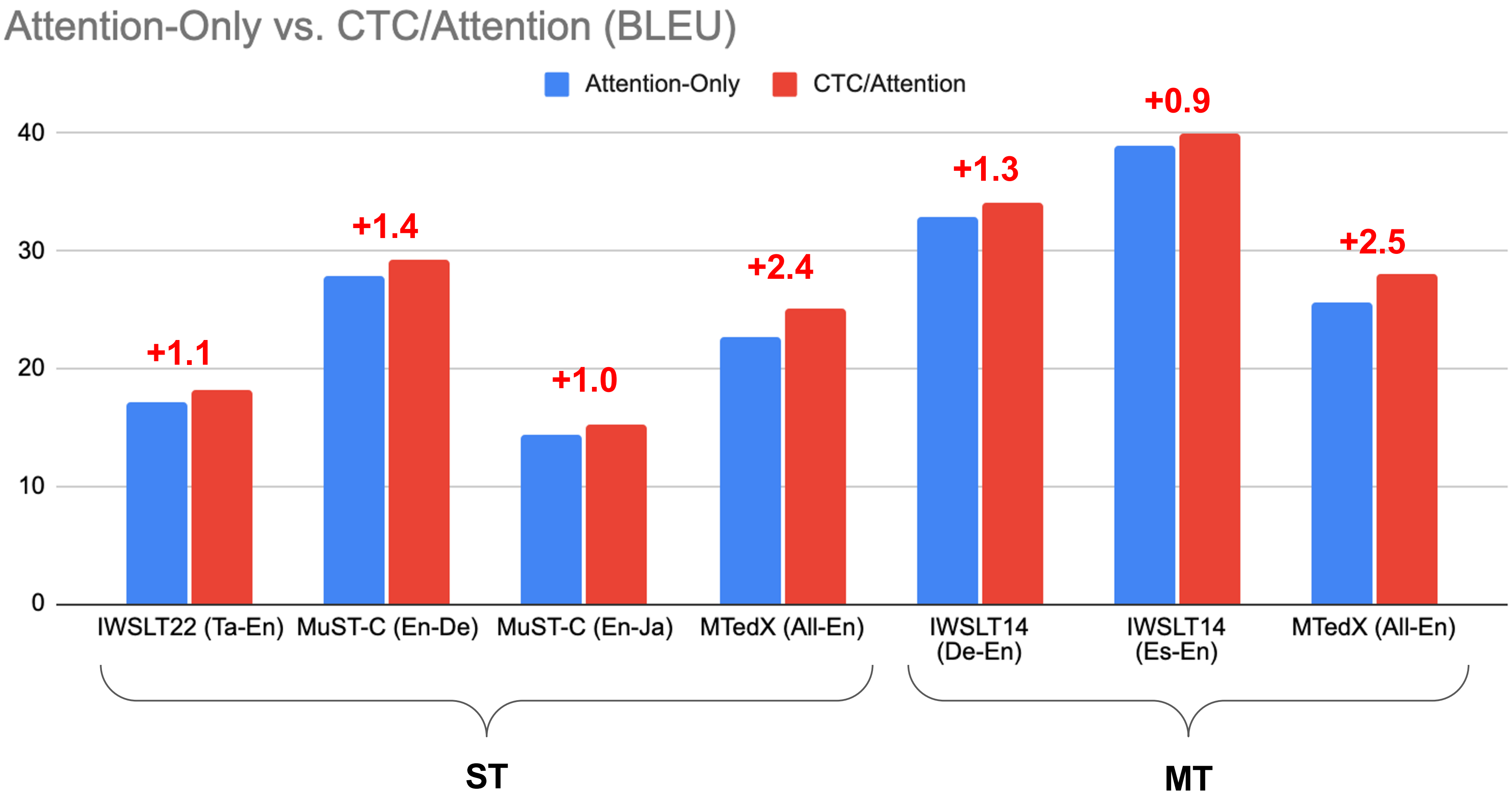
CTC **directly** handles end-detection by consuming all input frames

Joint CTC/Attention: Robust End-Detection



Carefully tuning length penalty may not be necessary!

Joint CTC/Attention: Results



Goodbye overturning!

ID	Type	Model Name	Child System(s)	Dialect Transfer	test1 BLEU(↑)	test2 BLEU(↑)
C1	Cascade	ASR Mixing Cascade	A1,B1	✗	16.4	-
C2	Cascade	+ ASR Rover Comb.	A2,B1	✗	16.7	-
C3	Cascade	+ MT Posterior Comb.	A2,B2	✗	17.5	18.6
C4	Cascade	ASR Mixing Cascade	A3,B3	✓	17.3	-
C5	Cascade	+ ASR Rover Comb.	A4,B3	✓	17.4	-
C6	Cascade	+ MT Posterior Comb.	A4,B4	✓	**17.9**	**19.4**
D1	E2E ST	Hybrid Multi-Decoder	-	✗	17.7	-
D2	Mix	+ ROVER Intermediates	A2	✗	18.1	19.1
D3	Mix	+ ST/MT Posterior Comb.	A2,B5	✗	18.7	19.7
D4	E2E ST	Hybrid Multi-Decoder	-	✓	18.2	-
D5	Mix	+ ROVER Intermediates	A4	✓	18.3	19.5
D6	Mix	+ ST/MT Posterior Comb.	A4,B5	✓	**18.9**	**19.8**
E1	Mix	Min. Bayes-Risk Ensemble	C3,D3	✗	19.2	20.4
E2	Mix	Min. Bayes-Risk Ensemble	C6,D6	✓	**19.5**	**20.8**

Table 3: Results of our cascaded, E2E, and integrated cascaded/E2E systems as measured by BLEU score on the blind test2 and provided test1 sets. *Dialect Transfer* indicates the use of either MGB2 or OPUS data. Rover, posterior combinations, and minimum bayes-risk ensembling were applied to both cascaded and E2E systems, with *Child System(s)* indicating the inputs to the resultant systems combinations.

Our tuning efforts have high correlation with the blind test set (test2)

Goodbye overturning!

Team / Condition / System	Architecture	Training Data	BLEU	Δ
CMU / basic / E1	Mix	TA/EN	20.4	-
CMU / dialect adapt / E2	Mix	TA/EN + MSA/EN	20.8	0.4
JHU / basic / primary	Cascaded	TA/EN	17.1	-
JHU / dialect adapt / primary	Cascaded	TA/EN + MSA/EN	18.9	1.8
ON-TRAC / basic /primary	End-to-End	TA/EN	12.4	-
ON-TRAC / unconstrained / post-eval	Cascaded	TA/EN + MSA/EN	14.4	2.0

Table 6: Summary of select systems for Dialect Shared Task (BLEU on test2). We highlight the BLEU improvements (Δ) obtained when training with additional MSA/English data compared with just the Tunisian/English (TA/EN) in the basic condition.

Anastasopoulos, Antonios, et al. "Findings of the IWSLT 2022 Evaluation Campaign." Proceedings of the 19th International Conference on Spoken Language Translation (IWSLT 2022). 2022.

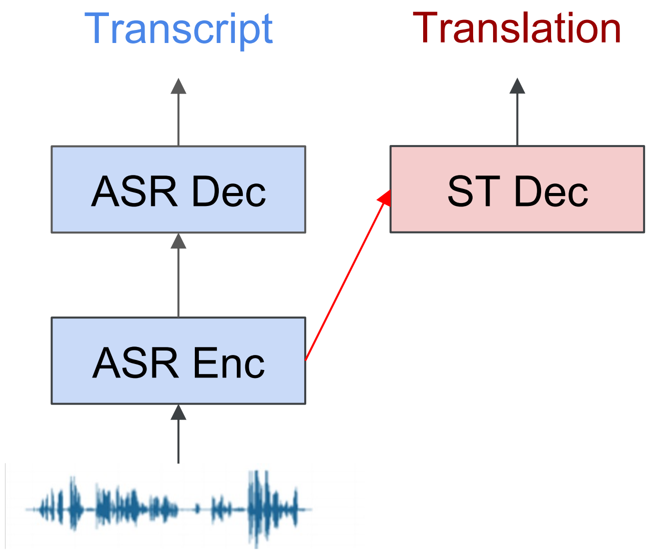
We got the nice result this time ☺

Today’s Talk

- CMU’s IWSLT 2022 Dialect Speech Translation System
  - **Part 1:** Controlling ST output lengths via joint CTC/attention
  - **Part 2:** Controlling/explaining ST via searchable ASR intermediates
- Explainable E2E Speech Translation via Operation Sequence Generation
  - **Part 3:** Explaining ST via word-level ASR alignments

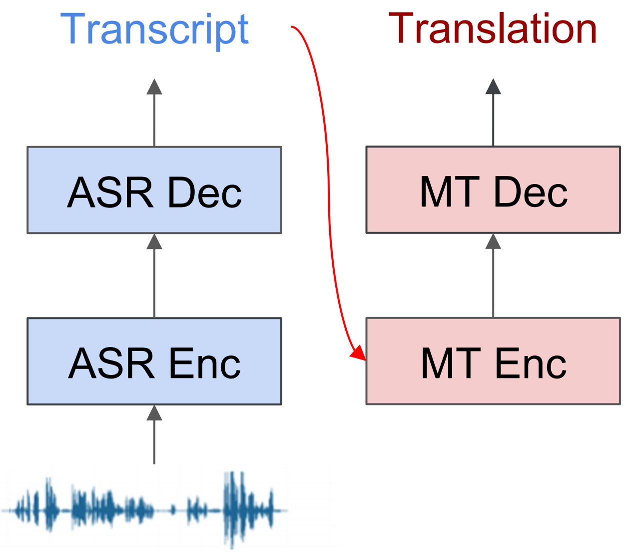
Better Speech Translation via Better Speech Recognition

Vanilla E2E (w/ ASR Multi-Task)



ASR and ST decodings are independent / parallel

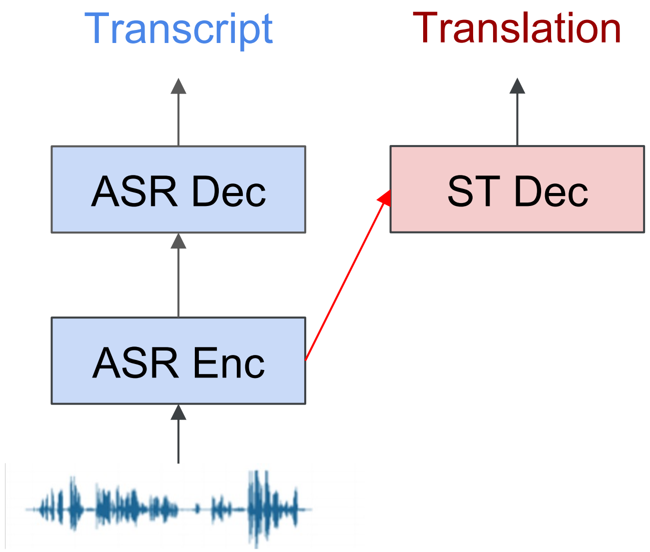
Fully Cascaded



Better transcription likely to yield better translation

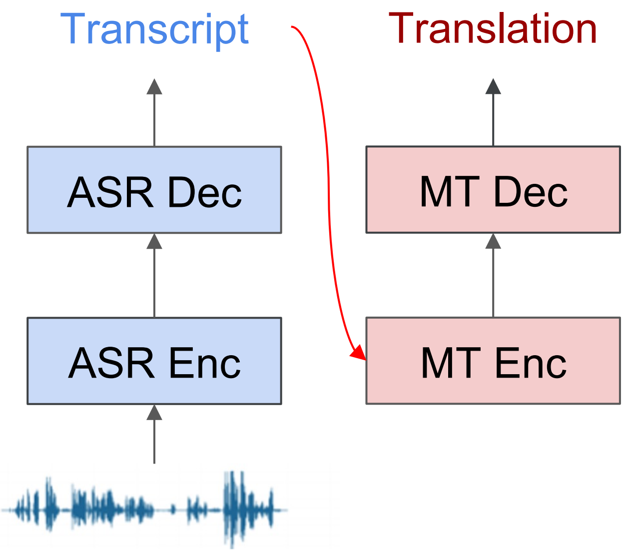
Better Speech Translation via Better Speech Recognition

**Vanilla E2E
(w/ ASR Multi-Task)**



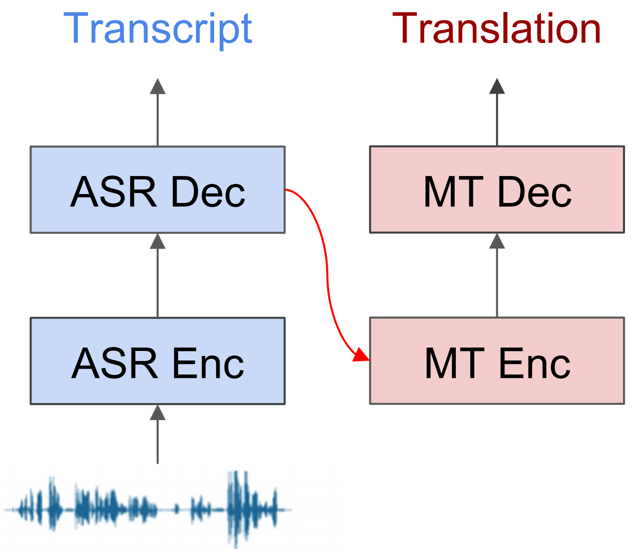
ASR and ST decodings are independent / parallel

Fully Cascaded



Better transcription likely to yield better translation

**E2E Multi-Decoder
(ASR Searchable Hidden Intermediates)**



Can we make an E2E differentiable cascade?

ASR Decoder State

E2E ASR based on attention

- Transcript is obtained by the conditional likelihood

$$\mathrm{argmax}_{\mathrm{W}}\, p(W|O) = \mathrm{argmax}_{\mathrm{W}} \prod_j p(w_j|\mathbf{h}_j)$$

- ASR decoder state

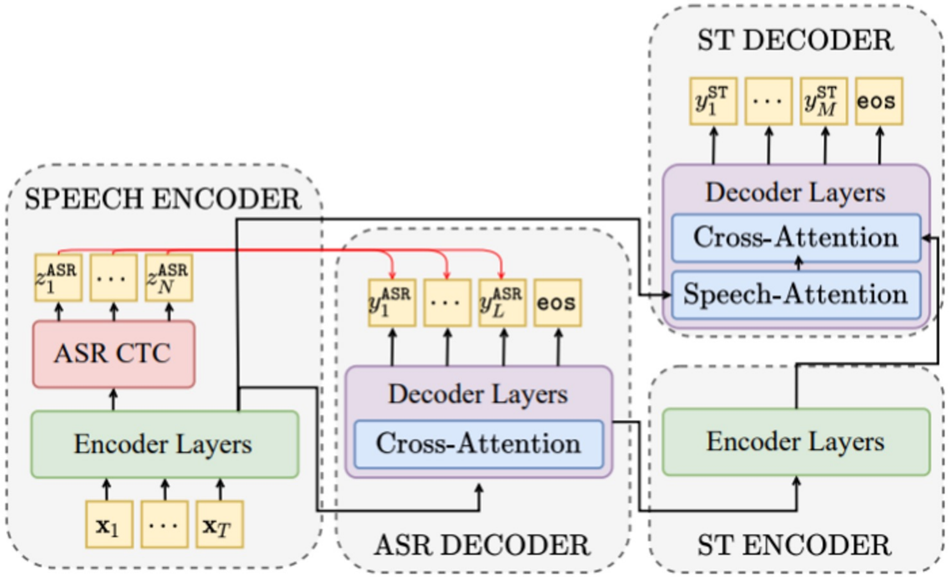
$$\mathbf{h}_j = \mathrm{Decoder}(\mathbf{h}_{j-1}, w_{j-1}, \mathrm{Encoder}(O))$$

  - ASR decoder state is ***differentiable*** (no argmax)
  - ASR decoder state is ***searchable***
    - During inference, w_{j-1} is obtained by search or fusion (beam search with a language model etc.)

    → We can incorporate various information with the decoder state $\mathbf{h}_j$
    - We call it ***Searchable Intermediates***
  - Note that the ASR encoder state **does not** have this property
    - $\mathbf{z}_t = \mathrm{Encoder}(O)$ does not have the token dependency

Multi-Decoder with Searchable Hidden Intermediates

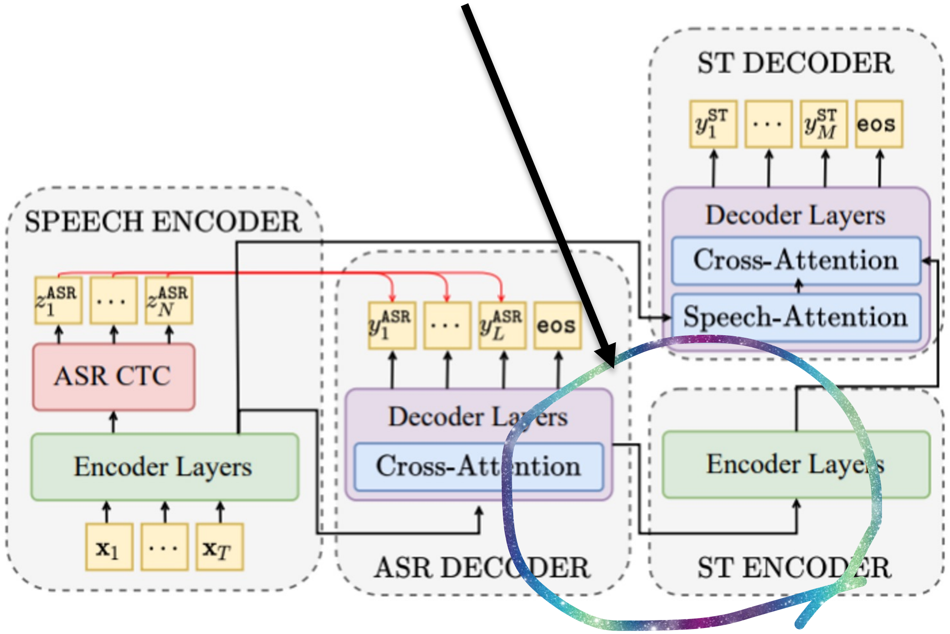
Searchable ASR Hidden Intermediates:
During inference, ASR decoder representations are retrieved (e.g. via beam search) and passed to the subsequent ST Encoder



$$\mathcal{L} = \lambda_1 \mathcal{L}_{\text{CE}}^{\text{ASR}} + \lambda_2 \mathcal{L}_{\text{CTC}}^{\text{ASR}} + \lambda_3 \mathcal{L}_{\text{CE}}^{\text{ST}}$$

Multi-Decoder with Searchable Hidden Intermediates

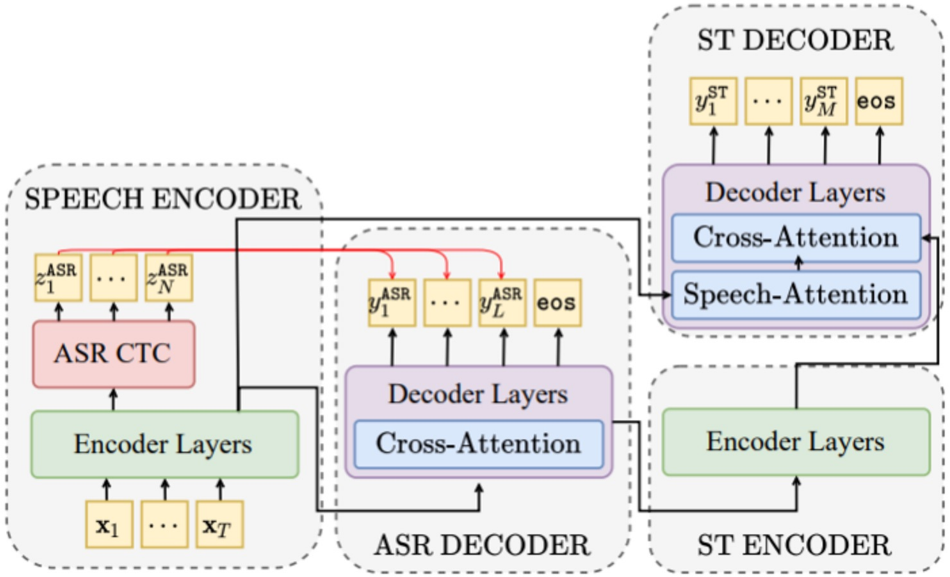
Searchable ASR Hidden Intermediates:
During inference, ASR decoder representations are retrieved (e.g. via beam search) and passed to the subsequent ST Encoder



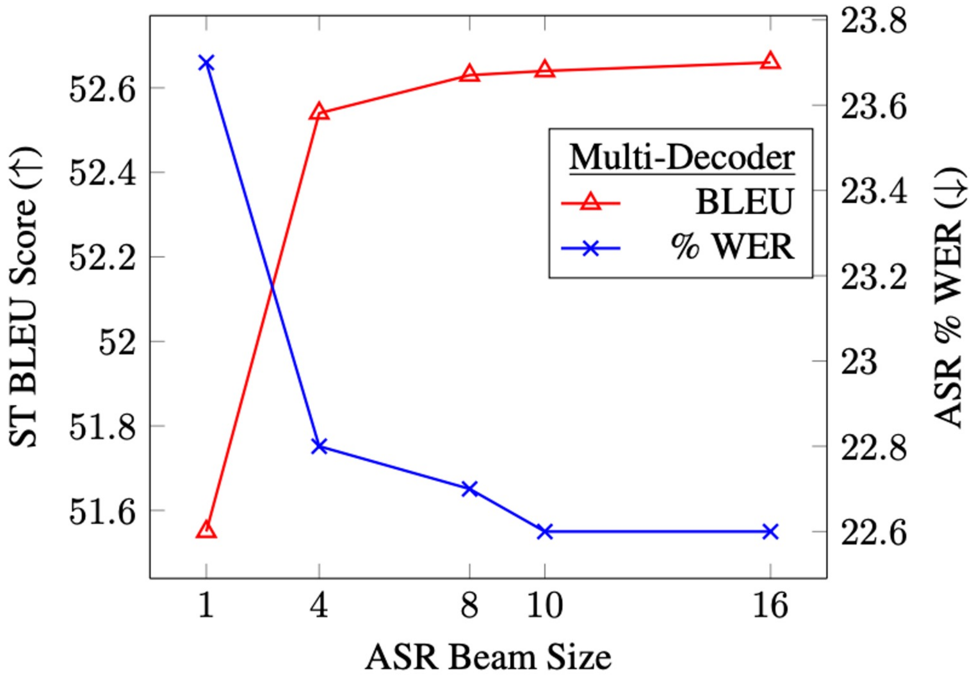
$$\mathcal{L} = \lambda_1 \mathcal{L}_{\text{CE}}^{\text{ASR}} + \lambda_2 \mathcal{L}_{\text{CTC}}^{\text{ASR}} + \lambda_3 \mathcal{L}_{\text{CE}}^{\text{ST}}$$

Multi-Decoder with Searchable Hidden Intermediates

Searchable ASR Hidden Intermediates:
During inference, ASR decoder representations are retrieved (e.g. via beam search) and passed to the subsequent ST Encoder



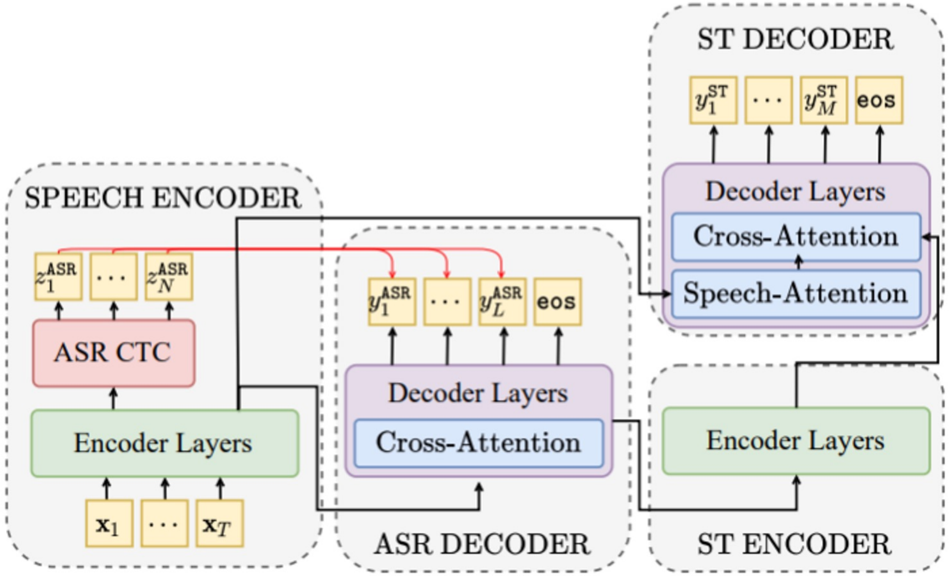
$$\mathcal{L} = \lambda_1 \mathcal{L}_{\text{CE}}^{\text{ASR}} + \lambda_2 \mathcal{L}_{\text{CTC}}^{\text{ASR}} + \lambda_3 \mathcal{L}_{\text{CE}}^{\text{ST}}$$



Better ASR → Better ST

Multi-Decoder with Searchable Hidden Intermediates

Searchable ASR Hidden Intermediates:
During inference, ASR decoder representations are retrieved (e.g. via beam search) and passed to the subsequent ST Encoder



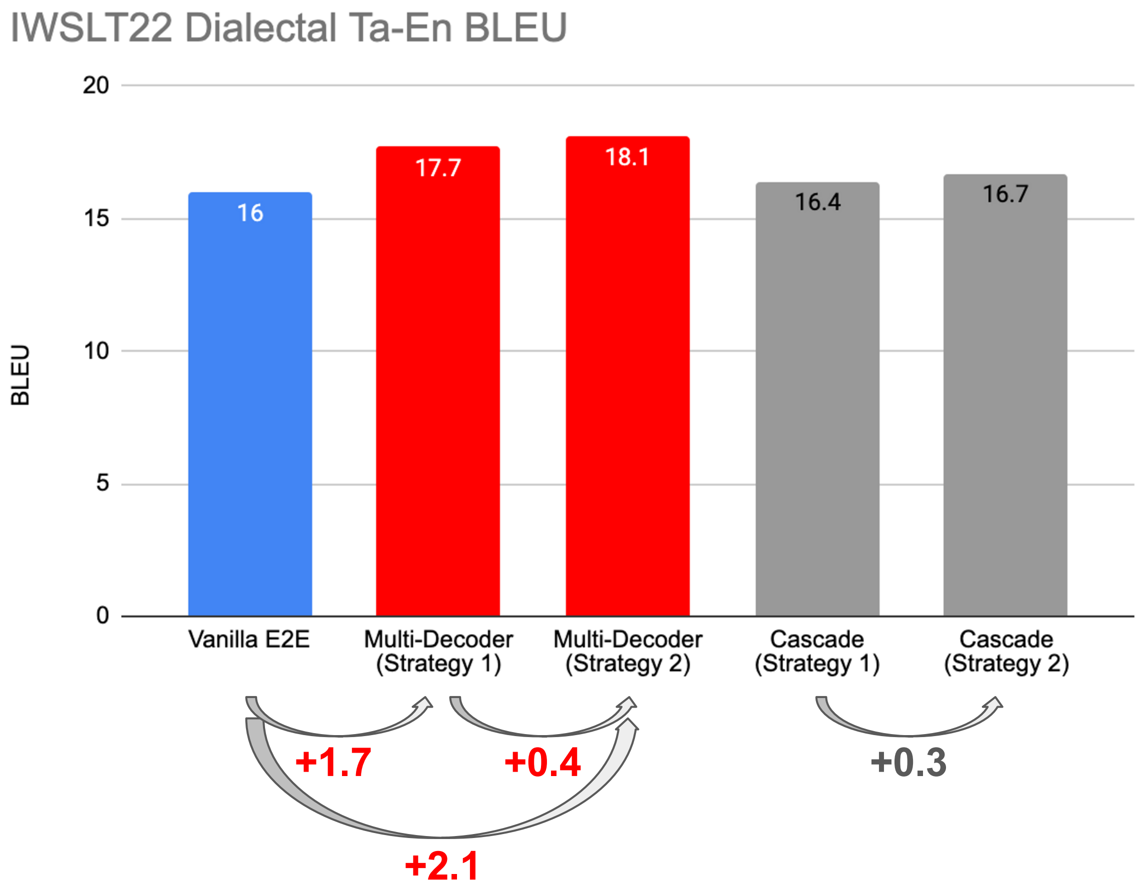
$$\mathcal{L} = \lambda_1 \mathcal{L}_{\text{CE}}^{\text{ASR}} + \lambda_2 \mathcal{L}_{\text{CTC}}^{\text{ASR}} + \lambda_3 \mathcal{L}_{\text{CE}}^{\text{ST}}$$

We can guide ASR hidden intermediate retrieval using external models!

Strategy 1:
Beam search over ASR output w/ use of external models (e.g. LM, CTC)

Strategy 2:
Use of post-processing to improve ASR output (e.g. ROVER ensembling)

Multi-Decoder with Searchable Hidden Intermediates



We can guide ASR hidden intermediate retrieval using external models!

Strategy 1:
Beam search over ASR output w/ use of external models (e.g. LM, CTC)

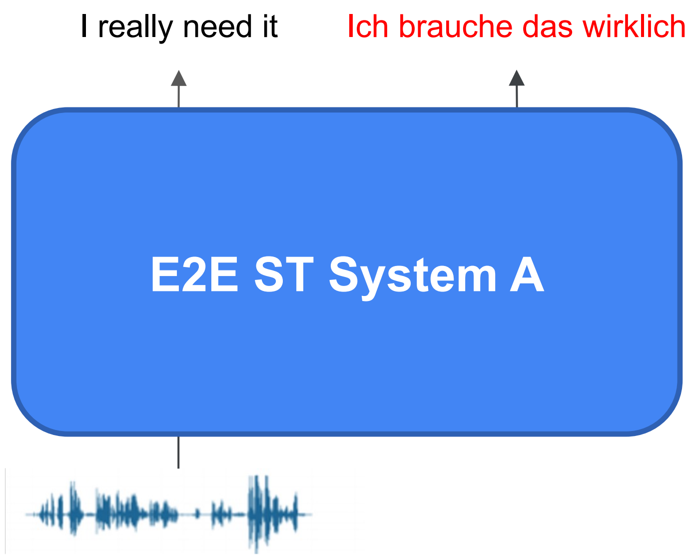
Strategy 2:
Use of post-processing to improve ASR output (e.g. ROVER ensembling)

Today's Talk

- CMU's IWSLT 2022 Dialect Speech Translation System
  - **Part 1:** Controlling ST output lengths via joint CTC/attention
  - **Part 2:** Controlling/explaining ST via searchable ASR intermediates

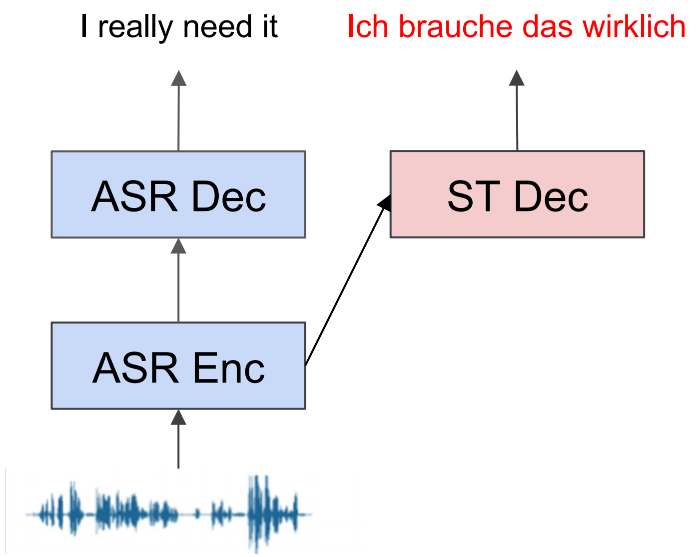
- Explainable E2E Speech Translation via Operation Sequence Generation
  - **Part 3:** Explaining ST via word-level ASR alignments

Which one is more explainable?



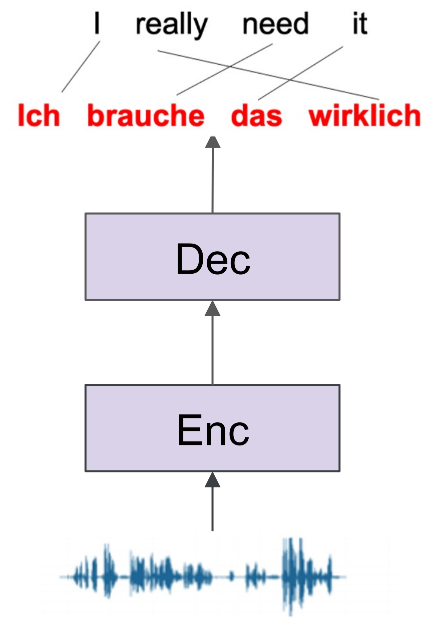
Word-Level Speech Translation Explanations

Vanilla E2E
(w/ ASR Multi-Task)



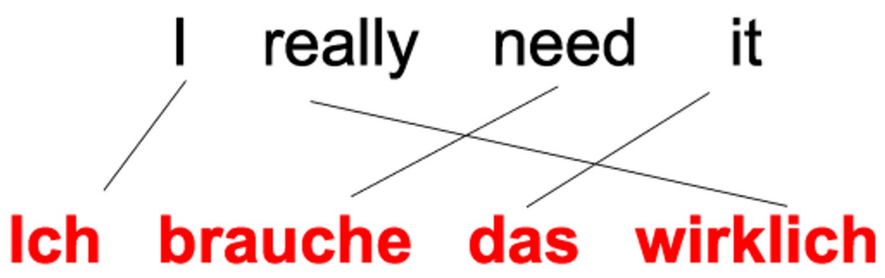
ASR and ST decodings
are independent / parallel

Explainable E2E



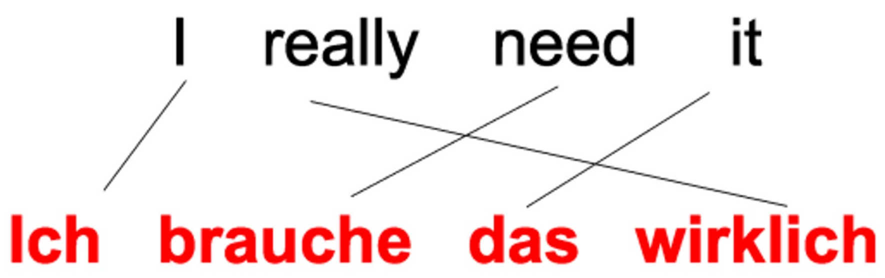
Can we simultaneously generate ASR/ST predictions + ***word-level alignments****?*

Align, Write, Re-order



Our goal is to get the speech translation result via this aligned information

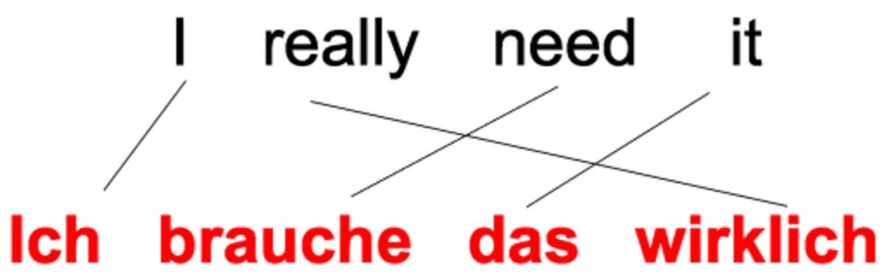
Align, Write, Re-order



(I, *[Position A], Ich*) (really, [Position B], wirklich) (need, [Position C], brauche) (it, [Position D], das)

Align to serialize

Align, Write, Re-order

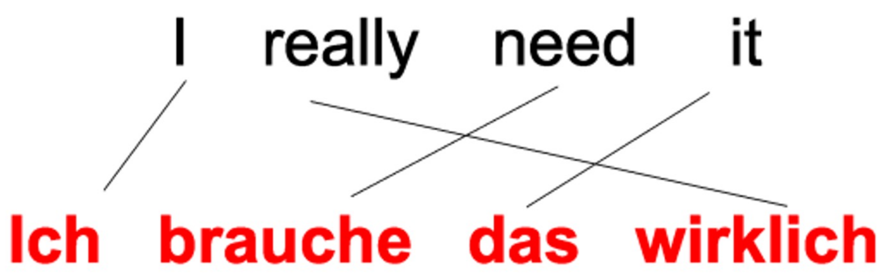


(I, [Position A], Ich) (really, [Position B], wirklich) (need, [Position C], brauche) (it, [Position D], das)

Ich wirklich brauche das

Write the aligned (synchronized) translation result

Align, Write, Re-order



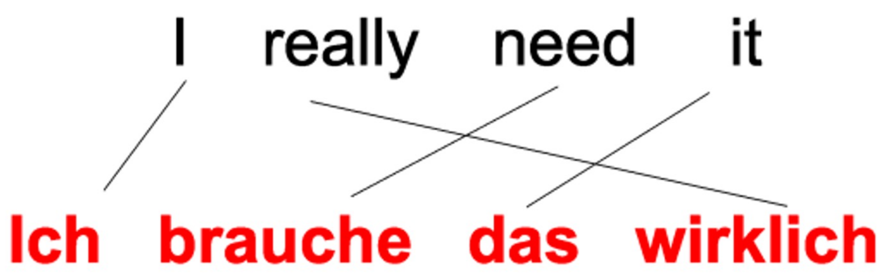
(I, [Position A], Ich) (really, [Position B], wirklich) (need, [Position C], brauche) (it, [Position D], das)

Ich wirklich brauche das

Ich *brauche* das wirklich

Reorder to get the final translation result

Align, Write, Re-order



(I, [Position A], Ich) (really, [Position B], wirklich) (need, [Position C], brauche) (it, [Position D], das)

Ich wirklich brauche das

Ich *brauche* das wirklich

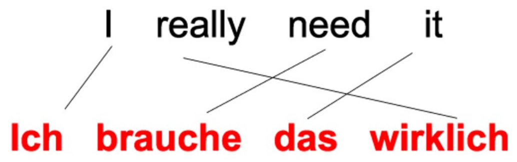
- **Explainable**
  - **Easy to analyze**
- **Streamable**
  - **Get the translation result as soon as we get the ASR result**

Operation Sequence Generation: Absolute Position

Objective: represent ASR/ST + word-alignment information as a single sequence

Strategy 1:

- Obtain word-level alignments on training data using statistical aligner (e.g. GIZA++)
- Define sequence of tuples of (source word, target word **absolute position,** target word)
- Insert target words into correct order in post-processing



(I, [0], Ich) (really, [3], wirklich) (need, [1], brauche) (it, [2], das)

Absolute position operation sequence

- By predicting target word absolute positions, we can generating translations **out-of-order** (while generating transcriptions **in-order**)

Operation Sequence Generation: Relative Shift

Strategy 2:

- Obtain word-level alignments on training data using statistical aligner (e.g. GIZA++)
- Define sequence of tuples of (source word, target word **relative shift**, target word)
- Insert target words into correct order in post-processing

Shifting Write-Head Operations
Based on prior MT work (Stahlberg et al., 2018)

[NO_OPS] - *no operation*
[SET_MARKER] - *place write-head marker*
[JMP_FWD] - *jump right to next write-head marker*
[JMP_BWD] - *jump left to prev write-head marker*
[NO_SRC] - *no aligned source word*
[NO_TGT] - *no aligned target word*
[EOP] - *end of tuple (not displayed for space)*

Raw System Output (Operation Sequence)
I [NO_OP] Ich have [NO_OP] habe spent [SET_MARKER] damit verbracht the [JMP_BWD] die

Transcription
I have spent the

Translation
Ich habe die damit verbracht *

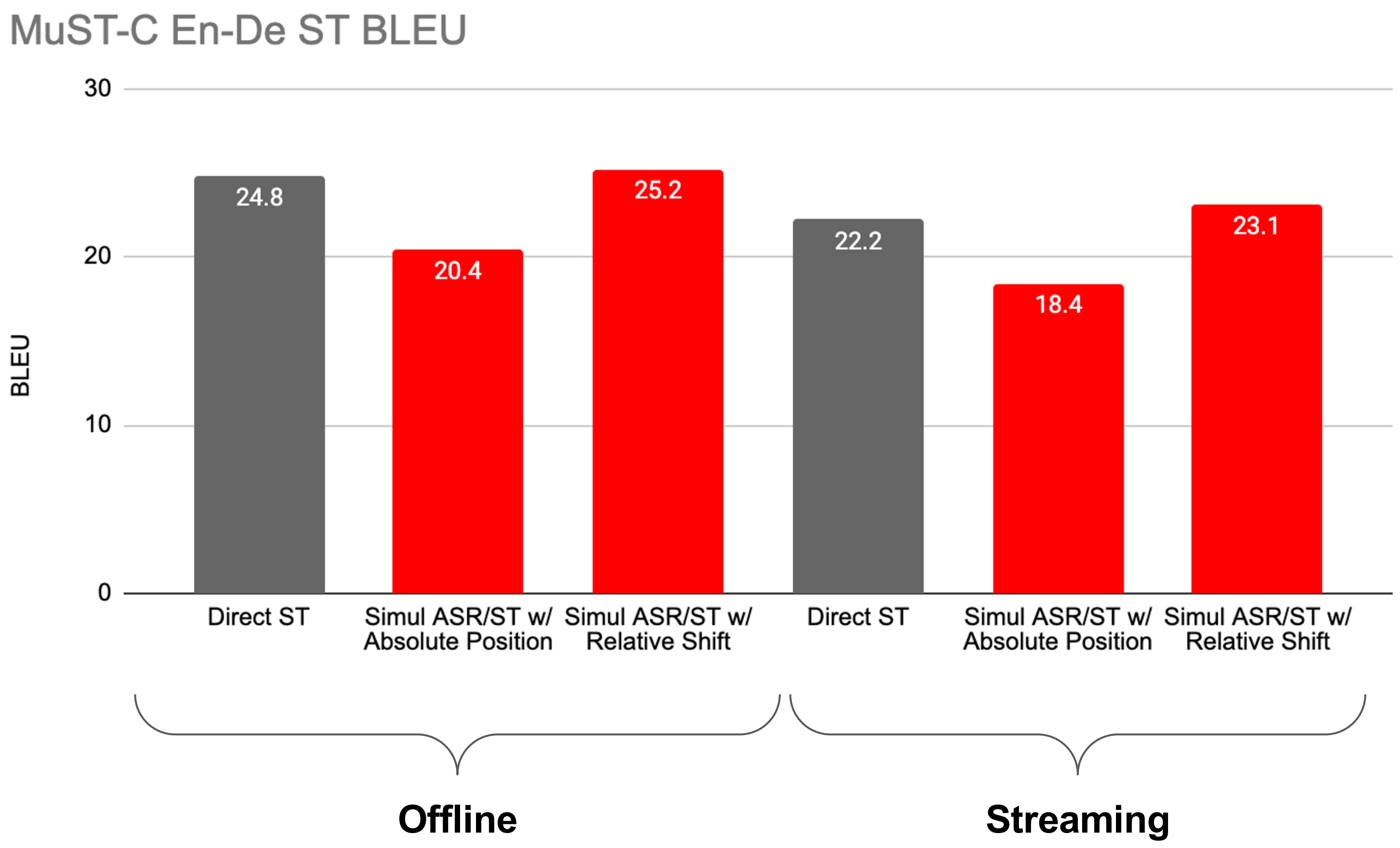
Special thanks to Prof. Graham Neubig

Operation Sequence Generation: Training and Inference

- **Training:**
  - After the data preparation, we ***just throw it*** to E2E ST Training (only data preparation)
- **Inference:**
  - ***just throw it*** to E2E ST beam search
  - Extract translation results with re-ordering

Demo: https://i.imgur.com/9MT5NoH.gifv

Operation Sequence Generation: Results



Absolute Position:
Difficult to generalize; performance lags behind direct ST models

Relative Shift:
On-par with direct ST models → achieves explainability without sacrificing performance!

Takeaways

End-to-end systems do not have to be black-boxes:

- **Part 1:** CTC alignments stabilize the length problem of autoregressive decoders
- **Part 2:** External model correction of searchable ASR intermediates improves ST
- **Part 3:** Word-level explainability does not sacrifice translation quality

We are putting them to ESPnet ESPnet

Let’s work together on Controllable and Explainable E2E Speech Translation!

Thank You!

human language technology
center of excellence

Carnegie Mellon University
Language Technologies Institute

YAHOO! JAPAN