PhD Defense

Model Architectures and Training Techniques for Multilingual Speech-to-Text Translation

Phuong-Hang Le

Advisors: **Didier Schwab** & **Benjamin Lecouteux**

March 25, 2024

Why study speech?

Why study speech?

Spoken language is the most prevalent and fundamental medium of human communication.

Why study speech?

Spoken language is the most prevalent and fundamental medium of human communication.

- Long predating written language

Why study speech?

Spoken language is the most prevalent and fundamental medium of human communication.

- Long predating written language
- 3 000 languages and dialects *without written forms*

Why study speech?

Spoken language is the most prevalent and fundamental medium of human communication.

- Long predating written language
- 3 000 languages and dialects *without written forms*
- Literacy rates below 50% in many countries (even with established writing systems).

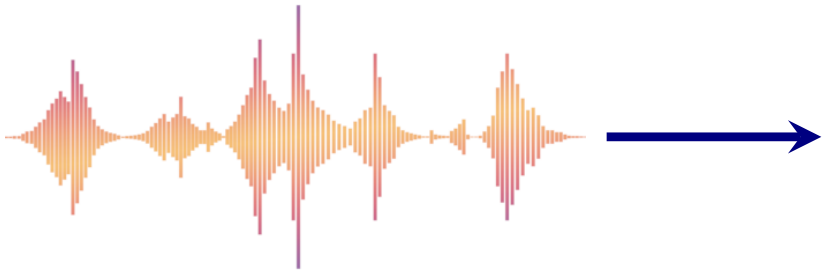
Why study speech?

Spoken language is the most prevalent and fundamental medium of human communication.

- Long predating written language
- 3 000 languages and dialects *without written forms*
- Literacy rates below 50% in many countries (even with established writing systems).

→ Spoken language has been one of the most important research topics in computational language processing.

Why multilingual speech-to-text translation?

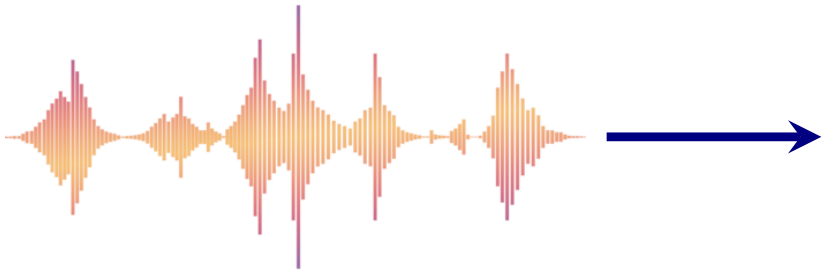


"*Bonjour le monde*"

Hello World

Speech-to-text translation **(ST)**

Why multilingual speech-to-text translation?



"*Bonjour le monde*"

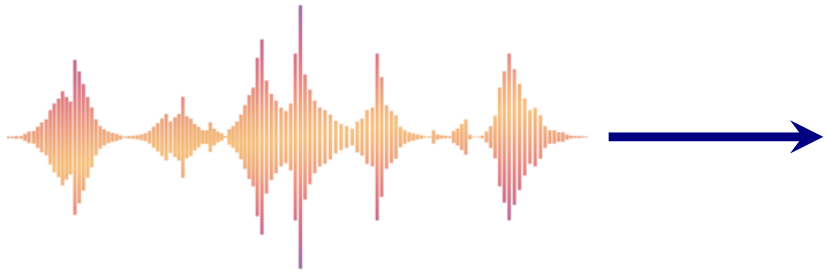
Hello World

Speech-to-text translation **(ST)**

Numerous real-world applications:

- Translations for people with hearing impairments
- Aiding professionals such as journalists and interpreters
- Voice assistants, customer support systems, etc.

Why multilingual speech-to-text translation?



"*Bonjour le monde*"

Hello World

Speech-to-text translation **(ST)**

Numerous real-world applications:

- Translations for people with hearing impairments
- Aiding professionals such as journalists and interpreters
- Voice assistants, customer support systems, etc.

Multilingual ST: A single system that can translate between multiple language pairs

- Facilitating communication across language barriers

Classical approach to speech-to-text translation

Classical approach: Cascaded systems of automatic speech recognition (ASR) and machine translation (MT).

- **ASR:** speech-to-text transcription (e.g., English audio to English text).
- **MT:** text-to-text translation (e.g., English text to French text).

Classical approach to speech-to-text translation

Classical approach: Cascaded systems of automatic speech recognition (ASR) and machine translation (MT).

- **ASR:** speech-to-text transcription (e.g., English audio to English text).
- **MT:** text-to-text translation (e.g., English text to French text).

Classical approach to speech-to-text translation

Classical approach: Cascaded systems of automatic speech recognition (ASR) and machine translation (MT).

- **ASR:** speech-to-text transcription (e.g., English audio to English text).
- **MT:** text-to-text translation (e.g., English text to French text).

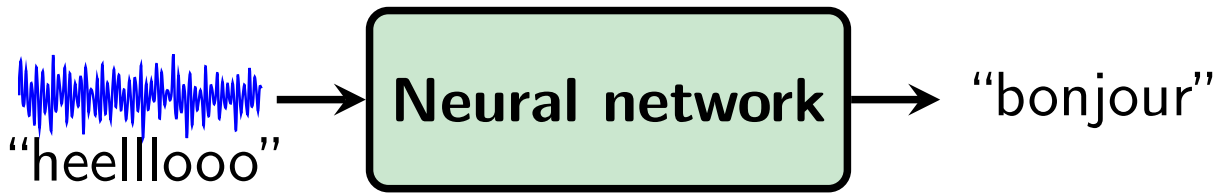
Major limitations:

✗ Error propagation

✗ High latency.

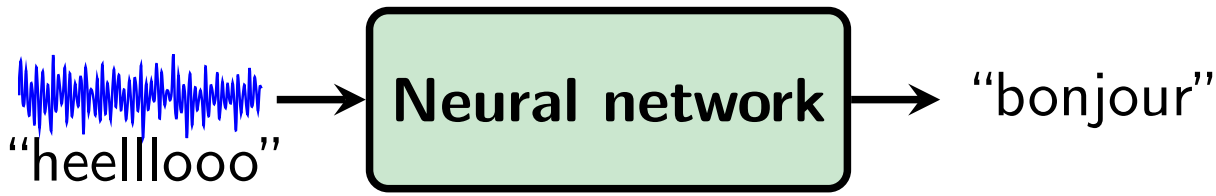
Modern approaches to speech-to-text translation

End-to-end models: Directly producing translation text without involving transcripts in-between.



Modern approaches to speech-to-text translation

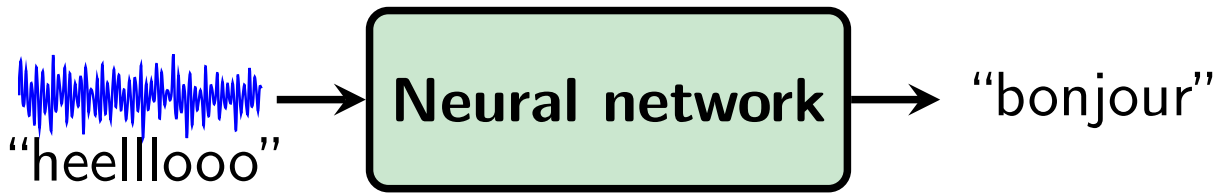
End-to-end models: Directly producing translation text without involving transcripts in-between.



First attempts [Berard et al., 2016, Duong et al., 2016] achieved promising results.

Modern approaches to speech-to-text translation

End-to-end models: Directly producing translation text without involving transcripts in-between.



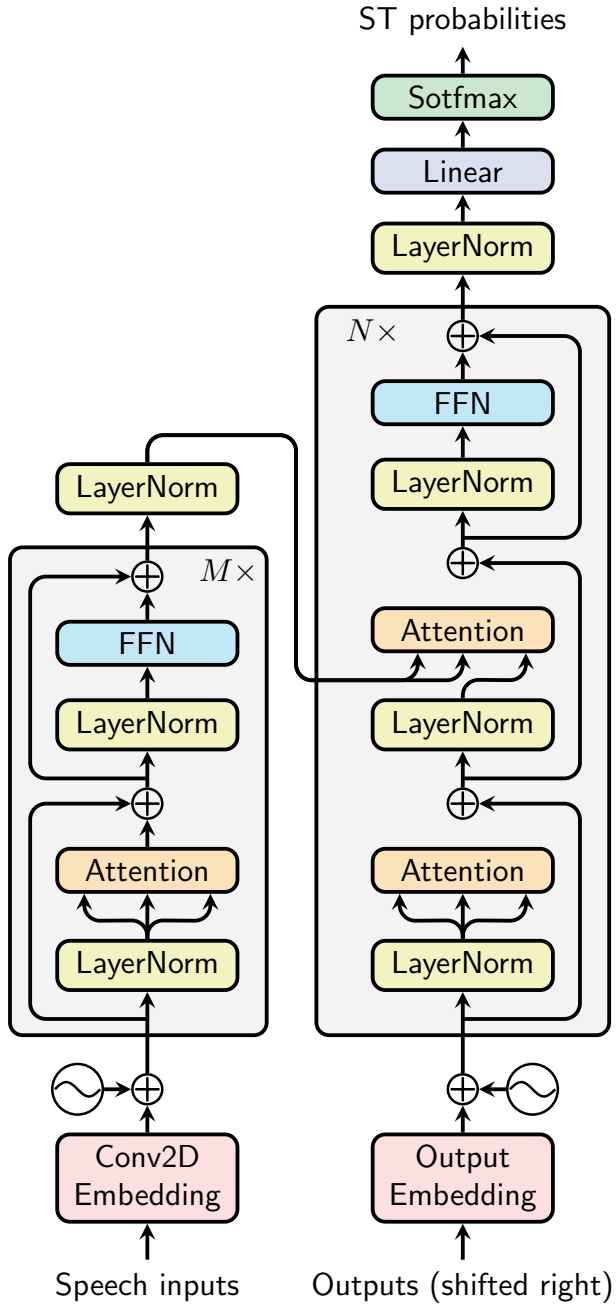
First attempts [Berard et al., 2016, Duong et al., 2016] achieved promising results.

But quickly became the dominant approach thanks to advances in two major research areas:

Model Architectures and **Training Techniques**.

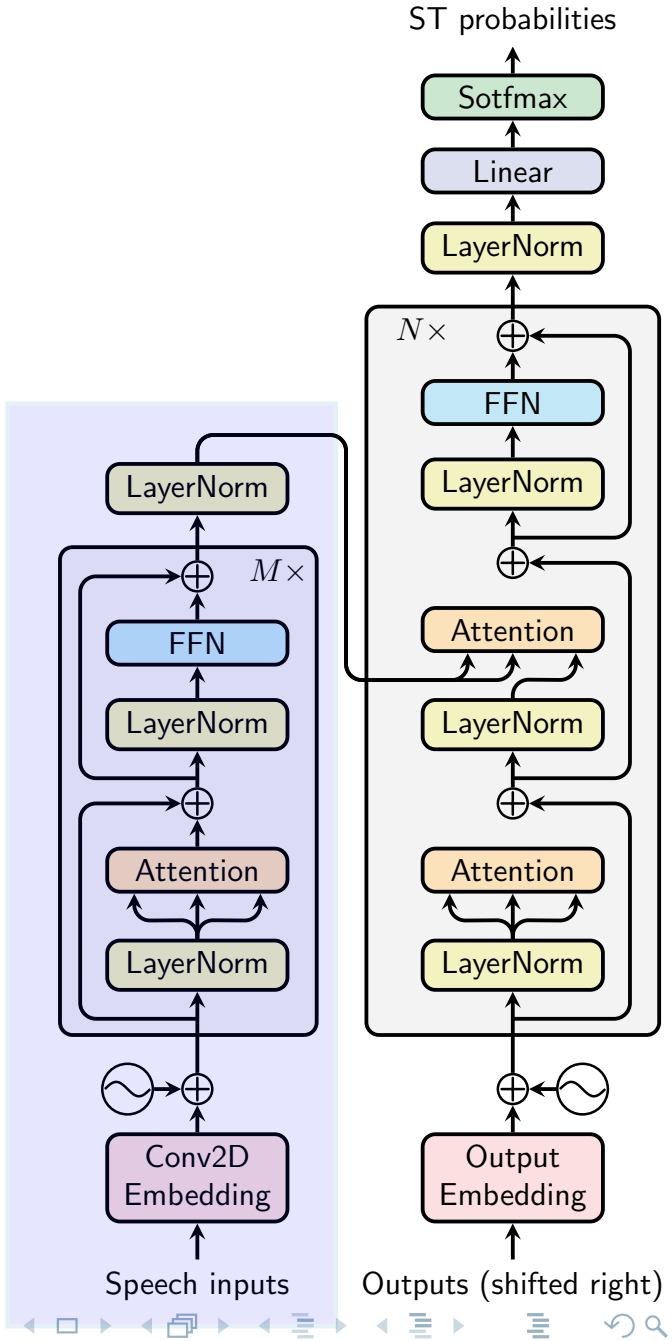
Modern approaches: Model architectures

Dominant models are variants of **the Transformer** [Vaswani et al., 2017].



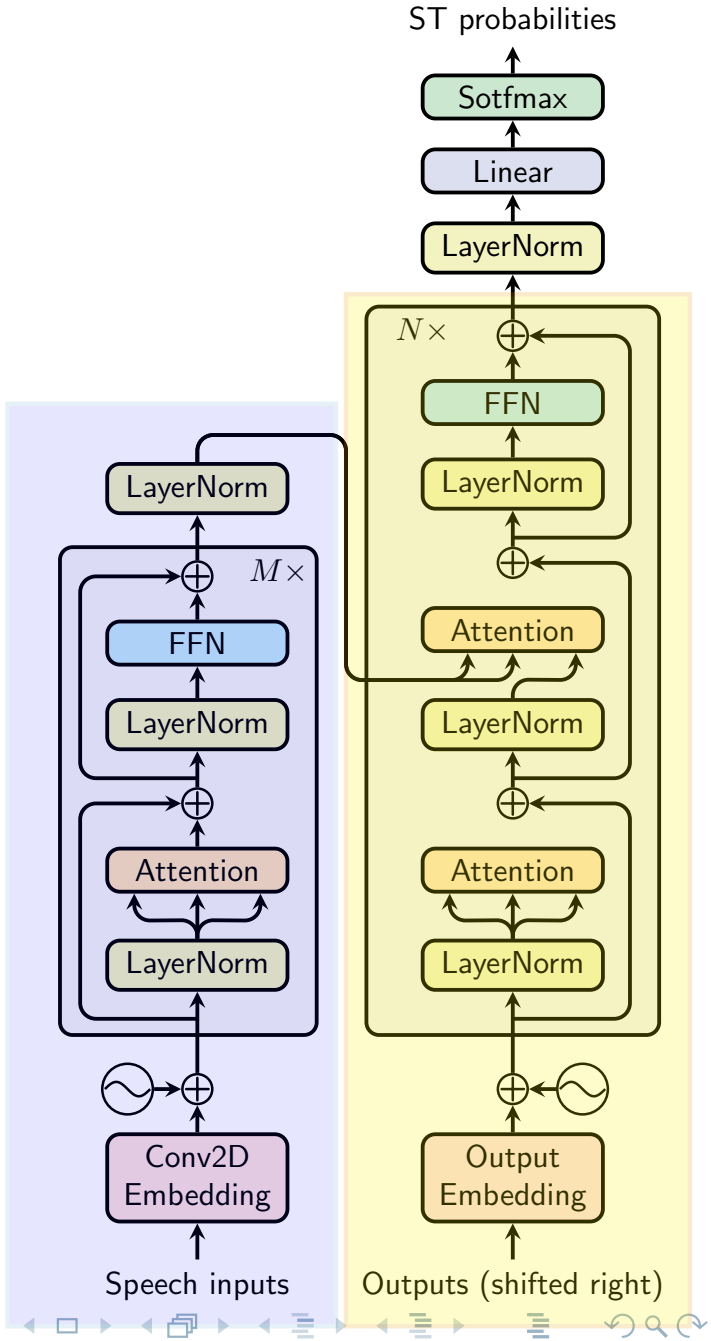
Modern approaches: Model architectures

Dominant models are variants of **the Transformer** [Vaswani et al., 2017].



Modern approaches: Model architectures

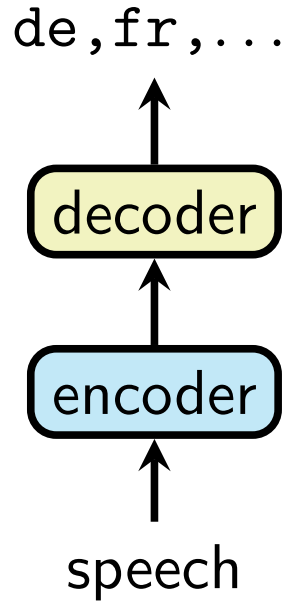
Dominant models are variants of **the Transformer** [Vaswani et al., 2017].



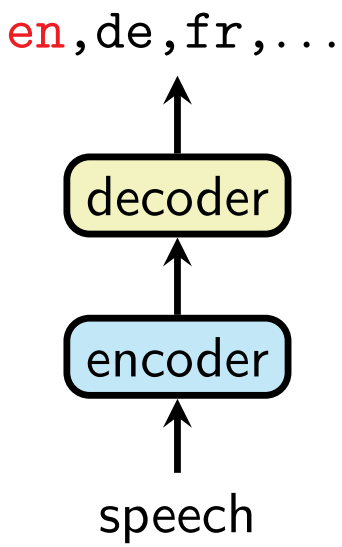
Modern approaches: Model architectures

Dominant models are variants of **the Transformer** [Vaswani et al., 2017].

At a high-level: Consisting of a speech encoder and a text decoder.

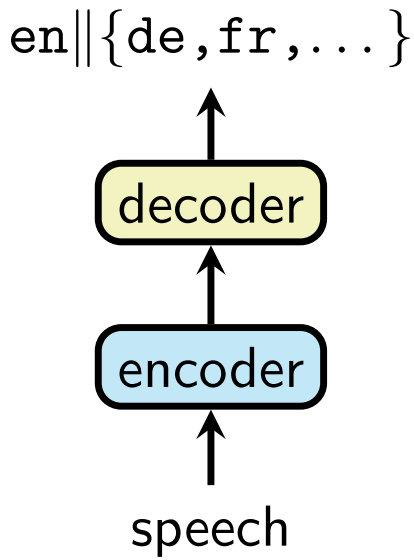


Modern approaches: Model architectures



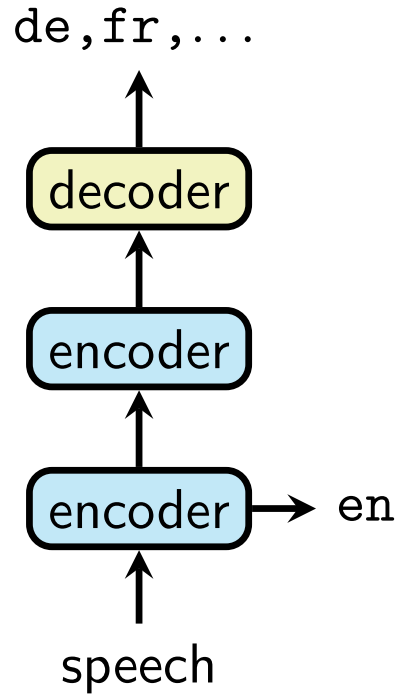
English as additional language

[Gangi et al., 2019]



Concatenated transcript

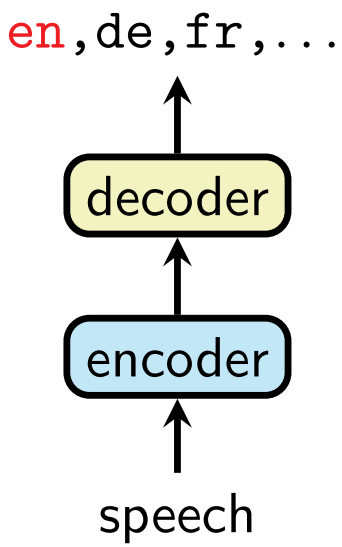
[Sperber et al., 2020, Dong et al., 2021a]



Cascaded encoders

[Dong et al., 2021b, Xu et al., 2021]

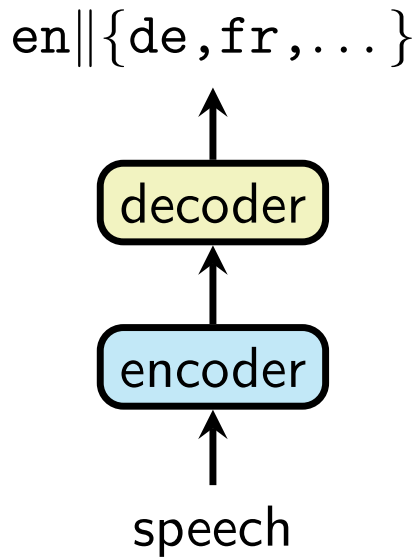
Modern approaches: Model architectures



English as additional language

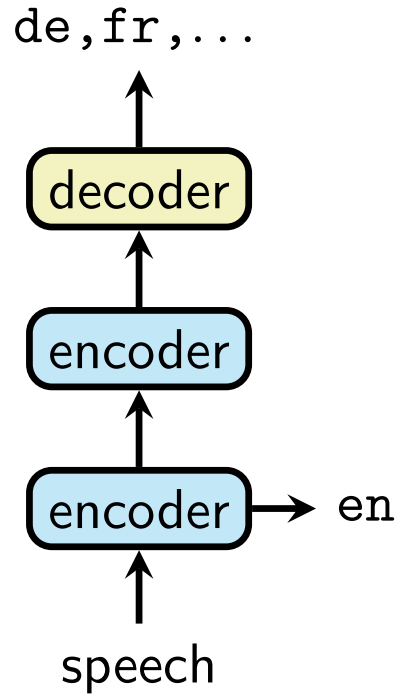
[Gangi et al., 2019]

simply using transcripts
as an additional language



Concatenated transcript

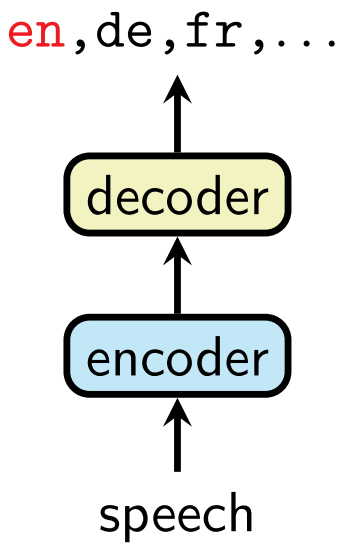
[Sperber et al., 2020, Dong et al., 2021a]



Cascaded encoders

[Dong et al., 2021b, Xu et al., 2021]

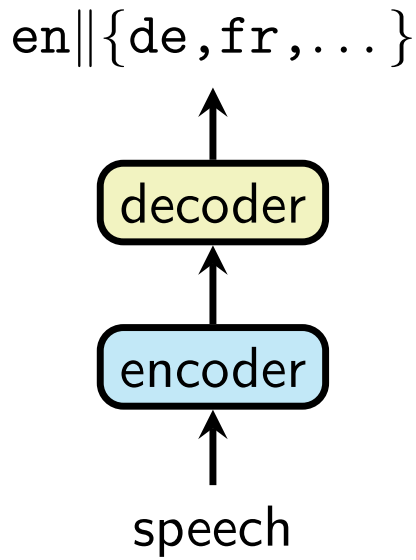
Modern approaches: Model architectures



English as additional language

[Gangi et al., 2019]

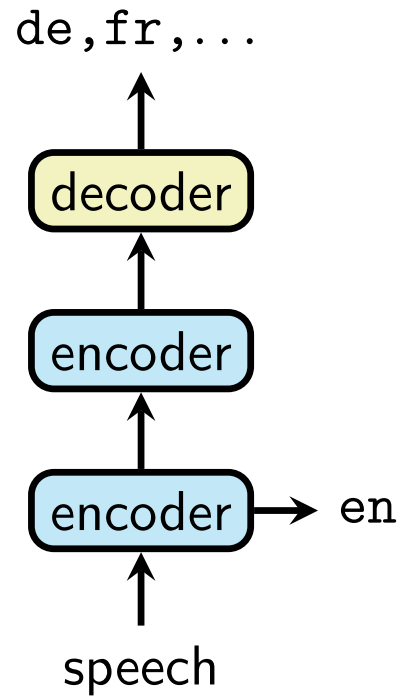
simply using transcripts as an additional language



Concatenated transcript

[Sperber et al., 2020, Dong et al., 2021a]

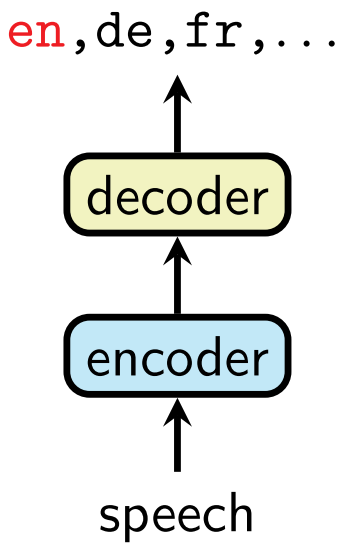
concatenate transcript with translation, and let the model predicts the concatenated text



Cascaded encoders

[Dong et al., 2021b, Xu et al., 2021]

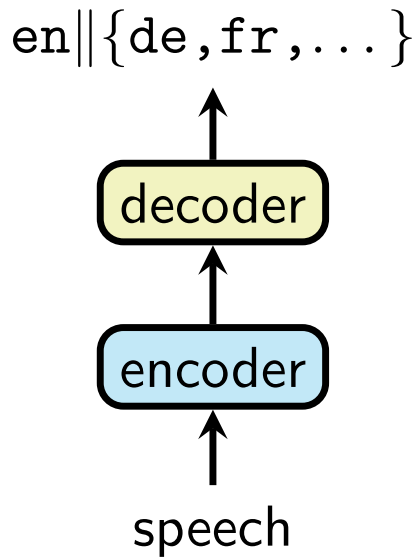
Modern approaches: Model architectures



English as additional language

[Gangi et al., 2019]

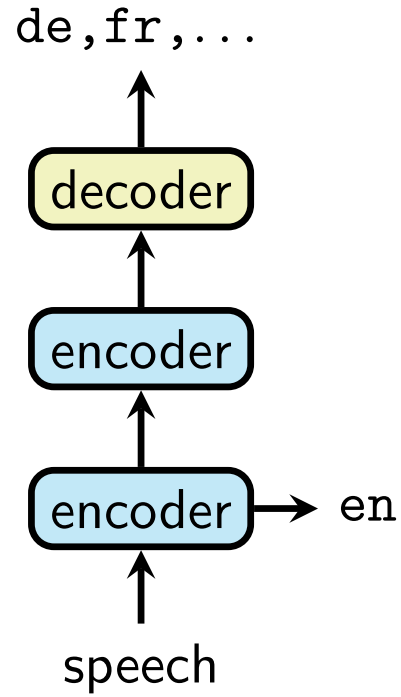
simply using transcripts as an additional language



Concatenated transcript

[Sperber et al., 2020, Dong et al., 2021a]

concatenate transcript with translation, and let the model predicts the concatenated text

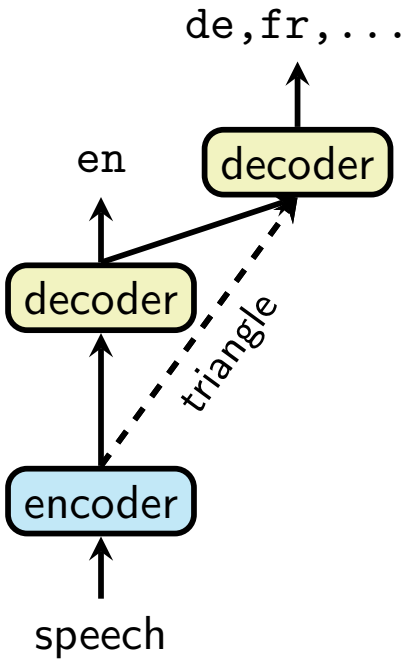


Cascaded encoders

[Dong et al., 2021b, Xu et al., 2021]

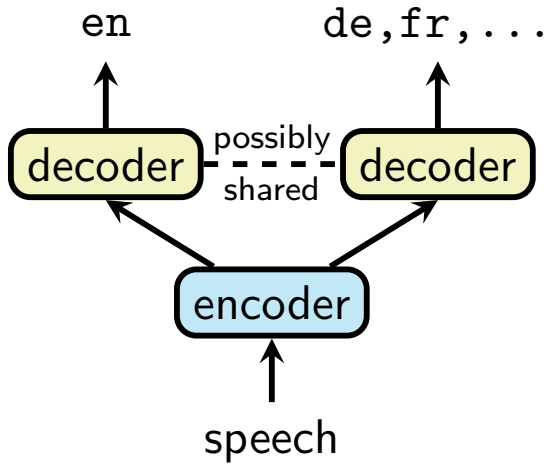
encoders are cascaded to ease the modelling task

Modern approaches: Model architectures



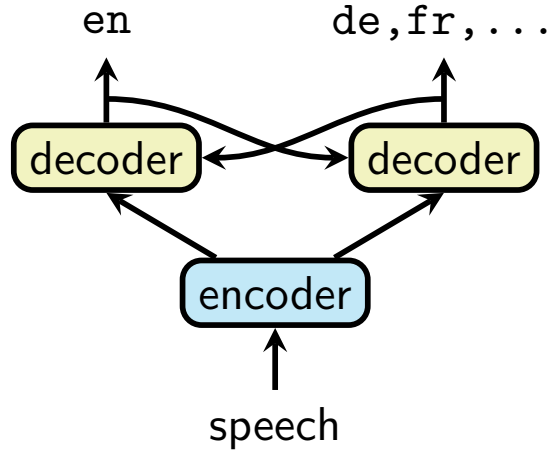
Two-stage or triangle

[Anastasopoulos and Chiang, 2018, Sperber et al., 2019]



Independent decoders

[Weiss et al., 2017, Anastasopoulos and Chiang, 2018]



Interactive decoding

[Liu et al., 2020]

Modern approaches: Model architectures

Common theme of the above models: Employing recognition task (ASR) to help translation task (ST), in a **multi-task learning** fashion.

Question:

Can we make multi-task modelling more effective for ST?

Modern approaches: Training techniques

Modern approaches: Training techniques

Two major paradigms:

- **Multi-task learning**.
- **Pre-training** then **fine-tuning**

Modern approaches: Training techniques

Multi-task learning: Training ST jointly with auxiliary tasks (e.g., ASR or MT).

Modern approaches: Training techniques

Multi-task learning: Training ST jointly with auxiliary tasks (e.g., ASR or MT).

Pre-training:

- Unsupervised pre-training: Pre-training on raw text or audio without *labeled* data [Schneider et al., 2019, Baevski et al., 2020a,b, Bapna et al., 2021, Tang et al., 2022]

Modern approaches: Training techniques

Multi-task learning: Training ST jointly with auxiliary tasks (e.g., ASR or MT).

Pre-training:

- Unsupervised pre-training: Pre-training on raw text or audio without *labeled* data [Schneider et al., 2019, Baevski et al., 2020a,b, Bapna et al., 2021, Tang et al., 2022]
- Supervised pre-training: Pre-training auxiliary tasks (typically ASR or MT) on *labeled* data [Bérard et al., 2018, Bansal et al., 2019, Wang et al., 2020a, Inaguma et al., 2020].

Modern approaches: Training techniques

Multi-task learning: Training ST jointly with auxiliary tasks (e.g., ASR or MT).

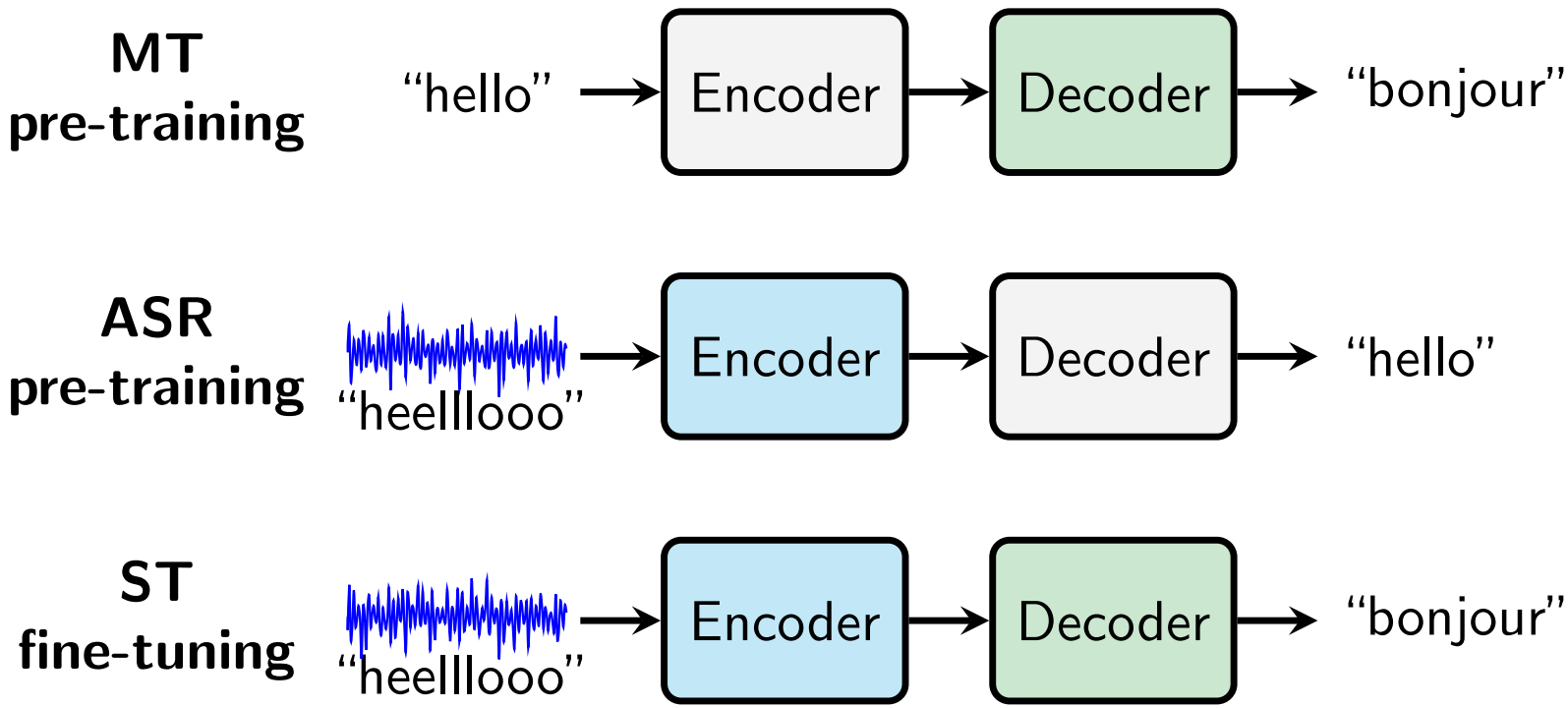
Pre-training:

- Unsupervised pre-training: Pre-training on raw text or audio without *labeled* data [Schneider et al., 2019, Baevski et al., 2020a,b, Bapna et al., 2021, Tang et al., 2022]
- Supervised pre-training: Pre-training auxiliary tasks (typically ASR or MT) on *labeled* data [Bérard et al., 2018, Bansal et al., 2019, Wang et al., 2020a, Inaguma et al., 2020].

Fine-tuning: Initializing ST training with pre-trained components.

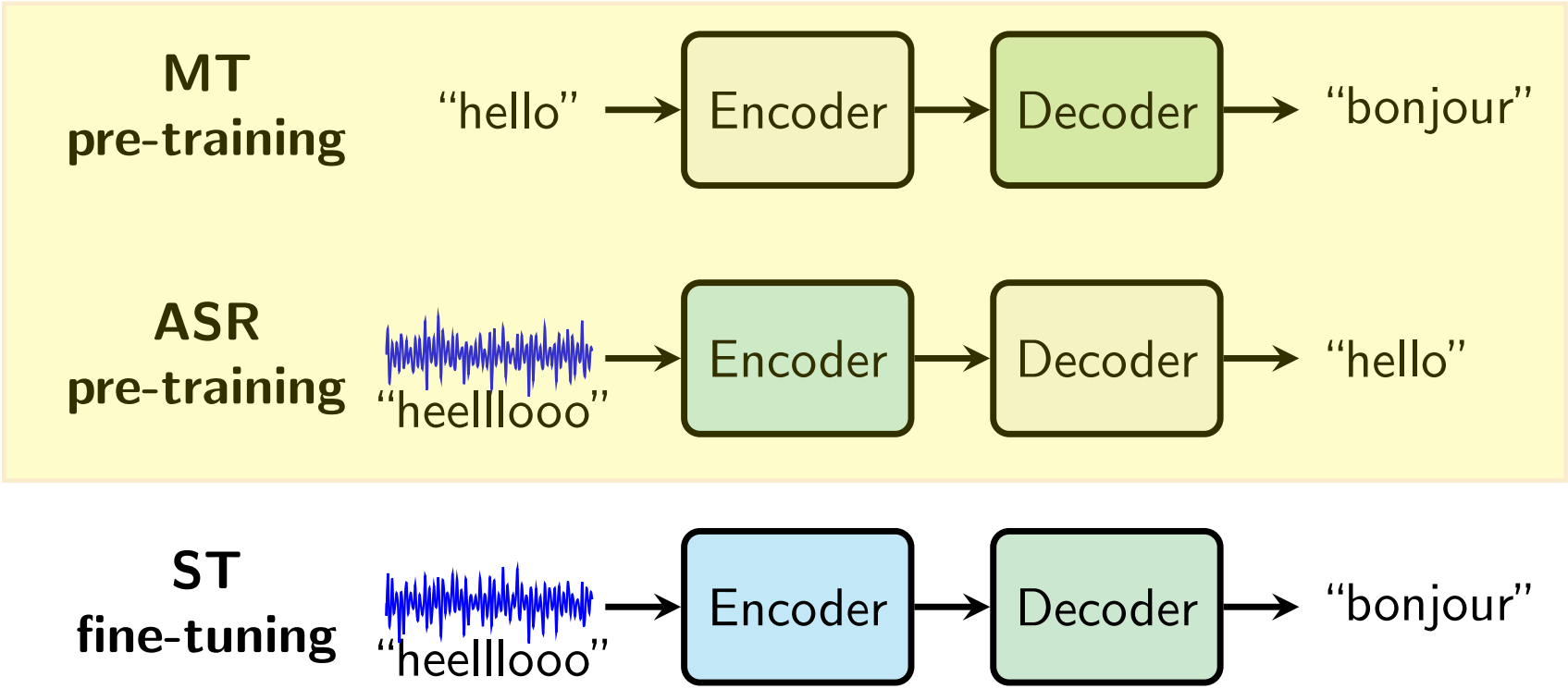
Modern approaches: Training techniques

Example of supervised pre-training and fine-tuning



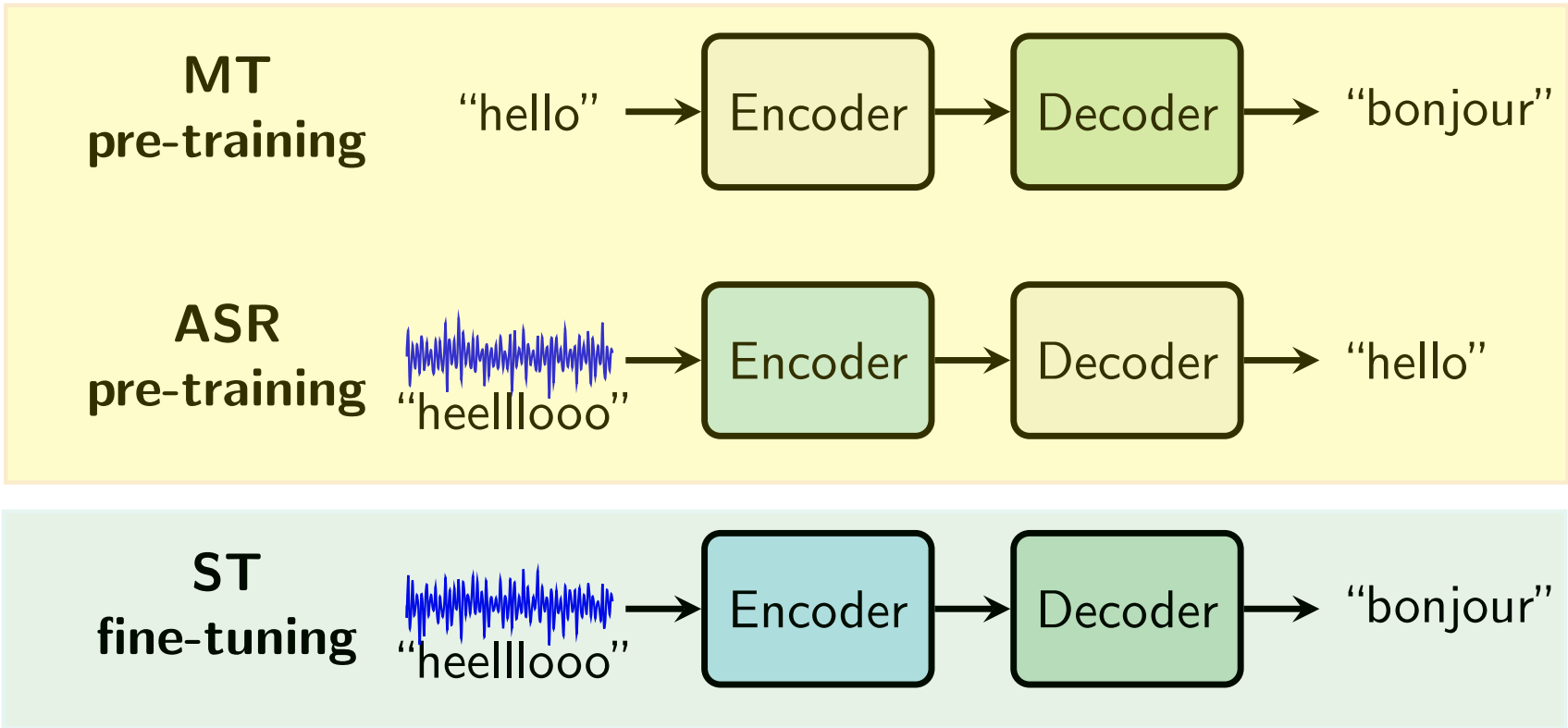
Modern approaches: Training techniques

Example of supervised pre-training and fine-tuning



Modern approaches: Training techniques

Example of supervised pre-training and fine-tuning



Modern approaches: Training techniques

Question:

Can we make pre-training and fine-tuning more effective for ST?

Thesis contributions

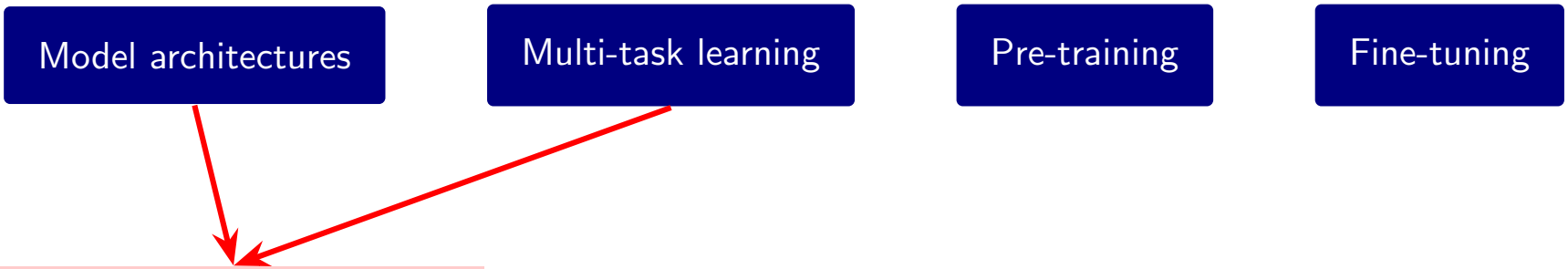
Model architectures

Multi-task learning

Pre-training

Fine-tuning

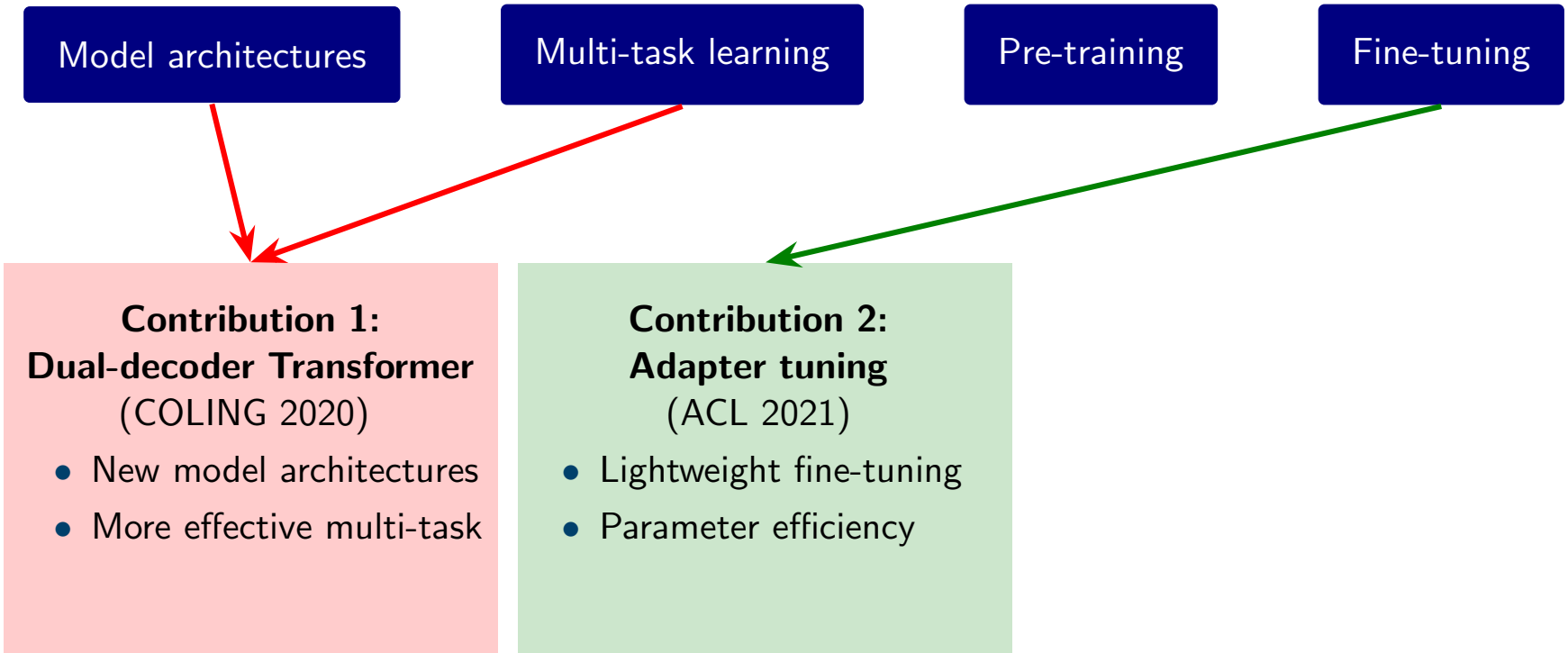
Thesis contributions



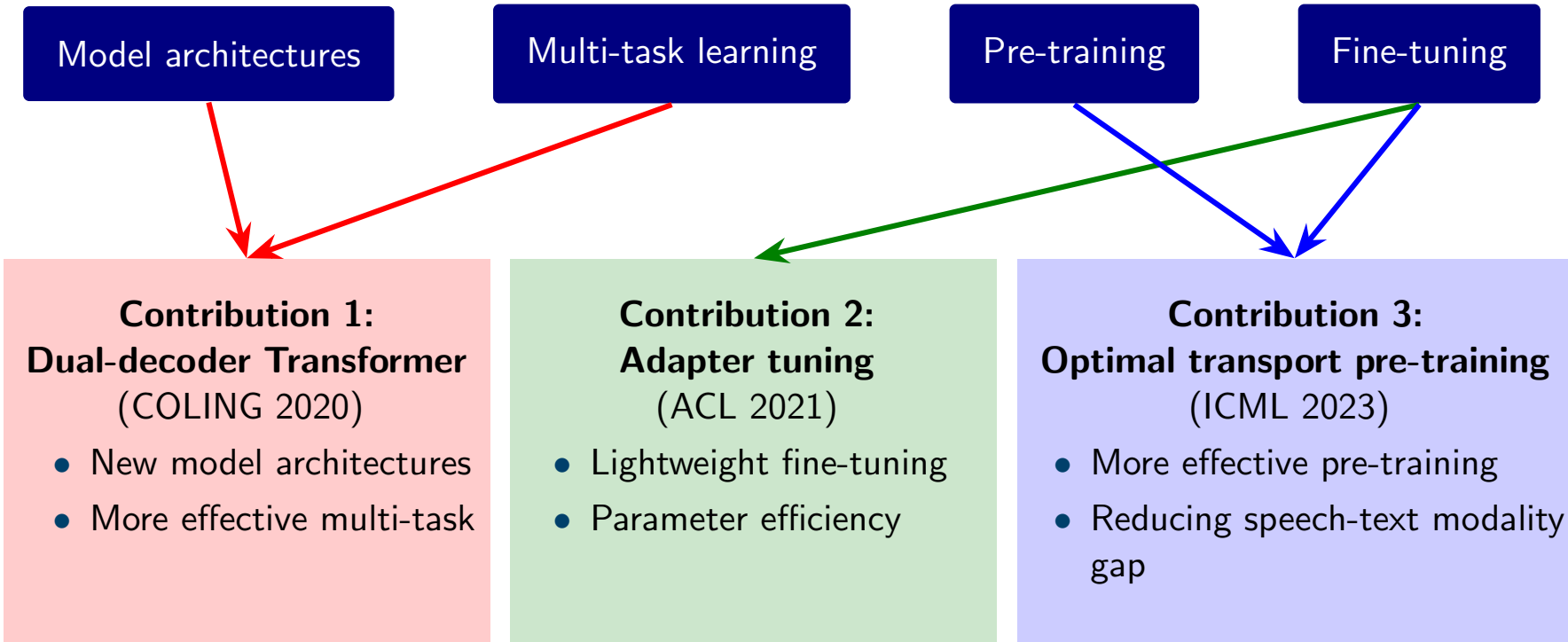
Contribution 1:
Dual-decoder Transformer
(COLING 2020)

- New model architectures
- More effective multi-task

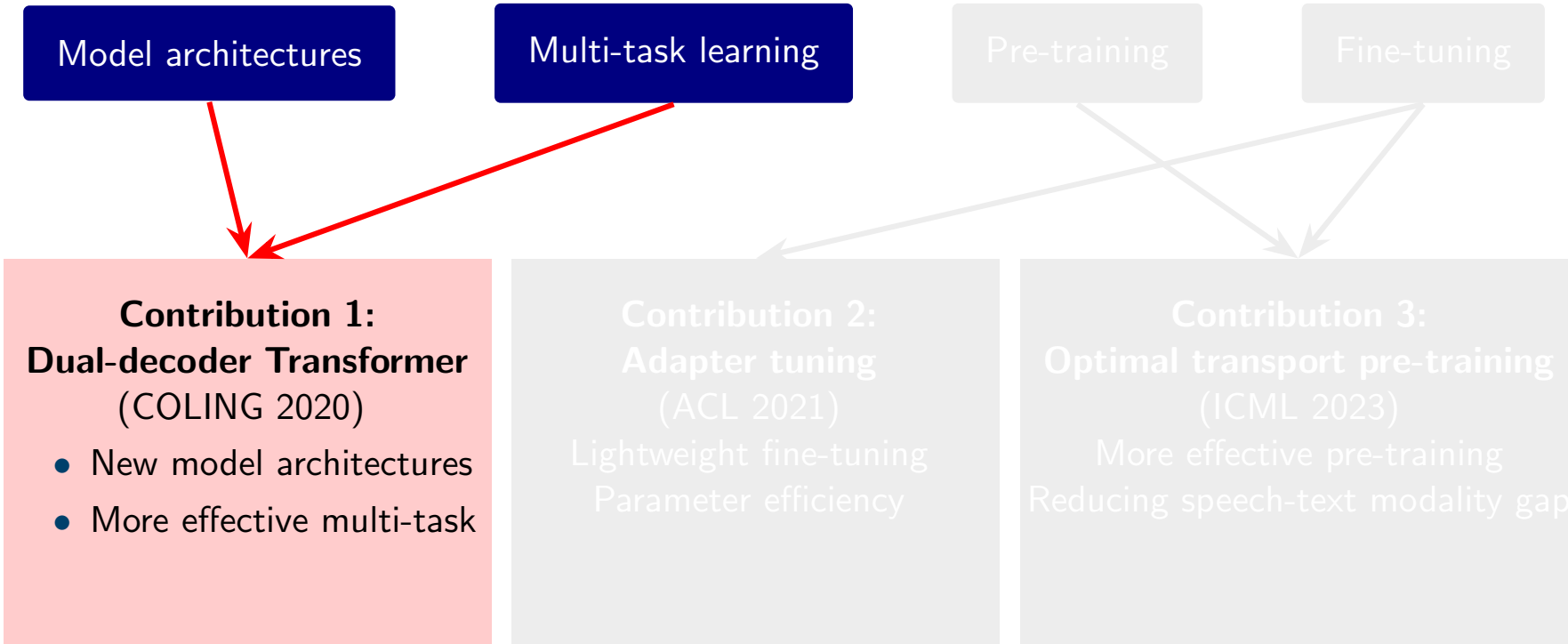
Thesis contributions



Thesis contributions



Contribution 1: Dual-decoder Transformer

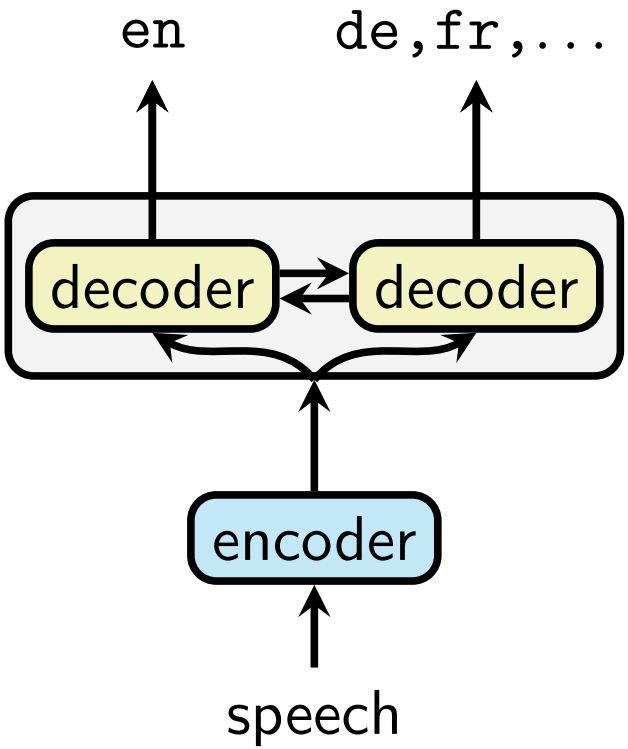


Hang Le, Juan Pino, Changhan Wang, Jiatao Gu, Didier Schwab, Laurent Besacier. Dual-decoder Transformer for Joint Automatic Speech Recognition and Multilingual Speech Translation. In *International Conference on Computational Linguistics* (**COLING 2020, oral**).

Dual-decoder Transformer: Motivation

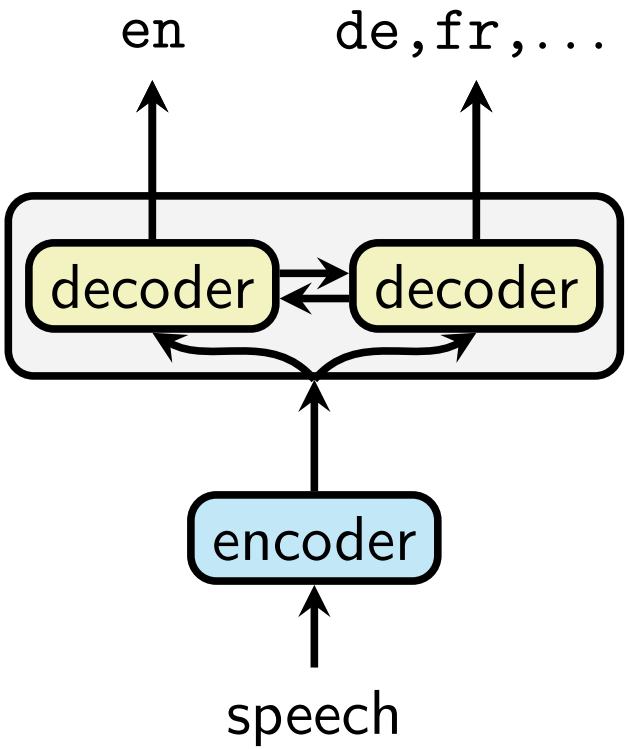
- Hypothesis: **ASR and ST are complementary to each other**
- Target task: **joint ASR and ST**
- Applications: display of transcripts alongside translation is helpful

Dual-decoder Transformer: High-level design



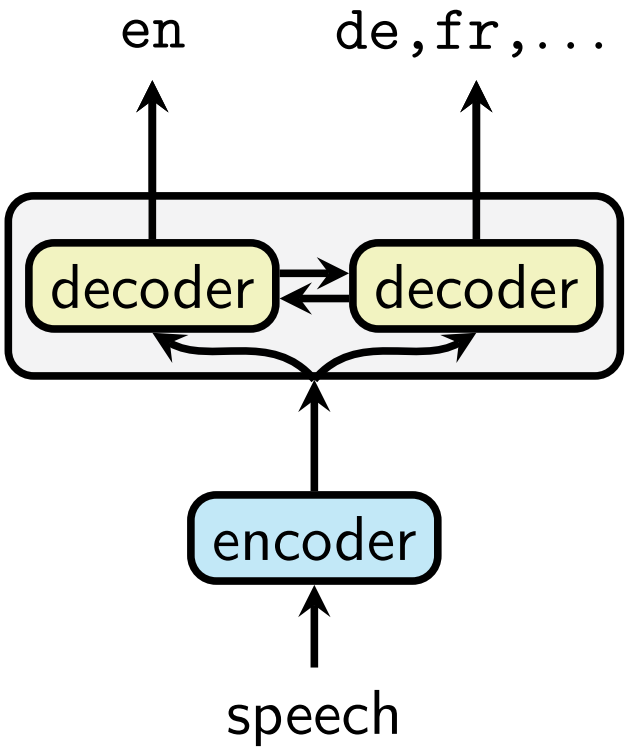
- Two common Transformer decoders:
  - One performs transcription (ASR)
  - One performs translation (ST)

Dual-decoder Transformer: High-level design



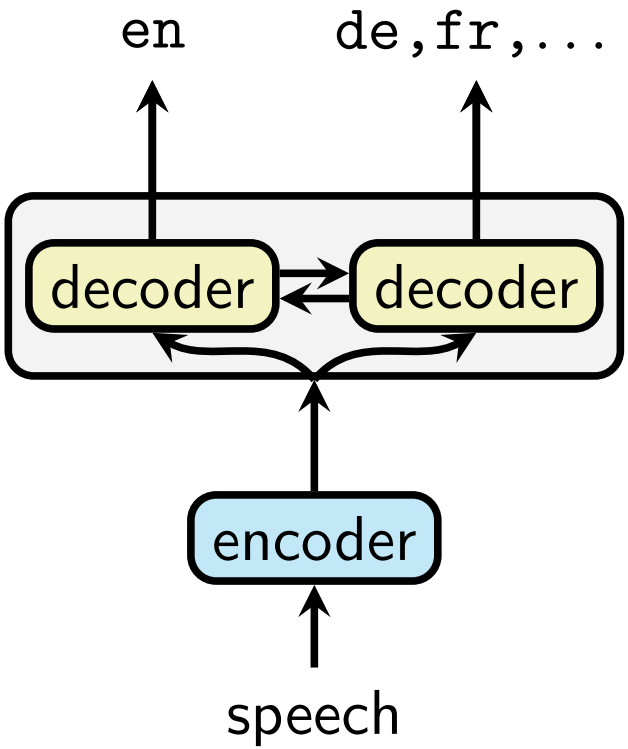
- Two common Transformer decoders:
  - One performs transcription (ASR)
  - One performs translation (ST)
- Flexible: level of interaction between decoders is a design choice.

Dual-decoder Transformer: High-level design



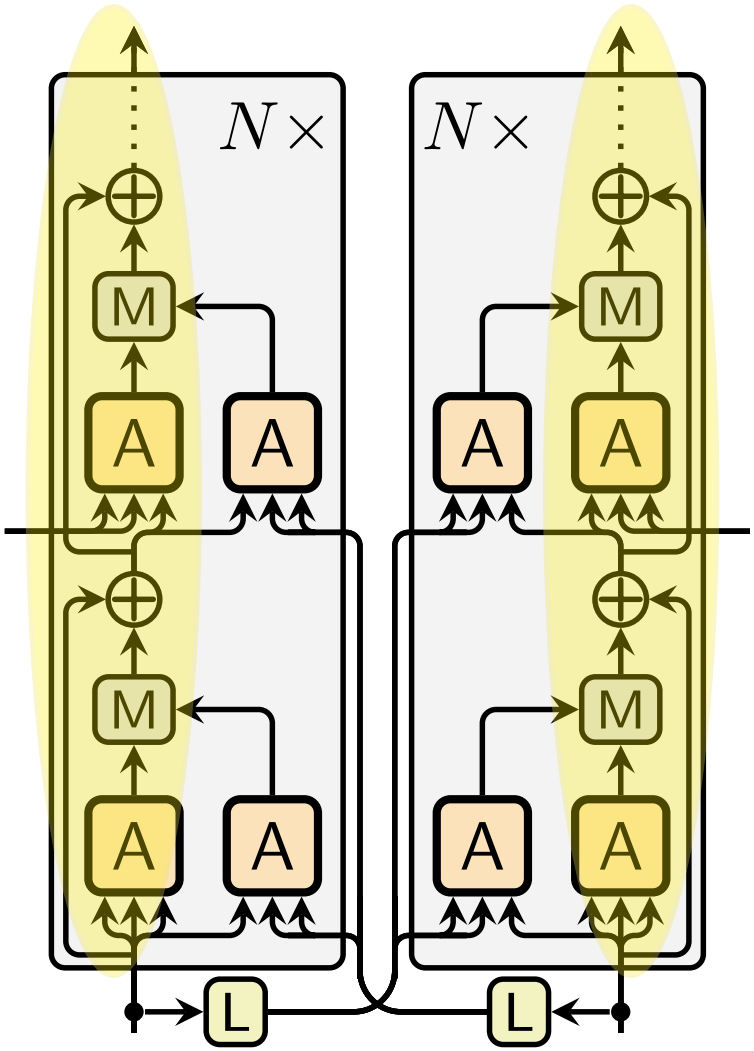
- Two common Transformer decoders:
  - One performs transcription (ASR)
  - One performs translation (ST)
- Flexible: level of interaction between decoders is a design choice.
- Can be seen as a general framework that includes several existing models as special cases.

Dual-decoder Transformer: High-level design



- Two common Transformer decoders:
  - One performs transcription (ASR)
  - One performs translation (ST)
- Flexible: level of interaction between decoders is a design choice.
- Can be seen as a general framework that includes several existing models as special cases.
- Two new variants are proposed:
  - **Cross dual-decoder Transformer**
  - **Parallel dual-decoder Transformer**

First variant: Cross dual-decoder Transformer



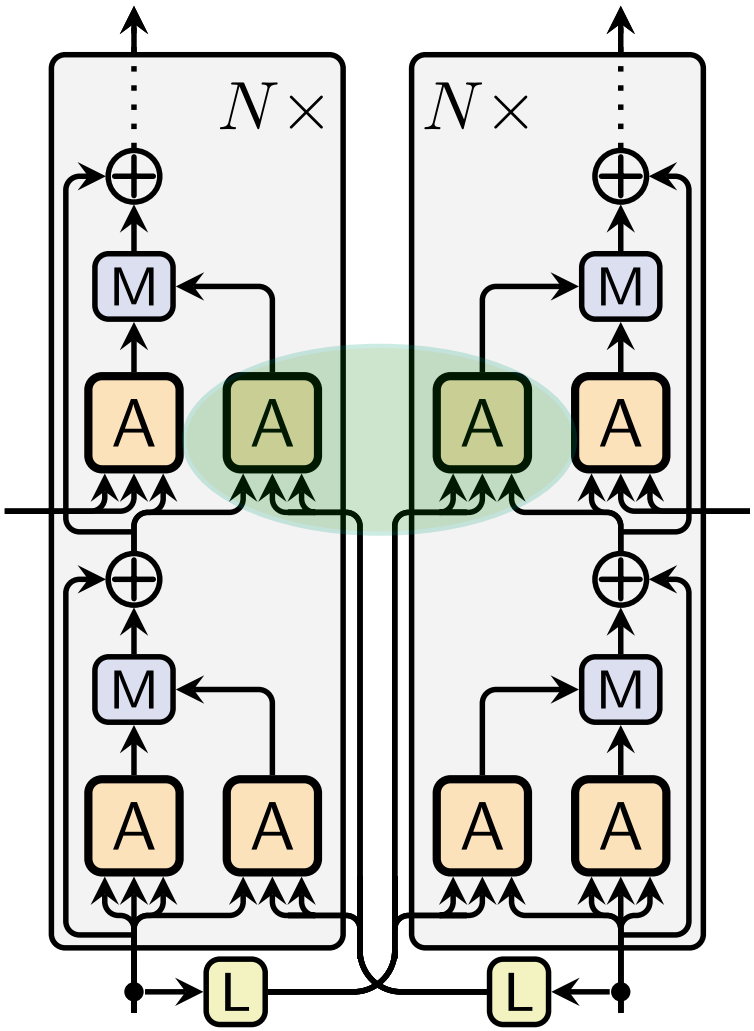
A: Attention, **M**: Merge,
L: LayerNorm.

.

Four additional **dual-attention** layers at each decoder block:

- Lower: dual-attention at self
- Upper: dual-attention at source

First variant: Cross dual-decoder Transformer



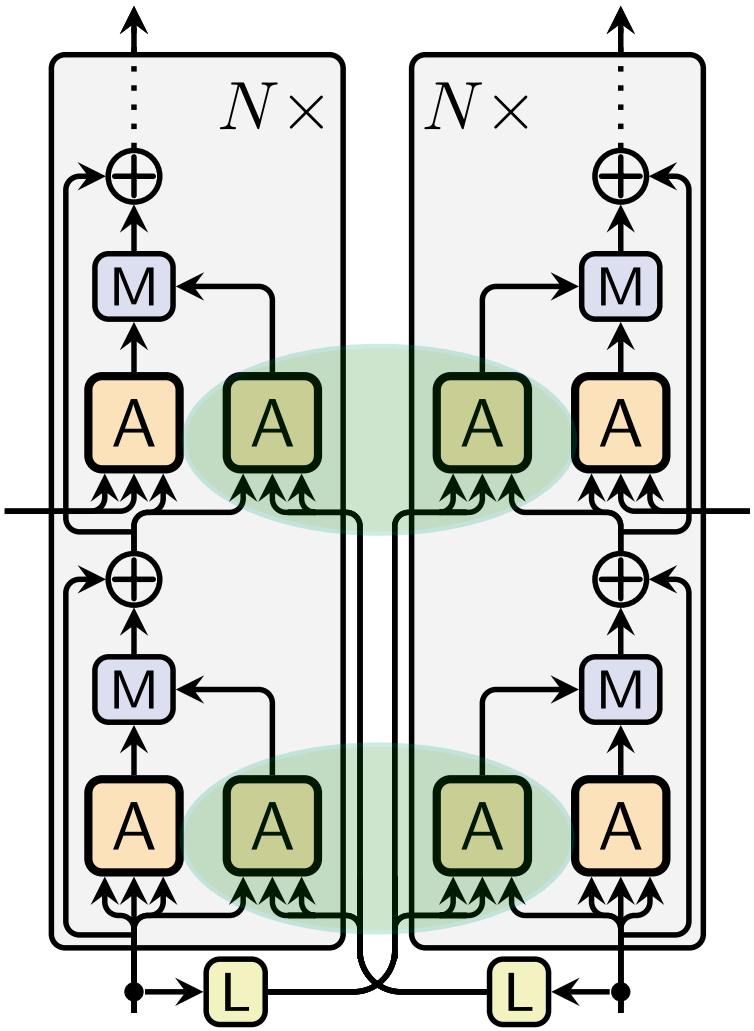
A: Attention, **M**: Merge,
L: LayerNorm.

Four additional **dual-attention** layers at each decoder block:

- Lower: dual-attention at self
- Upper: dual-attention at source

Each dual-attention merges information from the input of the other decoder.

First variant: Cross dual-decoder Transformer



A: Attention, **M**: Merge,
L: LayerNorm.

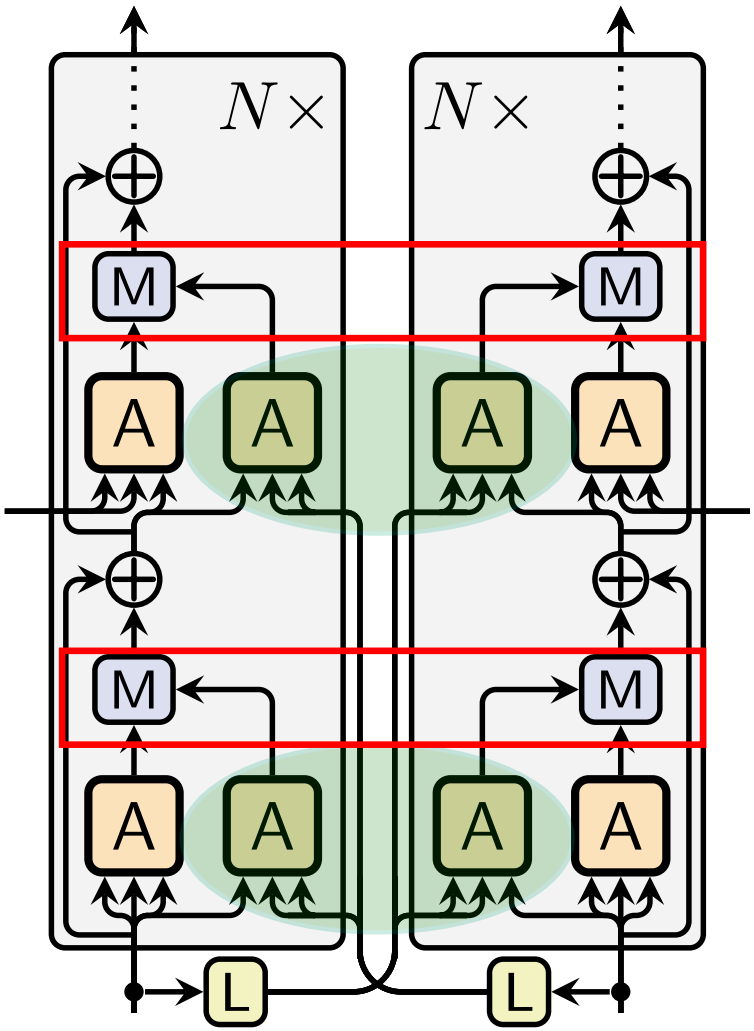
.

Four additional **dual-attention** layers at each decoder block:

- Lower: dual-attention at self
- Upper: dual-attention at source

Each dual-attention merges information from the input of the other decoder.

First variant: Cross dual-decoder Transformer



A: Attention, **M**: Merge,
L: LayerNorm.

.

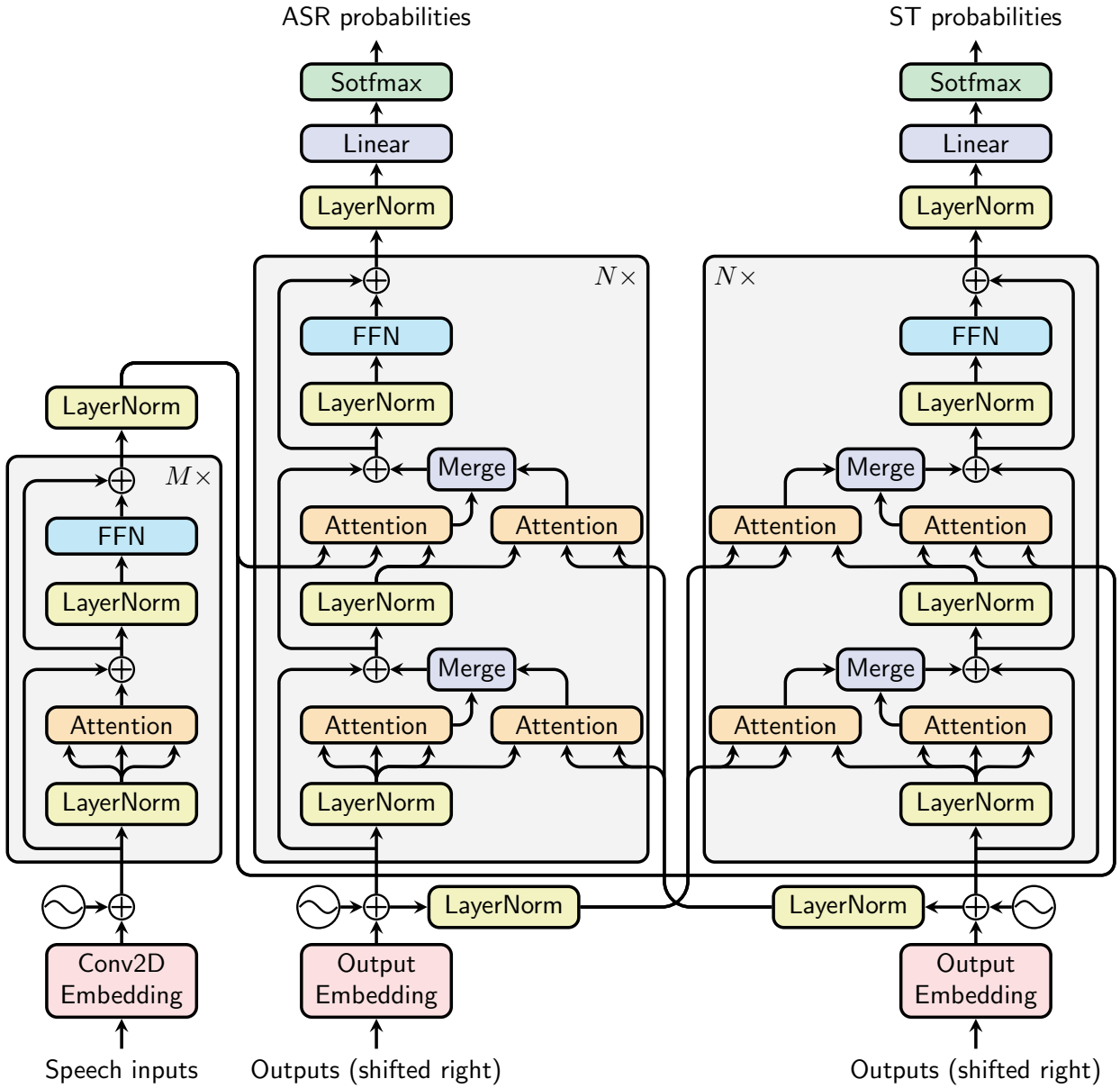
Four additional **dual-attention** layers at each decoder block:

- Lower: dual-attention at self
- Upper: dual-attention at source

Each dual-attention merges information from the input of the other decoder.

First variant: Cross dual-decoder Transformer

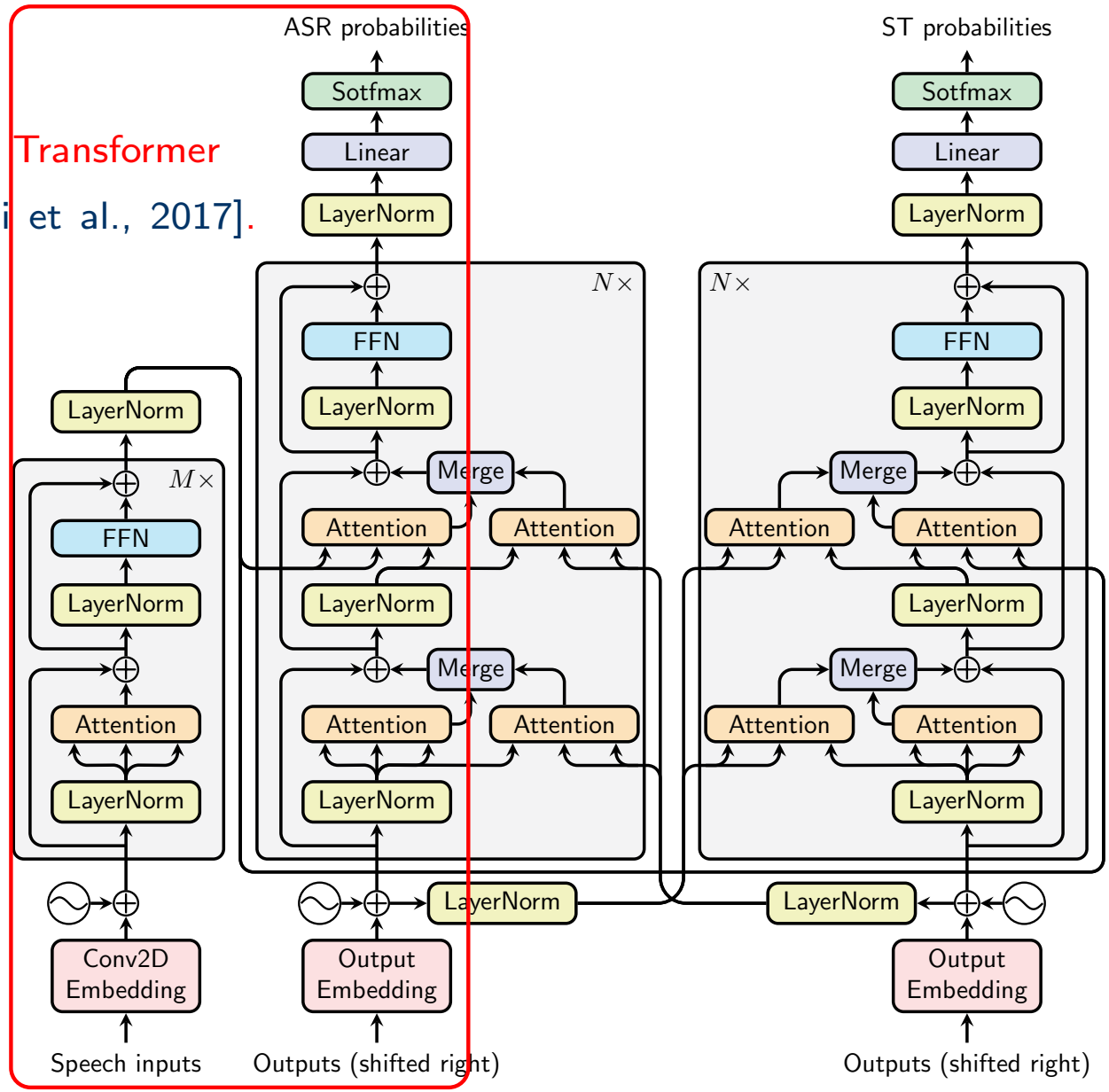
Detailed architecture



First variant: Cross dual-decoder Transformer

Detailed architecture

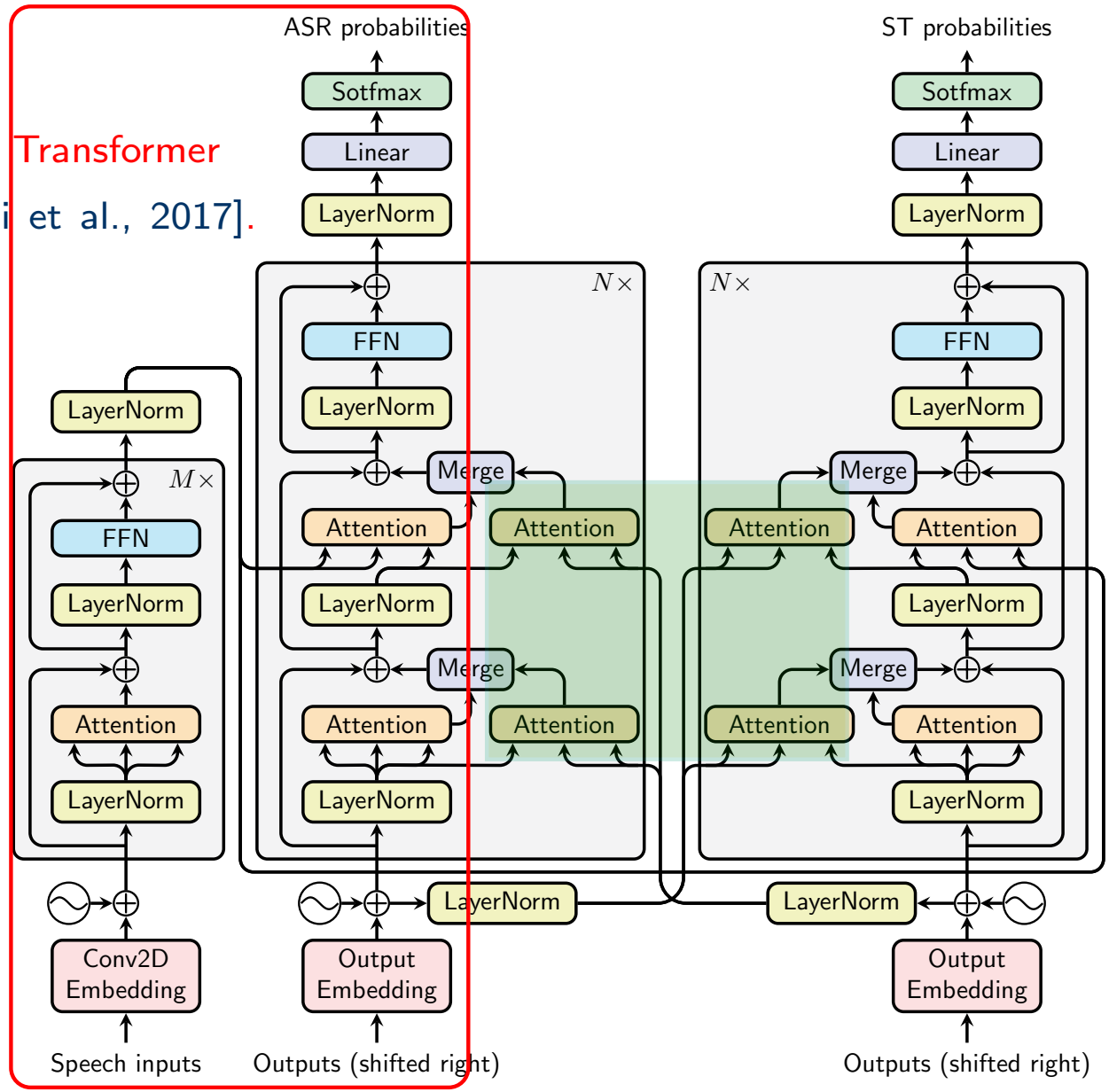
Original Transformer [Vaswani et al., 2017].



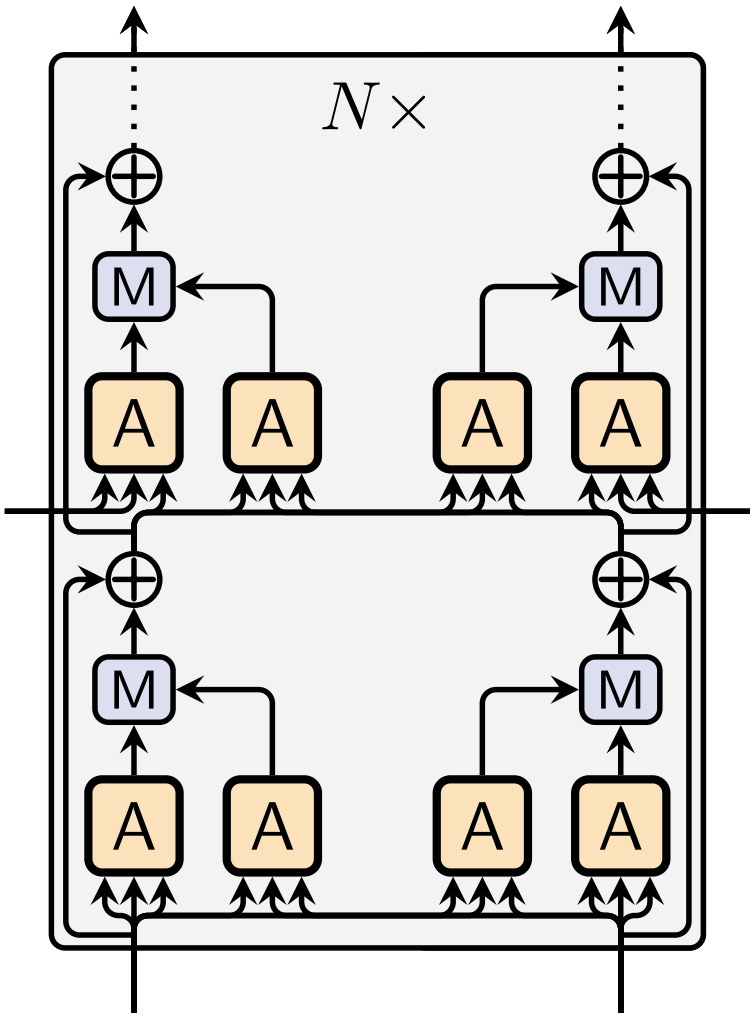
First variant: Cross dual-decoder Transformer

Detailed architecture

Original Transformer [Vaswani et al., 2017].



Second variant: Parallel dual-decoder Transformer

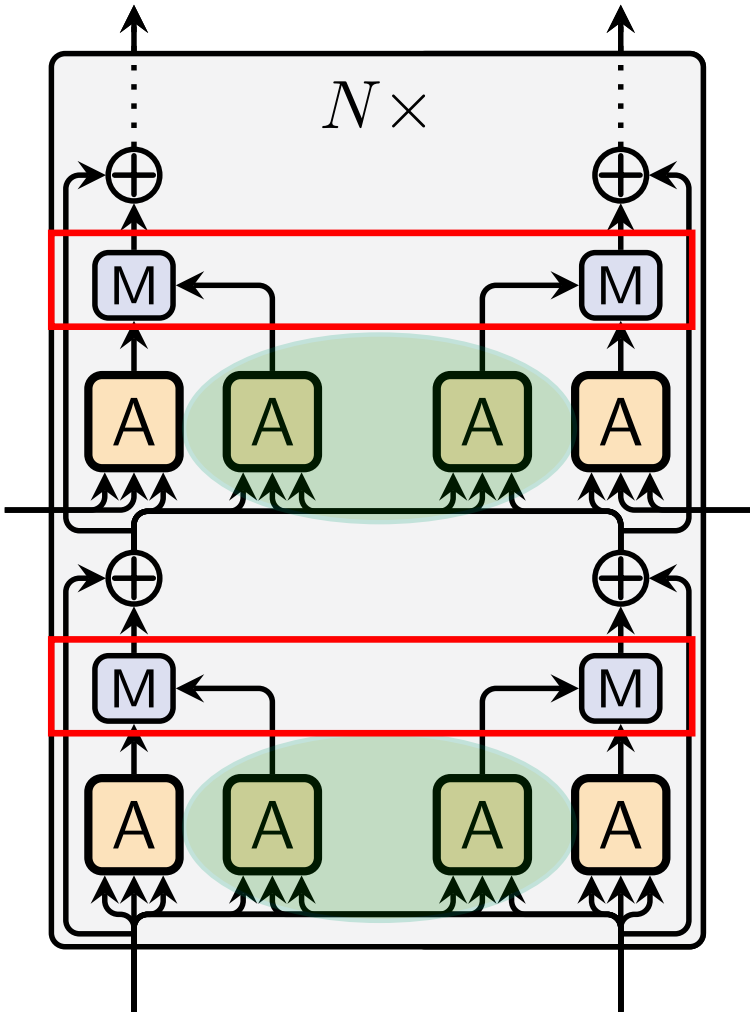


A: Attention, **M**: Merge.

Similar to cross dual-decoder Transformer but with higher level of dependency.

Each dual-attention merges information from the same level of abstraction from the other decoder.

Second variant: Parallel dual-decoder Transformer



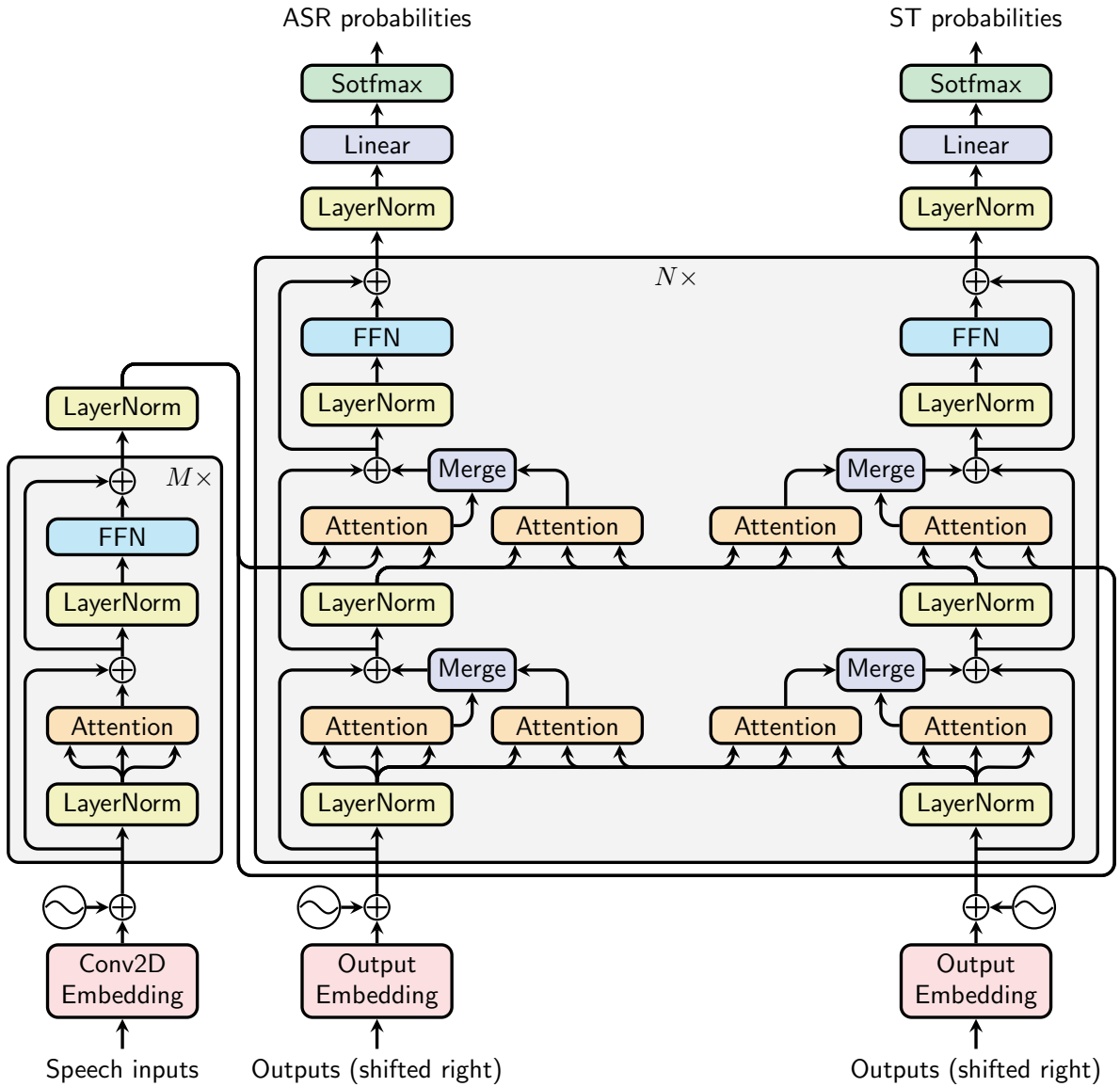
A: Attention, **M**: Merge.

Similar to cross dual-decoder Transformer but with higher level of dependency.

Each dual-attention merges information from the same level of abstraction from the other decoder.

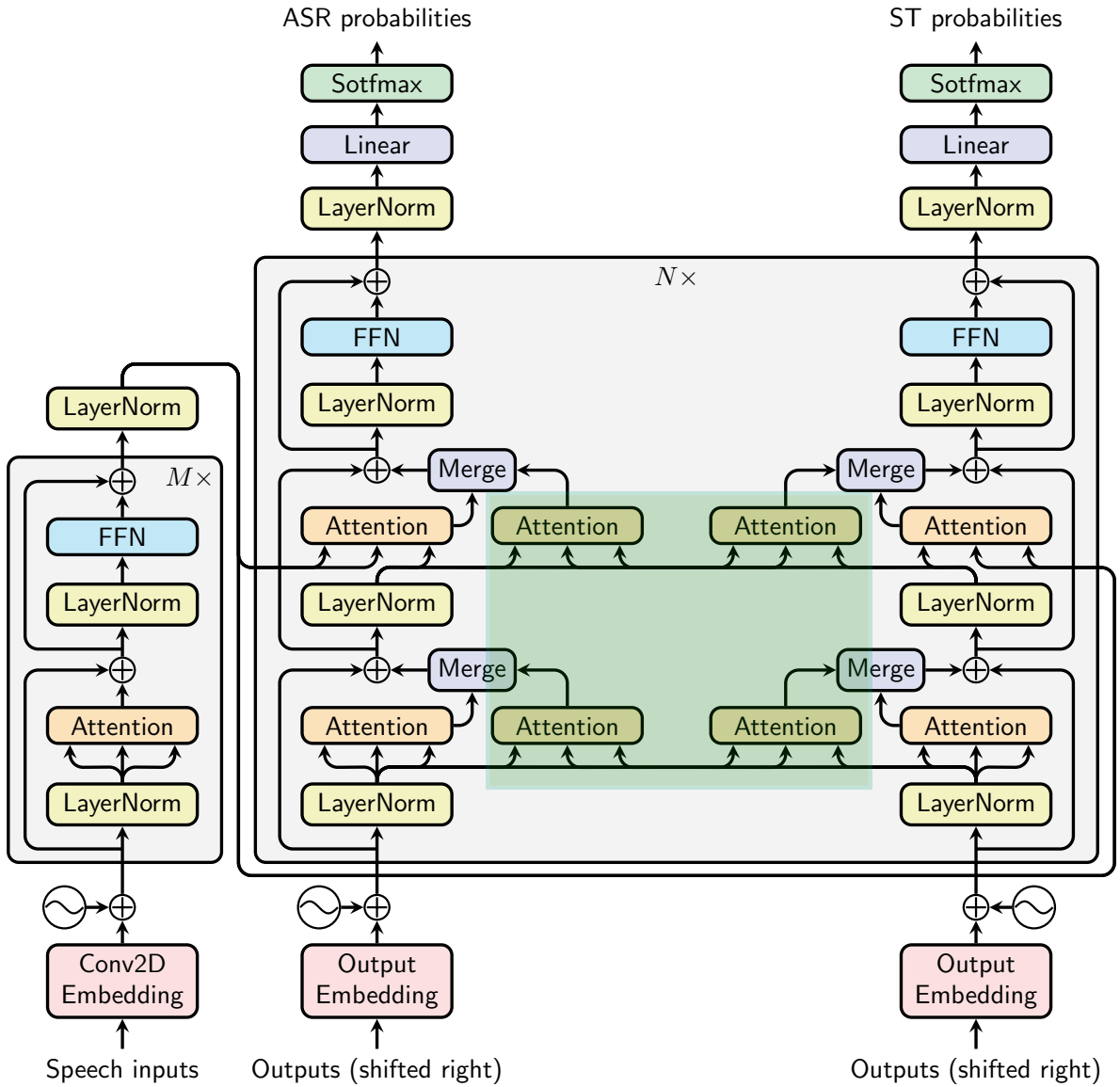
Second variant: Parallel dual-decoder Transformer

Detaild architecture



Second variant: Parallel dual-decoder Transformer

Detaild architecture



Beam-search decoding procedure

Reminder

- Autoregressive generation in single-decoder:

$$p(\mathbf{y} \mid \mathbf{x}) = \prod_{t=0}^{T} p(y_t \mid \mathbf{y}_{<t}, \mathbf{x})$$

Beam-search decoding: At decoding step $t+1$, keep the top B candidates based on their scores $\log p(y_t \mid \mathbf{y}_{<t}, \mathbf{x})$.

Beam-search decoding procedure

Reminder

- Autoregressive generation in single-decoder:

$$p(\mathbf{y} \mid \mathbf{x}) = \prod_{t=0}^{T} p(y_t \mid \mathbf{y}_{<t}, \mathbf{x})$$

Beam-search decoding: At decoding step $t+1$, keep the top B candidates based on their scores $\log p(y_t \mid \mathbf{y}_{<t}, \mathbf{x})$.

- Two independent decoders:

$$p(\mathbf{y}, \mathbf{z} \mid \mathbf{x}) = \prod_{t=0}^{T} p(y_t \mid \mathbf{y}_{<t}, \mathbf{x}) p(z_t \mid \mathbf{z}_{<t}, \mathbf{x}).$$

Beam search is simple: Keep two separate beams as the outputs are independent.

Beam-search decoding procedure

Reminder

- Autoregressive generation in single-decoder:

$$p(\mathbf{y} \mid \mathbf{x}) = \prod_{t=0}^{T} p(y_t \mid \mathbf{y}_{<t}, \mathbf{x})$$

Beam-search decoding: At decoding step $t+1$, keep the top B candidates based on their scores $\log p(y_t \mid \mathbf{y}_{<t}, \mathbf{x})$.

- Two independent decoders:

$$p(\mathbf{y}, \mathbf{z} \mid \mathbf{x}) = \prod_{t=0}^{T} p(y_t \mid \mathbf{y}_{<t}, \mathbf{x}) p(z_t \mid \mathbf{z}_{<t}, \mathbf{x}).$$

Beam search is simple: Keep two separate beams as the outputs are independent.

Beam-search dual-decoding procedure

Autoregressive generation in Dual-decoder Transformer:

$$p(\mathbf{y}, \mathbf{z} \mid \mathbf{x}) = \prod_{t=0}^{T} p(y_t \mid \mathbf{y}_{<t}, \mathbf{z}_{<t}, \mathbf{x}) p(z_t \mid \mathbf{y}_{<t}, \mathbf{z}_{<t}, \mathbf{x})$$

Beam-search dual-decoding procedure

Autoregressive generation in Dual-decoder Transformer:

$$p(\mathbf{y}, \mathbf{z} \mid \mathbf{x}) = \prod_{t=0}^{T} p(y_t \mid \mathbf{y}_{<t}, \mathbf{z}_{<t}, \mathbf{x}) p(z_t \mid \mathbf{y}_{<t}, \mathbf{z}_{<t}, \mathbf{x})$$

Single joint beam: At decoding step $t+1$, take top B predictions $(\hat{y}_t, \hat{z}_t)$ based on sum of scores

$$\log p(y_t \mid \mathbf{y}_{<t}, \mathbf{z}_{<t}, \mathbf{x}) + \log p(z_t \mid \mathbf{y}_{<t}, \mathbf{z}_{<t}, \mathbf{x}).$$

Dual-decoder Transformer: Finer-grained configurations

Question: Can the two tasks (ASR and ST) help each other?

Dual-decoder Transformer: Finer-grained configurations

Question: Can the two tasks (ASR and ST) help each other?

- **Asymmetric dual-decoder**: Either ASR attends ST or the vice-versa, but not both

Dual-decoder Transformer: Finer-grained configurations

Question: Can the two tasks (ASR and ST) help each other?

- **Asymmetric dual-decoder**: Either ASR attends ST or the vice-versa, but not both

Question: Which one should lead the joint task?

Dual-decoder Transformer: Finer-grained configurations

Question: Can the two tasks (ASR and ST) help each other?

- **Asymmetric dual-decoder**: Either ASR attends ST or the vice-versa, but not both

Question: Which one should lead the joint task?

- **Wait-k**: ST waits for k ASR steps (i.e. ASR is ahead of ST), or vice-versa.

Evaluation protocol

Dataset: MuST-C

- One-to-many ST dataset built from TED Talks (prepared speech)
  - Source speech: English
  - Target text: eight different European languages, including Dutch (Nl), French (Fr), German (De), Italian (It), Portuguese (Pt), Romanian (Ro), Russian (Ru), and Spanish (Es)
  - Sizes range from 385 hours to 504 hours.
- Training data: (Source speech, transcript text, translation text)

Evaluation protocol

Dataset: MuST-C

- One-to-many ST dataset built from TED Talks (prepared speech)
  - Source speech: English
  - Target text: eight different European languages, including Dutch (Nl), French (Fr), German (De), Italian (It), Portuguese (Pt), Romanian (Ro), Russian (Ru), and Spanish (Es)
  - Sizes range from 385 hours to 504 hours.
- Training data: (Source speech, transcript text, translation text)

Evaluation metrics

- ASR performance: **Word-error-rate (WER)**
  $\rightarrow$ the smaller the better

Evaluation protocol

Dataset: MuST-C

- One-to-many ST dataset built from TED Talks (prepared speech)
  - Source speech: English
  - Target text: eight different European languages, including Dutch (Nl), French (Fr), German (De), Italian (It), Portuguese (Pt), Romanian (Ro), Russian (Ru), and Spanish (Es)
  - Sizes range from 385 hours to 504 hours.
- Training data: (Source speech, transcript text, translation text)

Evaluation metrics

- ASR performance: **Word-error-rate (WER)**
  $\rightarrow$ the smaller the better
- ST performance: **BiLingual Evaluation Understudy (BLEU)**
  $\rightarrow$ the higher the better.

Experimental results

Model name	Design	Params	ST ↑	ASR ↓

Experimental results

Model name	Design	Params	ST ↑	ASR ↓
Independent decoders	`base`	45M	21.46	12.6
	`more params`	51M	22.21	12.9

Experimental results

Model name	Design	Params	ST ↑	ASR ↓
Independent decoders	`base`	45M	21.46	12.6
	`more params`	51M	22.21	12.9
Interactive decoding	[Liu et al., 2020]	48M	20.39	12.8

Experimental results

Model name	**Design**	**Params**	**ST** ↑	**ASR** ↓
Independent decoders	`base`	45M	21.46	12.6
	`more params`	51M	22.21	12.9
Interactive decoding	[Liu et al., 2020]	48M	20.39	12.8
Cross dual-decoder		48M	21.71	**12.2**

Experimental results

Model name	**Design**	**Params**	**ST** ↑	**ASR** ↓
Independent decoders	`base`	45M	21.46	12.6
	`more params`	51M	22.21	12.9
Interactive decoding	[Liu et al., 2020]	48M	20.39	12.8
Cross dual-decoder		48M	21.71	**12.2**
Parallel dual-decoder	`asymmetric`	50M	21.93	13.0
	`symmetric`	54M	**22.70**	12.7

Experimental results

Model name	**Design**	**Params**	**ST** ↑	**ASR** ↓
Independent decoders	`base`	45M	21.46	12.6
	`more params`	51M	22.21	12.9
Interactive decoding	[Liu et al., 2020]	48M	20.39	12.8
Cross dual-decoder		48M	21.71	**12.2**
Parallel dual-decoder	`asymmetric`	50M	21.93	13.0
	`symmetric`	54M	**22.70**	12.7
Parallel dual-decoder	`non-wait-`k	48M	**22.54**	**12.7**
	`ST-wait-ASR`	48M	**22.78**	**12.6**
	`ASR-wait-ST`	48M	21.85	13.6

Experimental results

Model name	Design	Params	ST ↑	ASR ↓
Independent decoders	`base`	45M	21.46	12.6
	`more params`	51M	22.21	12.9
Interactive decoding	[Liu et al., 2020]	48M	20.39	12.8
Cross dual-decoder		48M	21.71	**12.2**
Parallel dual-decoder	`asymmetric`	50M	21.93	13.0
	`symmetric`	54M	**22.70**	12.7
Parallel dual-decoder	`non-wait-`k	48M	**22.54**	**12.7**
	`ST-wait-ASR`	48M	**22.78**	**12.6**
	`ASR-wait-ST`	48M	21.85	13.6

Independent-`base` $<$ **Cross dual-decoder** $<$ **Independent**-`more params`

Experimental results

Model name	Design	Params	ST ↑	ASR ↓
Independent decoders	`base`	45M	21.46	12.6
	`more params`	51M	22.21	12.9
Interactive decoding	[Liu et al., 2020]	48M	20.39	12.8
Cross dual-decoder		48M	21.71	**12.2**
Parallel dual-decoder	`asymmetric`	50M	21.93	13.0
	`symmetric`	54M	**22.70**	12.7
Parallel dual-decoder	`non-wait-`k	48M	**22.54**	**12.7**
	`ST-wait-ASR`	48M	**22.78**	**12.6**
	`ASR-wait-ST`	48M	21.85	13.6

Parallel dual-decoder $>$ **Independent**-`more params`

Experimental results

Model name	Design	Params	ST ↑	ASR ↓
Independent decoders	`base`	45M	21.46	12.6
	`more params`	51M	22.21	12.9
Interactive decoding	[Liu et al., 2020]	48M	20.39	12.8
Cross dual-decoder		48M	21.71	**12.2**
Parallel dual-decoder	`asymmetric`	50M	21.93	13.0
	`symmetric`	54M	**22.70**	12.7
Parallel dual-decoder	`non-wait-`k	48M	**22.54**	**12.7**
	`ST-wait-ASR`	48M	**22.78**	**12.6**
	`ASR-wait-ST`	48M	21.85	13.6

Asymmetric $<$ Symmetric

Experimental results

Model name	Design	Params	ST ↑	ASR ↓
Independent decoders	base	45M	21.46	12.6
	more params	51M	22.21	12.9
Interactive decoding	[Liu et al., 2020]	48M	20.39	12.8
Cross dual-decoder		48M	21.71	**12.2**
Parallel dual-decoder	asymmetric	50M	21.93	13.0
	symmetric	54M	**22.70**	12.7
Parallel dual-decoder	non-wait-k	48M	**22.54**	**12.7**
	ST-wait-ASR	48M	**22.78**	**12.6**
	ASR-wait-ST	48M	21.85	13.6

ST-wait-ASR (ASR ahead) > ASR-wait-ST (ST ahead)

Several examples of outputs of the models

ASR outputs ≡ target	this tool that i'm using here is a little experiment
Independent	Voici une petite expérience.
Dual-decoder	Cette outil que j'utilise ici est une petite expérience.
Ground-truth	L'outil que j'utilise ici est une petite expérience.
ASR outputs ≡ target	now what does that have to do with the placebo effect
Independent	Maintenant, qu'est-ce que ça a à faire avec l'effet de place ?
Dual-decoder	Maintenant, qu'est-ce que ça a à voir avec l'effet placebo ?
Ground-truth	Maintenant, qu'est-ce que ça a à voir avec l'effet placebo ?
ASR outputs ≡ target	you have that much uncertainty
Independent	Vous avez beaucoup d'incertitude.
Dual-decoder	Vous avez tant d'incertitude.
Ground-truth	Vous avez tant d'incertitude.
ASR outputs ≡ target	and so that's a big challenge
Independent	C'est un grand défi.
Dual-decoder	Et donc c'est un grand défi.
Ground-truth	Et donc c'est un grand défi.
ASR outputs ≡ target	because frankly every project has its own marshmallow isn't it
Independent	Parce que, franchement, tous les projets ont son propre mélodie de Mars, n'est-ce pas ?
Dual-decoder	Parce que, franchement, chaque projet a son propre marshmallow, n'est-ce pas ?
Ground-truth	Parce que, franchement, chaque projet possède son propre marshmallow, n'est-ce pas ?
ASR outputs ≡ target	and it's because of the experiences i've had with them not in spite of the experiences i've had with them
Independent	Et c'est à cause des expériences que j'ai eu avec eux, pas malgré les expériences que j'ai eu avec eux.
Dual-decoder	Et c'est grâce aux expériences que j'ai eues avec elles, pas malgré les expériences que j'ai eues avec elles.
Ground-truth	Et c'est grâce aux expériences que j'ai eues avec elles, pas malgré les expérience que j'ai eues avec elles.

Conclusion

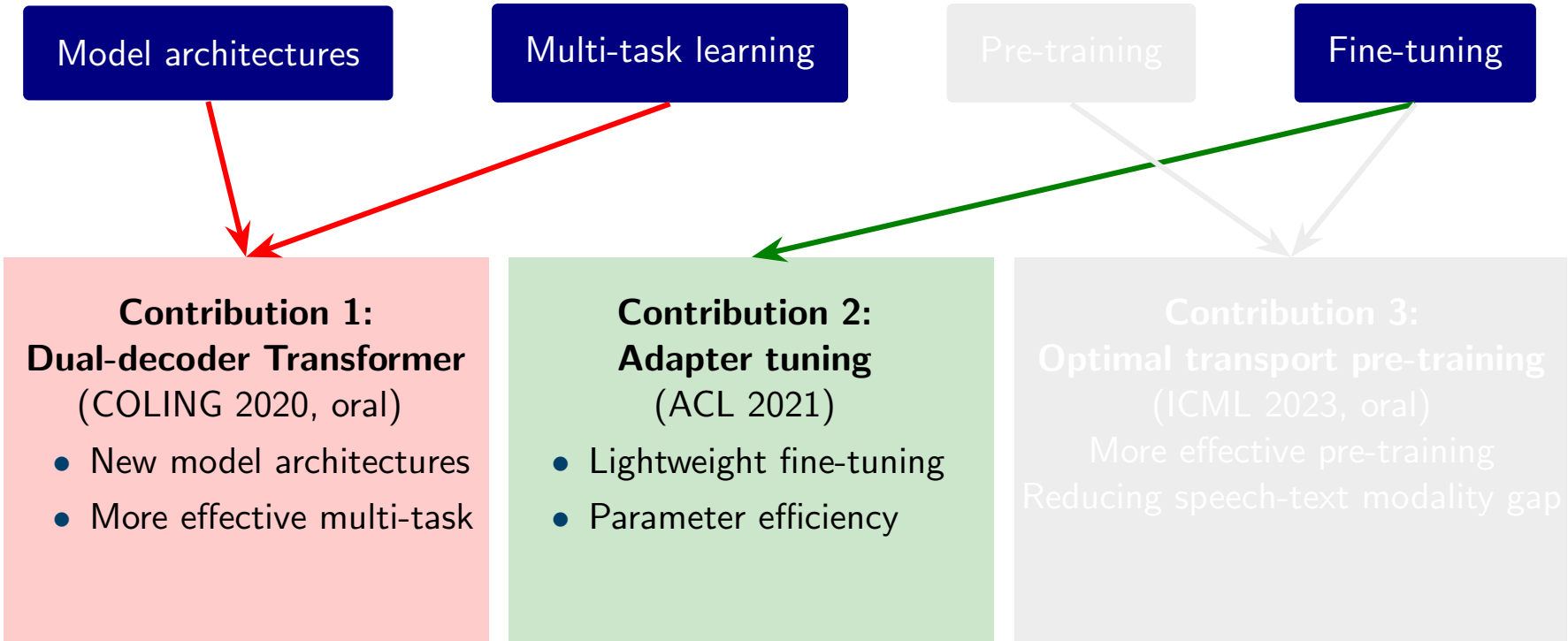
Conclusion

- **Novel model architecture** that jointly transcribes and translates an input speech.
  - Different variants: cross vs. parallel, asymmetric vs. symmetric, wait-k, etc.

Conclusion

- **Novel model architecture** that jointly transcribes and translates an input speech.
  - Different variants: cross vs. parallel, asymmetric vs. symmetric, wait-k, etc.
- Show that **ASR and ST are complementary** and can help each other

Contribution 2: Adapter tuning



Hang Le, Juan Pino, Changhan Wang, Jiatao Gu, Didier Schwab, Laurent Besacier. Lightweight Adapter Tuning for Multilingual Speech Translation. In *Annual Meeting of the Association for Computational Linguistics* (**ACL 2021**).

Motivation

Question often raised in designing multilingual translation systems:

Versatility or **Specialization**?

- Versatility: single multilingual model
- Specialization: system consisting of multiple bilingual models.

Motivation

Question often raised in designing multilingual translation systems:

Versatility or **Specialization**?

- Versatility: single multilingual model
  - Lower training and deployment complexities (both in terms of time and space requirements)
  - Needs re-training for updates or addition of new languages.
- Specialization: system consisting of multiple bilingual models.

Motivation

Question often raised in designing multilingual translation systems:

Versatility or **Specialization**?

- Versatility: single multilingual model
  - Lower training and deployment complexities (both in terms of time and space requirements)
  - Needs re-training for updates or addition of new languages.
- Specialization: system consisting of multiple bilingual models.
  - High modularity: Easy to update a component or to add a new language
  - Typically high performance, especially on high-resource languages
  - High training and deployment complexities.

Motivation

Question often raised in designing multilingual translation systems:

Versatility or **Specialization**?

- Versatility: single multilingual model
  - Lower training and deployment complexities (both in terms of time and space requirements)
  - Needs re-training for updates or addition of new languages.
- Specialization: system consisting of multiple bilingual models.
  - High modularity: Easy to update a component or to add a new language
  - Typically high performance, especially on high-resource languages
  - High training and deployment complexities.

Motivation

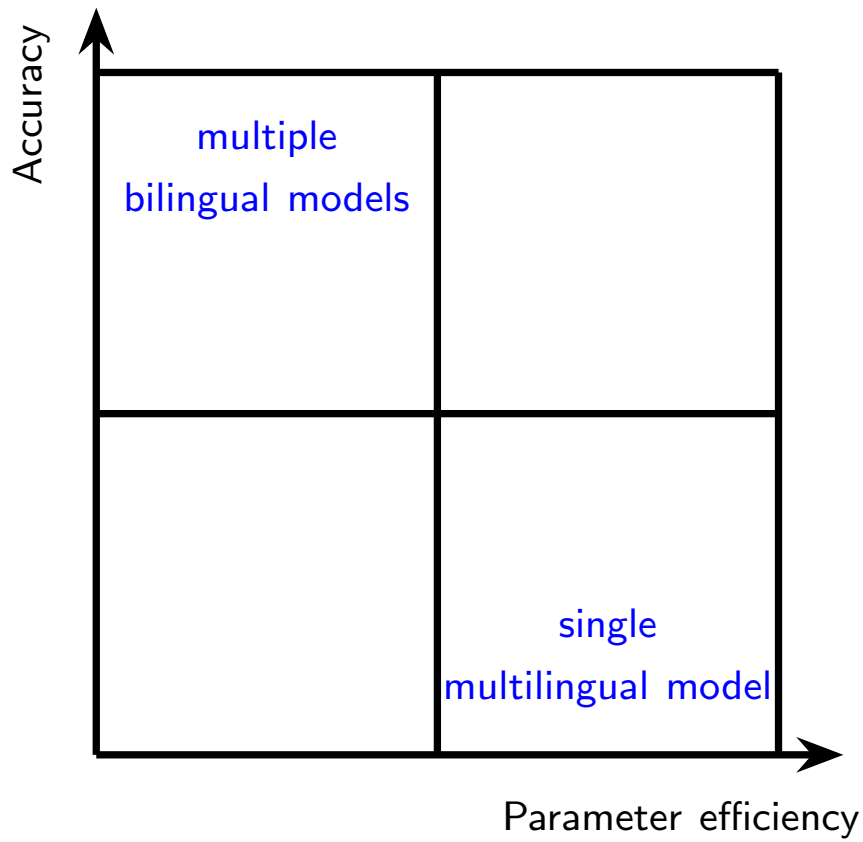
Versatility or **Specialization**?

Adapter tuning has best of both worlds.

Motivation

Versatility or **Specialization**?

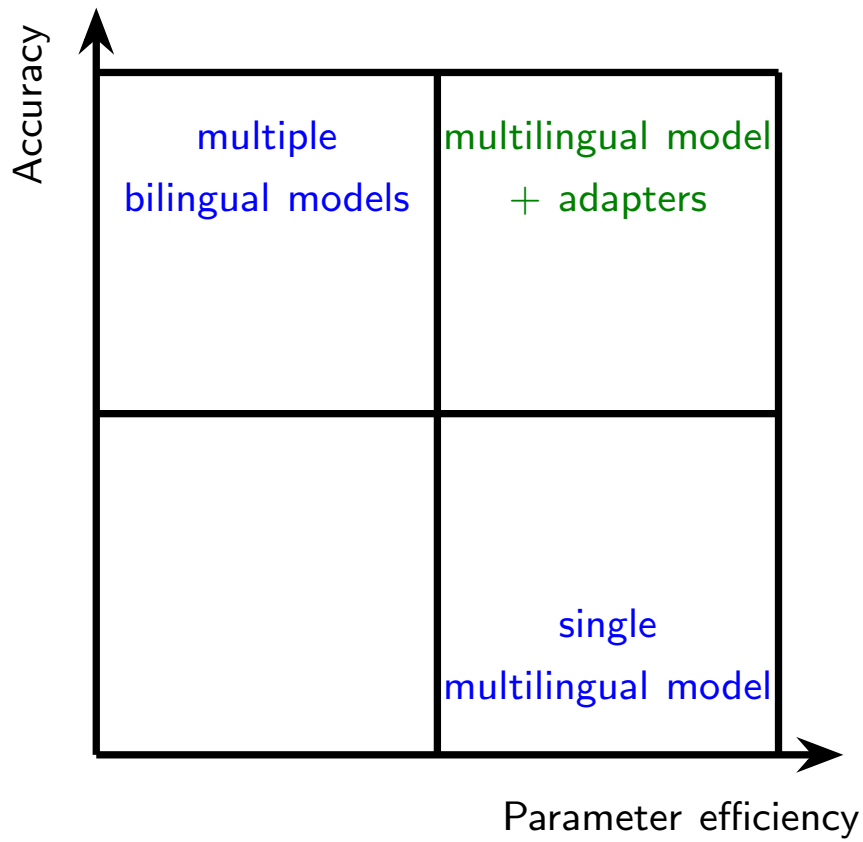
Adapter tuning has best of both worlds.



Motivation

Versatility or **Specialization**?

Adapter tuning has best of both worlds.



Adapter layers

Definition: An adapter module is a component that is added to a pre-trained model for fine-tuning purpose [Rebuffi et al., 2017].

Adapter layers

Definition: An adapter module is a component that is added to a pre-trained model for fine-tuning purpose [Rebuffi et al., 2017].

Adapter layers

Definition: An adapter module is a component that is added to a pre-trained model for fine-tuning purpose [Rebuffi et al., 2017].

- Adapter modules can be introduced into a Transformer in a serial or parallel fashion.

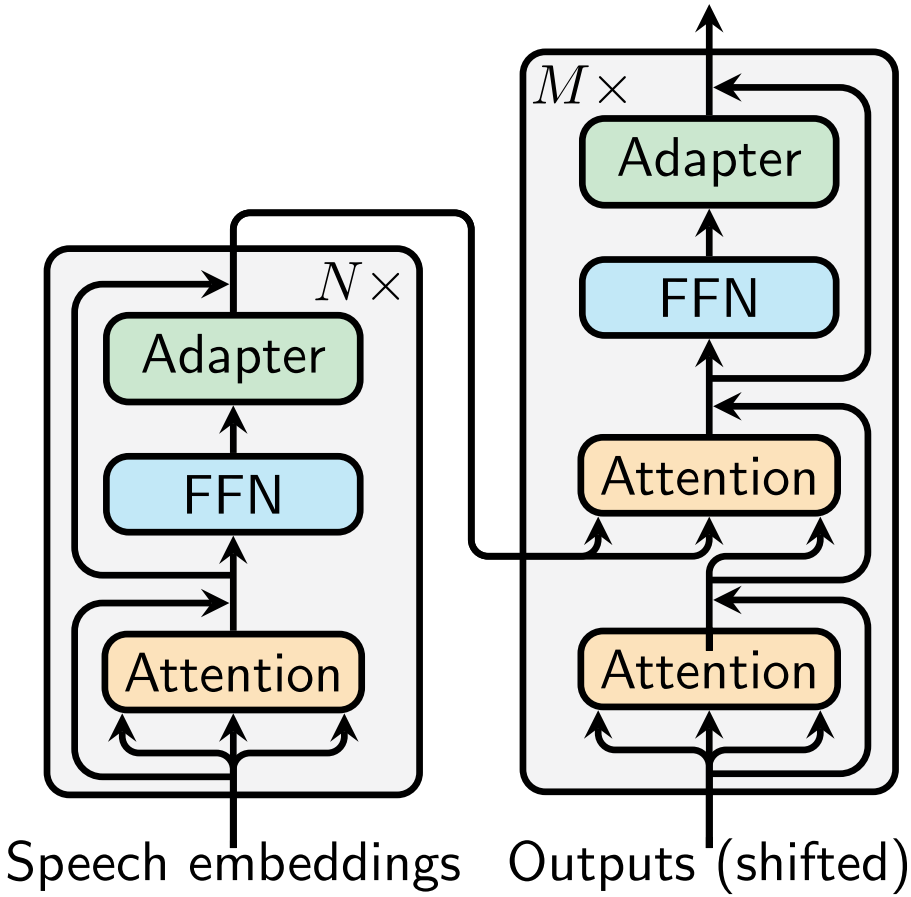
Adapter layers

Definition: An adapter module is a component that is added to a pre-trained model for fine-tuning purpose [Rebuffi et al., 2017].

- Adapter modules can be introduced into a Transformer in a serial or parallel fashion.
- f: a component of the backbone model, g: an adapter layer. Instead of $\mathbf{y} = f(\mathbf{x})$, the new "adapted output" is given by:

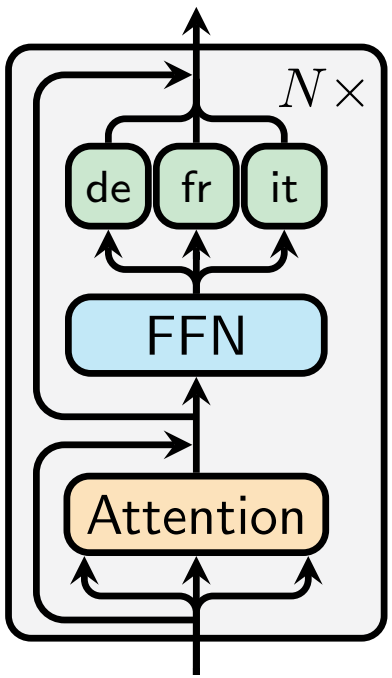
$$\mathbf{y}_{\text{serial}} = g(f(\mathbf{x})), \qquad \mathbf{y}_{\text{parallel}} = f(\mathbf{x}) + g(\mathbf{x})$$

Adapter layers for the Transformer



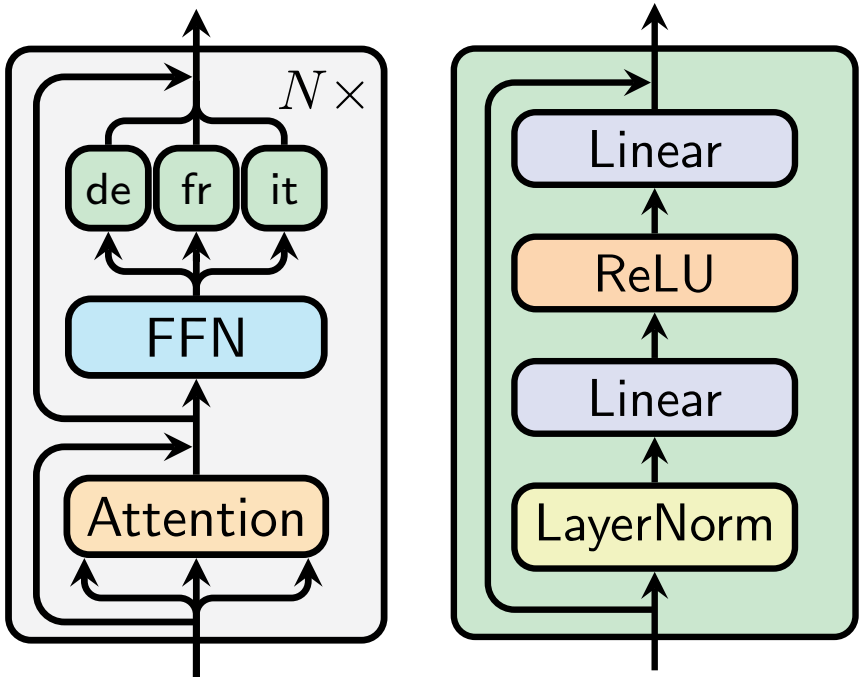
Transformer with adapters at its feedforward network (FFN) sub-layers

Adapters for multilingual speech-to-text translation



We propose **language-specific adapters** for multilingual ST.

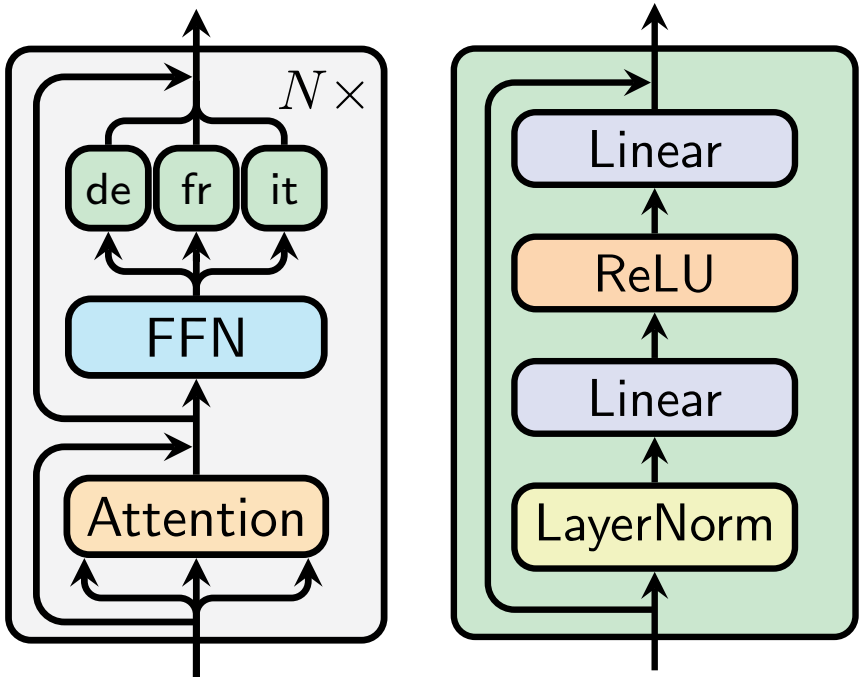
Adapters for multilingual speech-to-text translation



A typical adapter cell

We propose **language-specific adapters** for multilingual ST.

Adapters for multilingual speech-to-text translation



A typical adapter cell

We propose **language-specific adapters** for multilingual ST.

Adapter tuning:

1. Pre-train a backbone model
2. Add adapters for each language pair
3. Finetune adapters on the corresponding bilingual data (backbone is frozen).

Evaluating adapters

We explore two scenarios to evaluate our adapters:

Evaluating adapters

We explore two scenarios to evaluate our adapters:

Refinement

- Start from a fully trained multilingual ST backbone
- Refine on each language pair to close the gap with bilingual models.

Evaluating adapters

We explore two scenarios to evaluate our adapters:

Refinement

- Start from a fully trained multilingual ST backbone
- Refine on each language pair to close the gap with bilingual models.

Transfer learning

- Start from a pre-trained ASR encoder and mBART50 [Tang et al., 2020] decoder
- Connect existing pre-trained components to perform multilingual ST.

Refinement results

On MuST-C dataset

Adapter ENC	Adapter DEC	Finetune ENC	Finetune DEC	# params (M) trainable/total	BLEU
Multiple bilingual models				8×31.1/8×31.1	23.82
Single multilingual model				32.1/32.1	23.76
-	✓	-	-	8×0.4/35.3	23.96
✓	✓	-	-	8×1.2/41.7	24.36
-	-	-	✓	8×14.6/8×32.1	24.75
-	-	✓	✓	8×32.1/8×32.1	**24.90**

Orignal training data: sizes ranging from **385 hours to 504 hours**

Refinement results

On MuST-C dataset

Adapter ENC	Adapter DEC	Finetune ENC	Finetune DEC	# params (M) trainable/total	BLEU
Multiple bilingual models				8×31.1/8×31.1	23.82
Single multilingual model				32.1/32.1	23.76
-	✓	-	-	8×0.4/35.3	23.96
✓	✓	-	-	8×1.2/41.7	24.36
-	-	-	✓	8×14.6/8×32.1	24.75
-	-	✓	✓	8×32.1/8×32.1	**24.90**

Orignal training data: sizes ranging from **385 hours to 504 hours**

Refinement results

On MuST-C dataset

Adapter ENC	Adapter DEC	Finetune ENC	Finetune DEC	# params (M) trainable/total	BLEU
Multiple bilingual models				8×31.1/8×31.1	23.82
Single multilingual model				32.1/32.1	23.76
-	✓	-	-	8×0.4/35.3	23.96
✓	✓	-	-	8×1.2/41.7	24.36
-	-	-	✓	8×14.6/8×32.1	24.75
-	-	✓	✓	8×32.1/8×32.1	**24.90**

Orignal training data: sizes ranging from **385 hours to 504 hours**

Adding adapters to decoder improves performance

Refinement results

On MuST-C dataset

Adapter ENC	Adapter DEC	Finetune ENC	Finetune DEC	# params (M) trainable/total	BLEU
Multiple bilingual models				8×31.1/8×31.1	23.82
Single multilingual model				32.1/32.1	23.76
-	✓	-	-	8×0.4/35.3	23.96
✓	✓	-	-	8×1.2/41.7	24.36
-	-	-	✓	8×14.6/8×32.1	24.75
-	-	✓	✓	8×32.1/8×32.1	**24.90**

Orignal training data: sizes ranging from **385 hours to 504 hours**

Adapters in encoder + decoder > Adapters in decoder only

Refinement results

On MuST-C dataset

Adapter ENC	Adapter DEC	Finetune ENC	Finetune DEC	# params (M) trainable/total	BLEU
Multiple bilingual models				8×31.1/8×31.1	23.82
Single multilingual model				32.1/32.1	23.76
-	✓	-	-	8×0.4/35.3	23.96
✓	✓	-	-	8×1.2/41.7	24.36
-	-	-	✓	8×14.6/8×32.1	24.75
-	-	✓	✓	8×32.1/8×32.1	**24.90**

Orignal training data: sizes ranging from **385 hours to 504 hours**

Finetune enc + dec > Finetune dec > Adapters in encoder + decoder

Refinement results

On imbalanced MuST-C dataset

Refinement results

On imbalanced MuST-C dataset

Adapter ENC	Adapter DEC	Finetune ENC	Finetune DEC	# params (M) trainable/total	BLEU
Single multilingual model				32.1/32.1	21.22
✓	✓	-	-	8×1.2/41.7	**21.92**
-	-	✓	✓	8×32.1/8×32.1	21.87

Imbalanced training data: sizes ranging from **38 hours to 504 hours**

Refinement results

On imbalanced MuST-C dataset

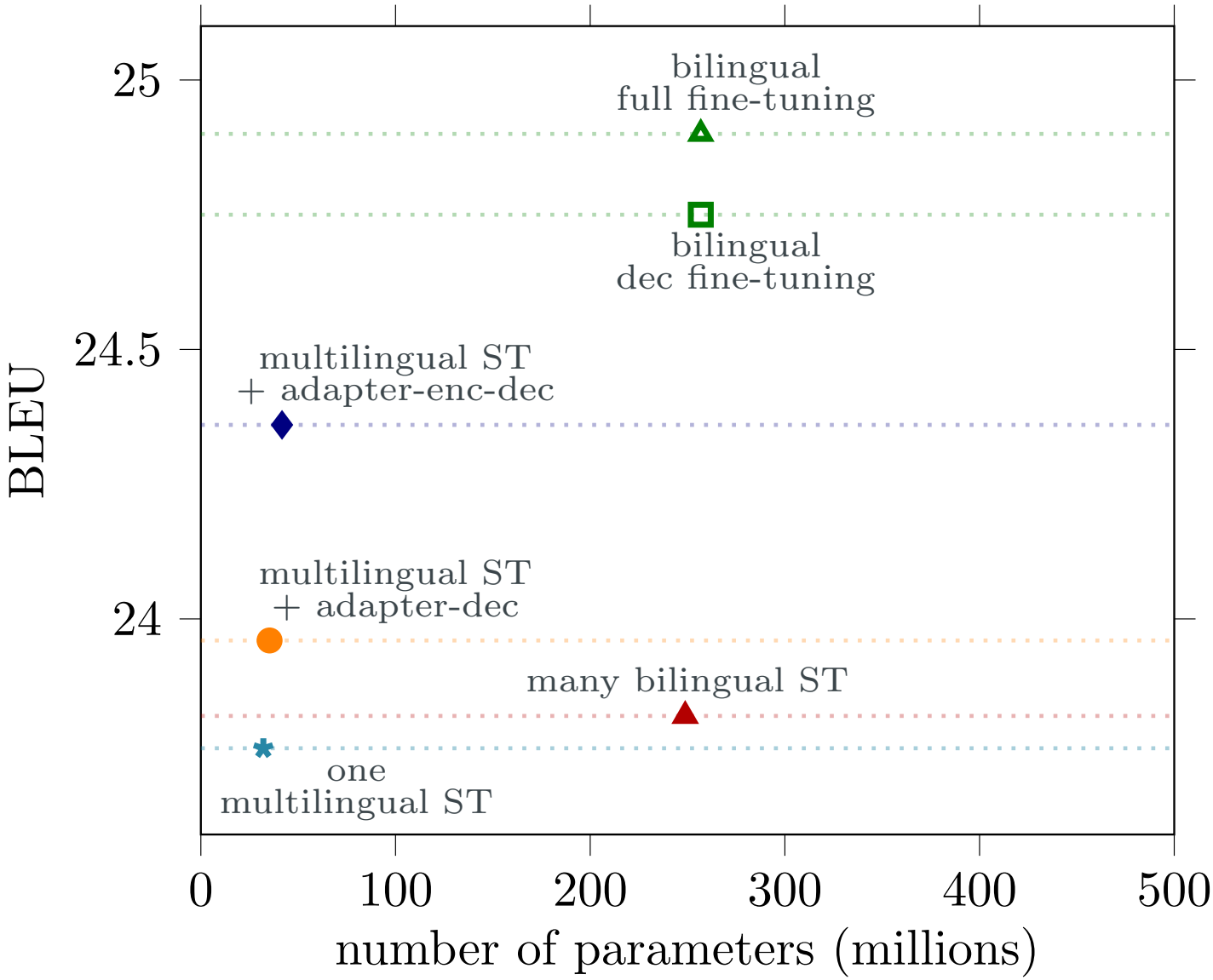
Adapter ENC	Adapter DEC	Finetune ENC	Finetune DEC	# params (M) trainable/total	BLEU
Single multilingual model				32.1/32.1	21.22
✓	✓	-	-	8×1.2/41.7	**21.92**
-	-	✓	✓	8×32.1/8×32.1	21.87

Imbalanced training data: sizes ranging from **38 hours to 504 hours**

Adapters in encoder + decoder > Finetune enc + dec

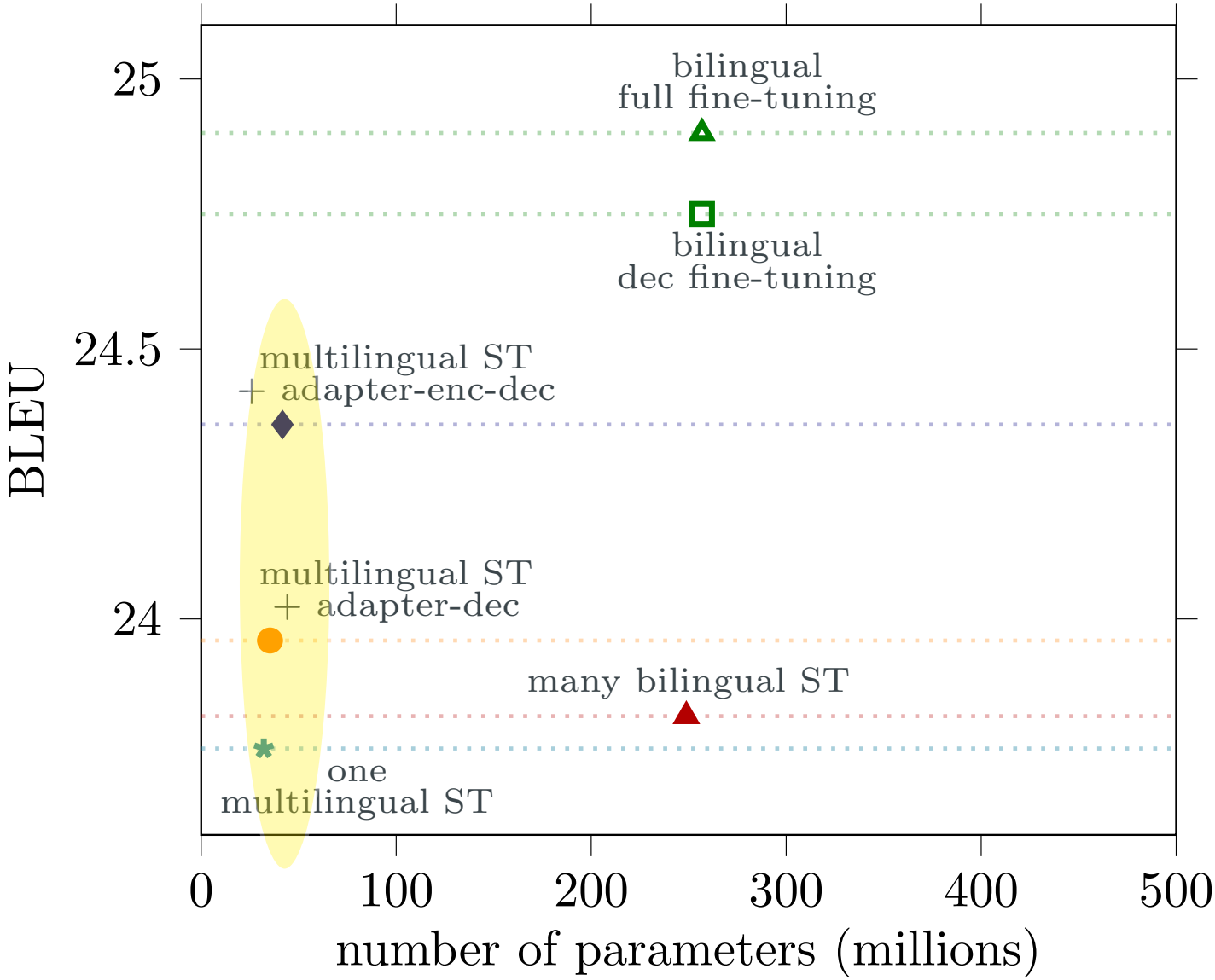
Refinement scenario

Data sizes ranging from 385 hours to 504 hours



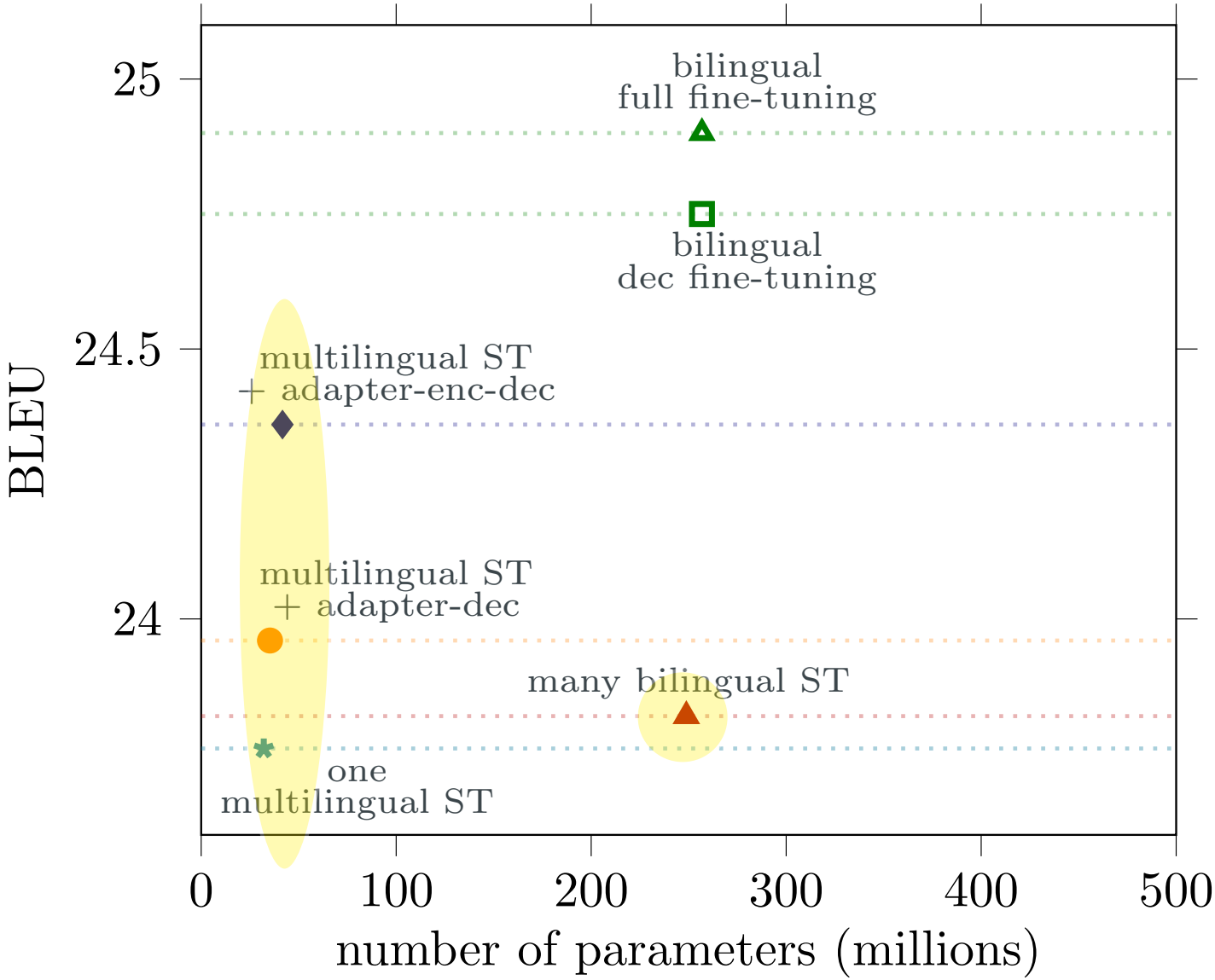
Refinement scenario

Data sizes ranging from 385 hours to 504 hours



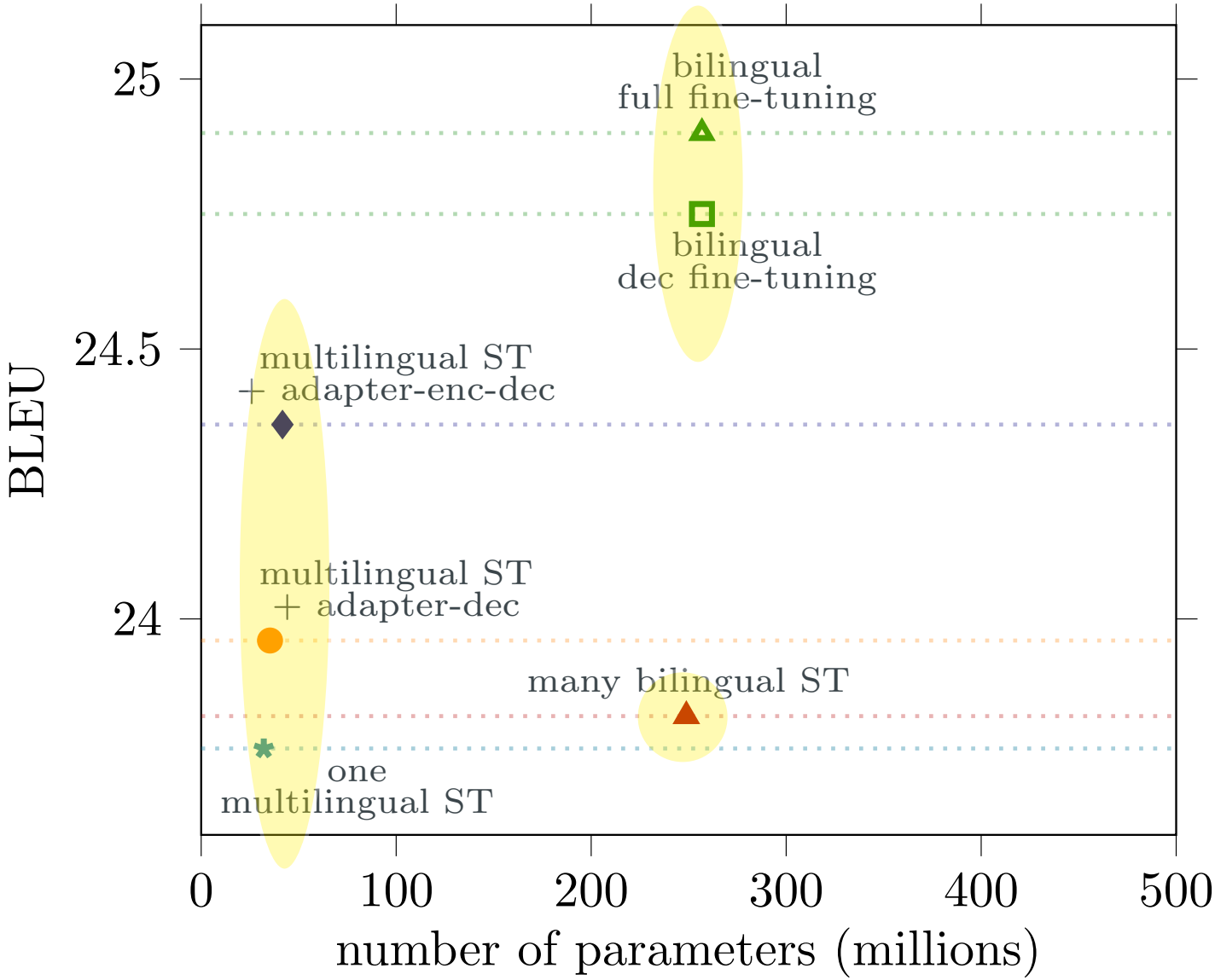
Refinement scenario

Data sizes ranging from 385 hours to 504 hours



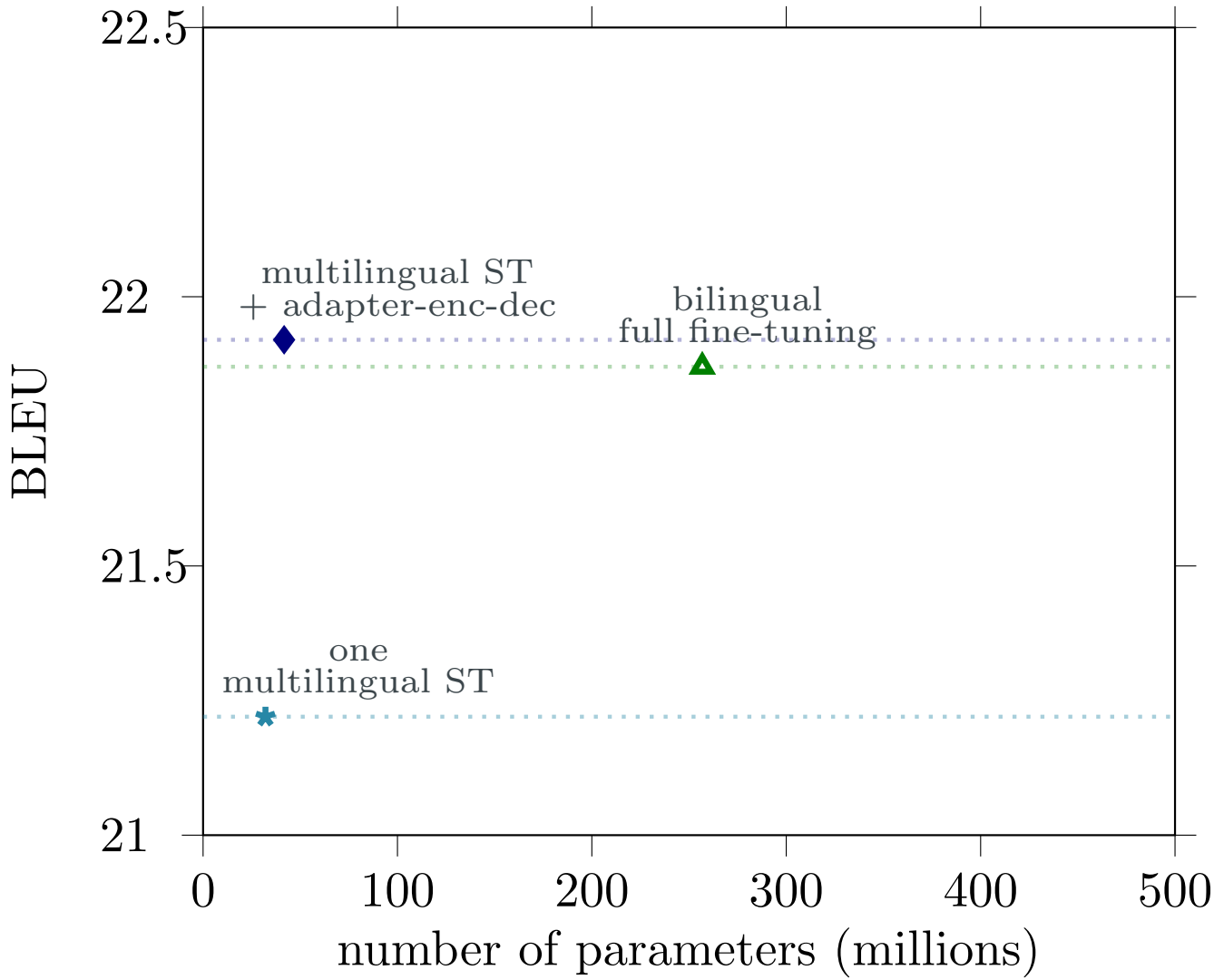
Refinement scenario

Data sizes ranging from 385 hours to 504 hours



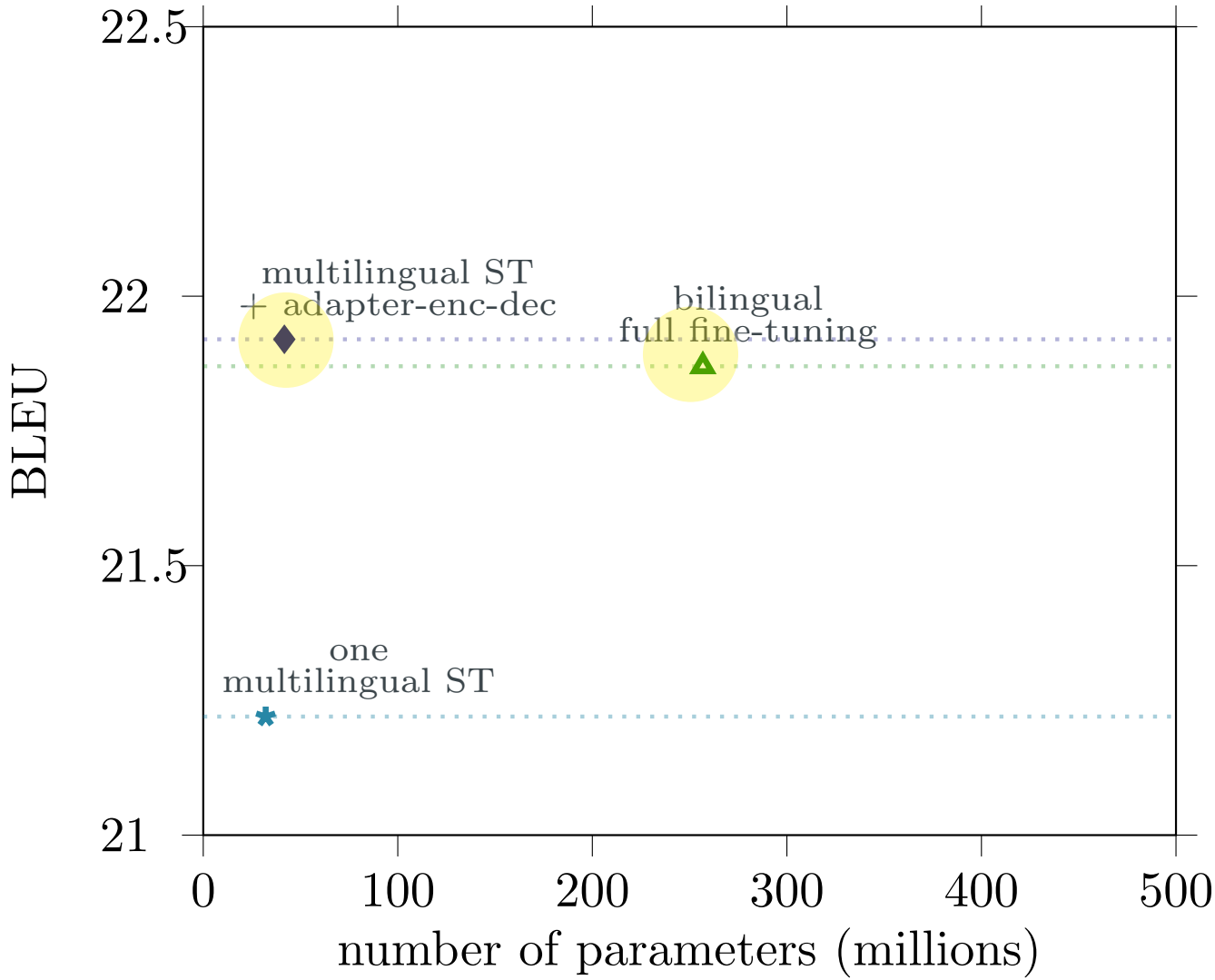
Refinement scenario: Imbalanced dataset

Data sizes ranging from 38 hours to 504 hours



Refinement scenario: Imbalanced dataset

Data sizes ranging from 38 hours to 504 hours



Transfer learning

Adapter ENC	Adapter DEC	Finetune xattn	Params trainable/total	BLEU
-	-	✓	38M / 486M	19.52
-	✓	-	101M / 587M	0.82
-	✓	✓	139M / 587M	23.39
✓	✓	-	152M / 638M	12.90
✓	✓	✓	190M / 638M	24.10

Transfer learning

Adapter ENC	Adapter DEC	Finetune xattn	Params trainable/total	BLEU
-	-	✓	38M / 486M	19.52
-	✓	-	101M / 587M	0.82
-	✓	✓	139M / 587M	23.39
✓	✓	-	152M / 638M	12.90
✓	✓	✓	190M / 638M	24.10

- Fine-tuning cross-attention is crucial to transfer to multilingual ST

Transfer learning

Adapter ENC	Adapter DEC	Finetune xattn	Params trainable/total	BLEU
-	-	✓	38M / 486M	19.52
-	✓	-	101M / 587M	0.82
-	✓	✓	139M / 587M	23.39
✓	✓	-	152M / 638M	12.90
✓	✓	✓	190M / 638M	24.10

- Fine-tuning cross-attention is crucial to transfer to multilingual ST
- Adding adapters to the backbone decoder or to both encoder and decoder further boosts performance.

Transfer learning

Adapter ENC	Adapter DEC	Finetune xattn	Params trainable/total	BLEU
-	-	✓	38M / 486M	19.52
-	✓	-	101M / 587M	0.82
-	✓	✓	139M / 587M	23.39
✓	✓	-	152M / 638M	12.90
✓	✓	✓	190M / 638M	24.10

- Fine-tuning cross-attention is crucial to transfer to multilingual ST
- Adding adapters to the backbone decoder or to both encoder and decoder further boosts performance.

→ **Adapter is able to connect off-the-shelf models in a modular fashion.**

Conclusion

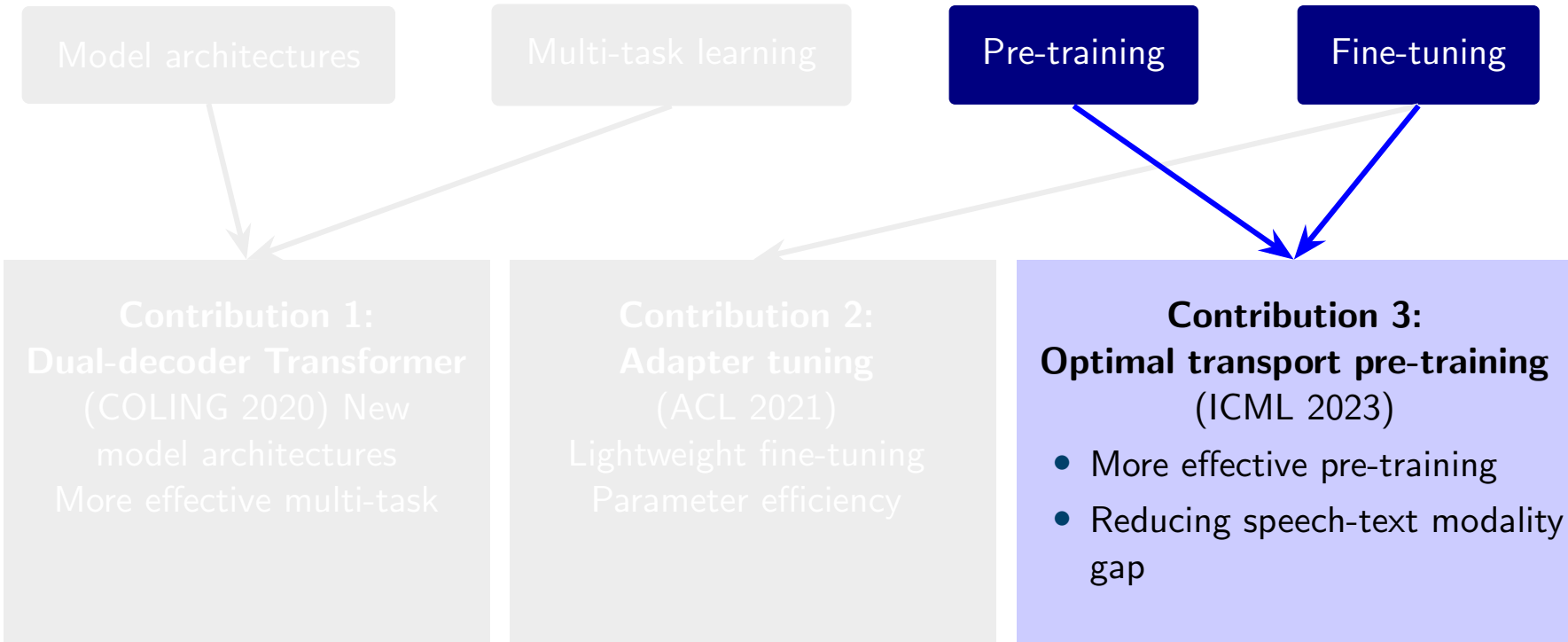
Conclusion

- Both versatility and specialization can be achieved in a multilingual system with adapters: better performance, high modularity, low maintenance cost

Conclusion

- Both versatility and specialization can be achieved in a multilingual system with adapters: better performance, high modularity, low maintenance cost
- Adapters can be used as a glue to connect off-the-shelf systems.

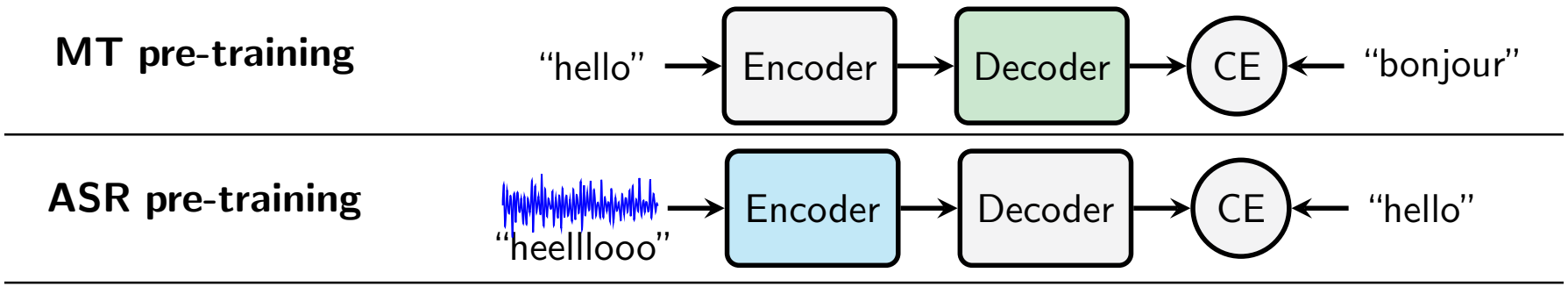
Contribution 3: Optimal transport for pre-training



Phuong-Hang Le, Hongyu Gong, Changhan Wang, Juan Pino, Benjamin Lecouteux, and Didier Schwab. Pre-training for Speech Translation: CTC Meets Optimal Transport. In *International Conference on Machine Learning* (**ICML 2023, oral**).

Motivation: Modality gap in standard pre-training method

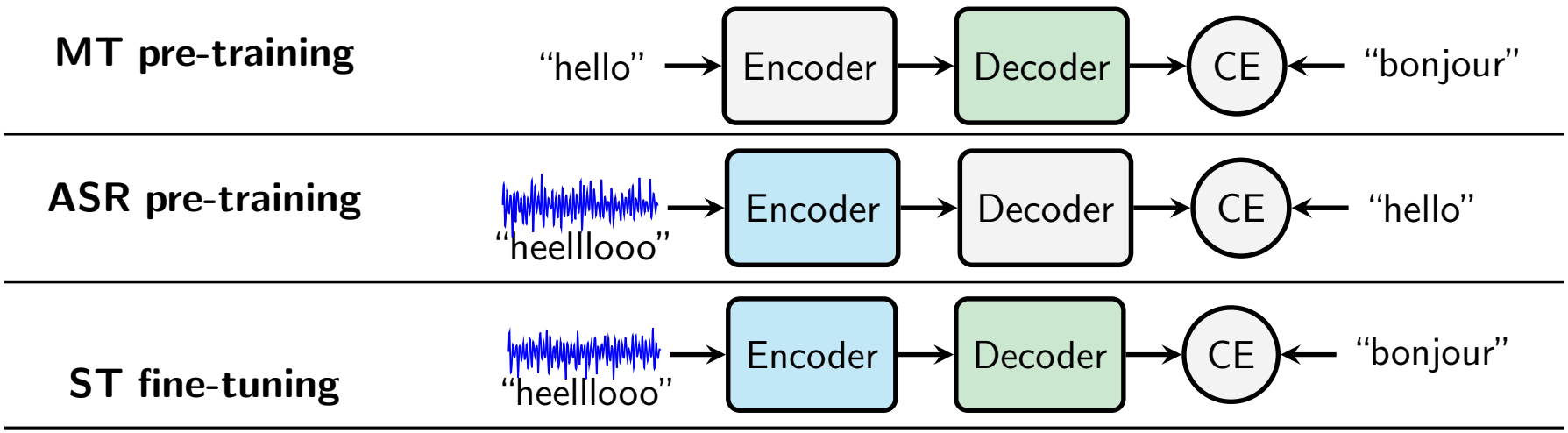
Standard ASR & MT pre-training for ST



CE: cross-entropy loss

Motivation: Modality gap in standard pre-training method

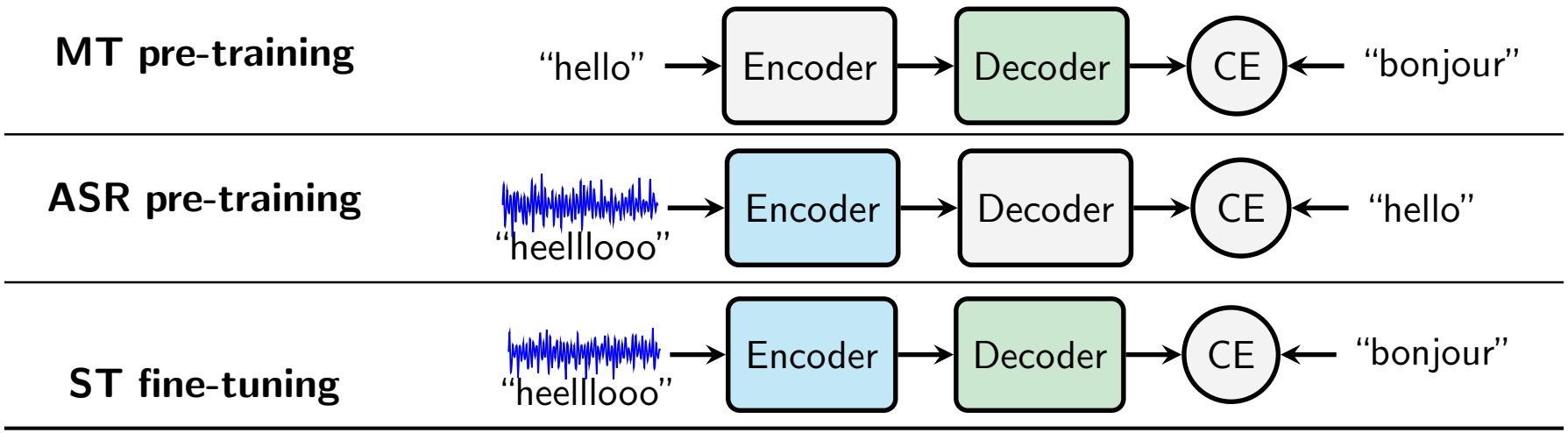
Standard ASR & MT pre-training for ST



CE: cross-entropy loss

Motivation: Modality gap in standard pre-training method

Standard ASR & MT pre-training for ST

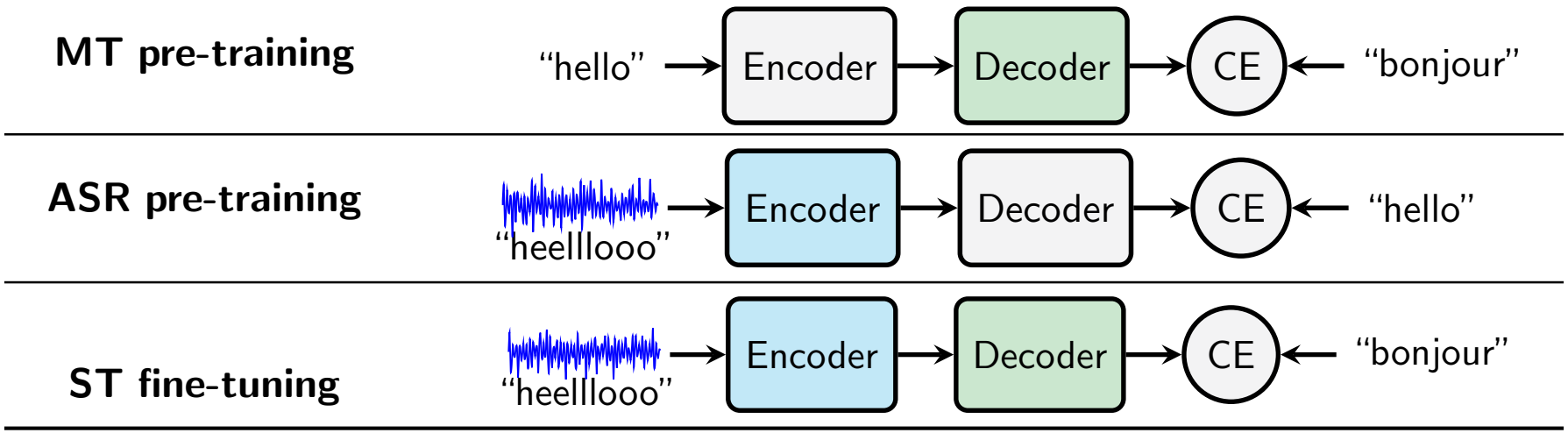


CE: cross-entropy loss

MT decoder has learned to align [Bahdanau et al., 2015] with text encoder and not with speech encoder.

Motivation: Modality gap in standard pre-training method

Standard ASR & MT pre-training for ST

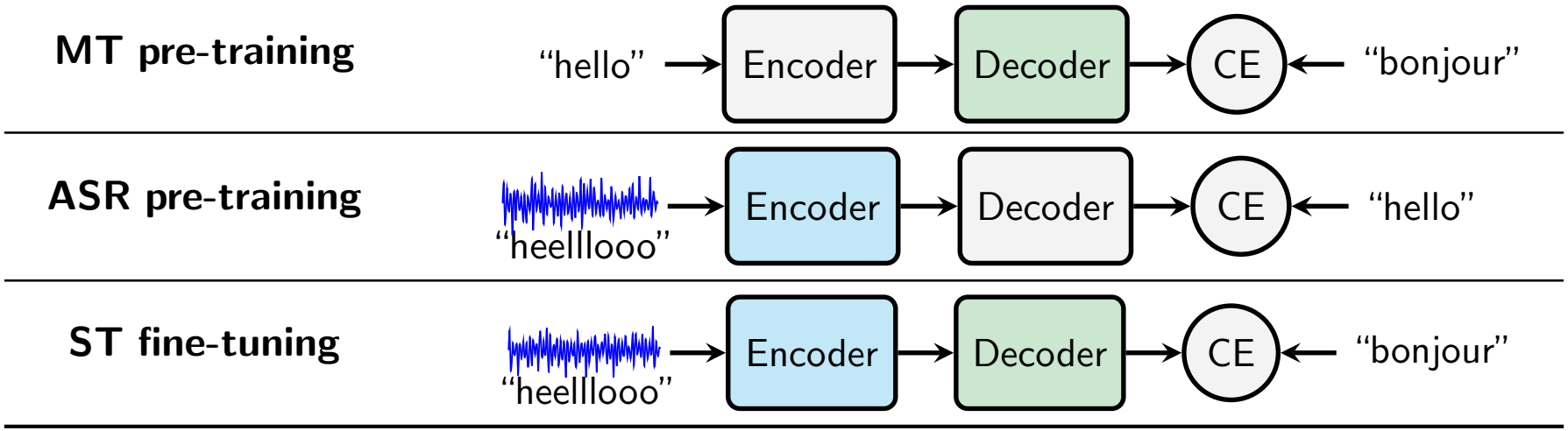


CE: cross-entropy loss

MT decoder has learned to align [Bahdanau et al., 2015] with text encoder and not with speech encoder.

Speech encoder has learned to align with ASR decoder and not with MT decoder.

Motivation: Modality gap in standard pre-training method



Lost of pre-trained alignment information: MT encoder and ASR decoder being discarded for fine-tuning.

Plugging MT decoder to speech encoder: mismatch, caused by **modality gap**.

Motivation: Modality gap in standard pre-training method

Definition of **modality gap:** **discrepancy between speech features and text features** (of the same sentence).

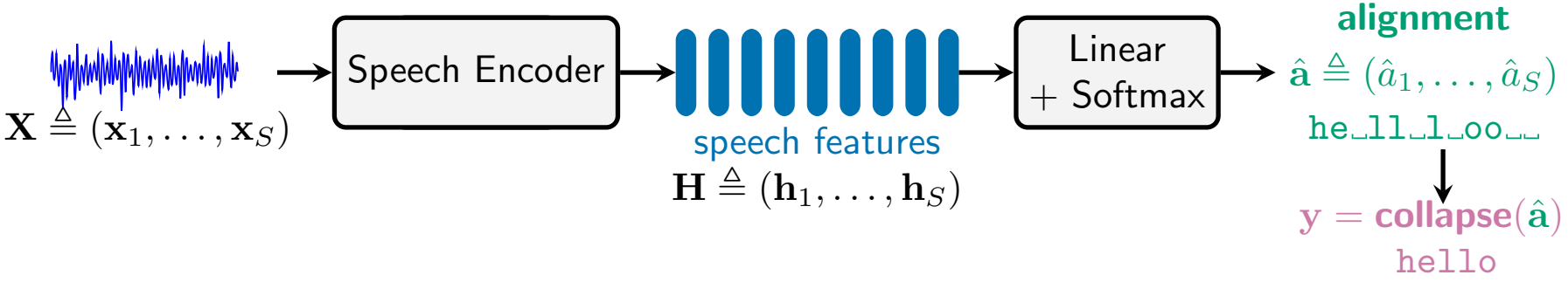
Motivation: Modality gap in standard pre-training method

Definition of **modality gap:** **discrepancy between speech features and text features** (of the same sentence).

In our case, that is discrepancy between text encoder output and speech encoder output.

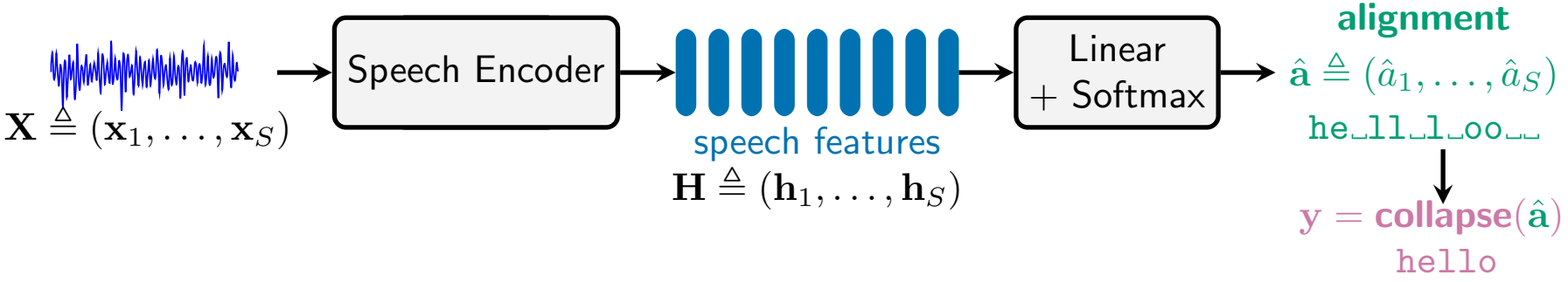
First proposal: use CTC to reduce modality gap

Review of CTC



First proposal: use CTC to reduce modality gap

Review of CTC

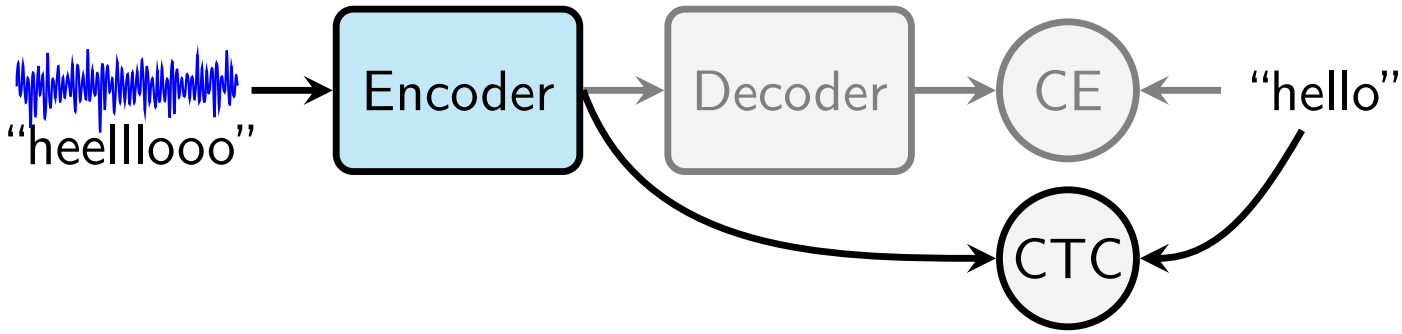


- CTC [Graves et al., 2006] predicts a token $\hat{a}_t \in \mathcal{V}$ for each time step t:

$$p(a_t \mid \mathbf{X}) = \text{softmax}(\mathbf{W}\mathbf{h}_t + \mathbf{b})[a_t] \;\forall a_t \in \mathcal{V},$$
$$\hat{a}_t = \underset{a_t \in \mathcal{V}}{\text{argmax}}\, p(a_t \mid \mathbf{X}), \tag{1}$$

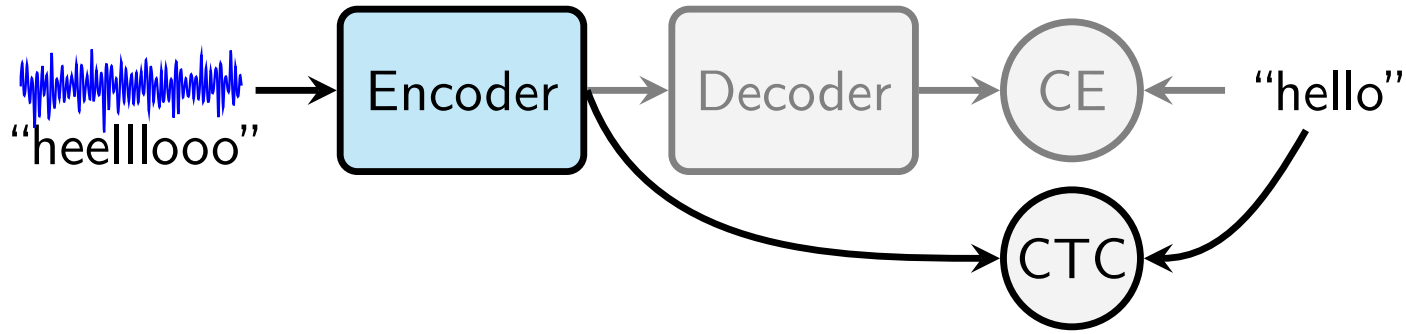
where $\mathbf{W} \in \mathbb{R}^{V \times d}, \mathbf{b} \in \mathbb{R}^V$ are weights and biases of final linear layer, and $\mathbf{v}[i]$ denotes i^{th} element of vector $\mathbf{v}$.

CTC can reduce modality gap



ASR pre-training with CTC. *CE is optional.*

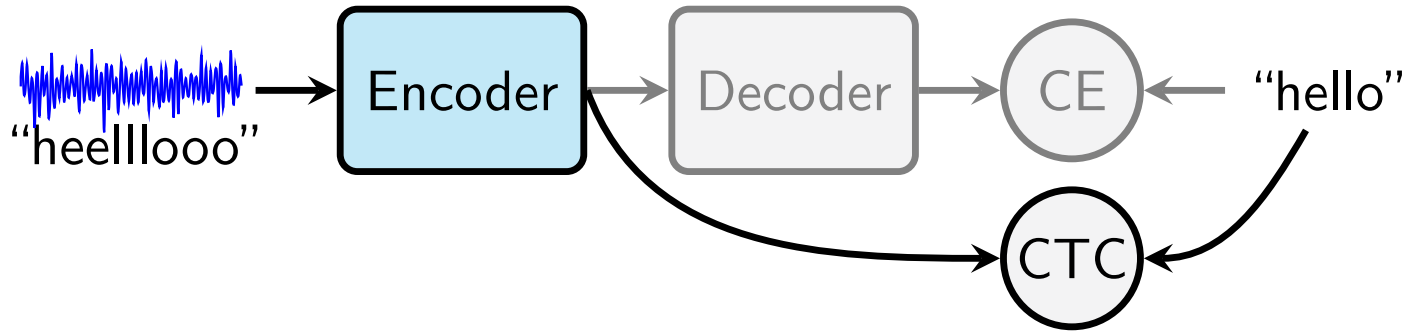
CTC can reduce modality gap



ASR pre-training with CTC. *CE is optional.*

- ASR encoder trained with CTC already learns to align speech input to text output without a decoder.

CTC can reduce modality gap

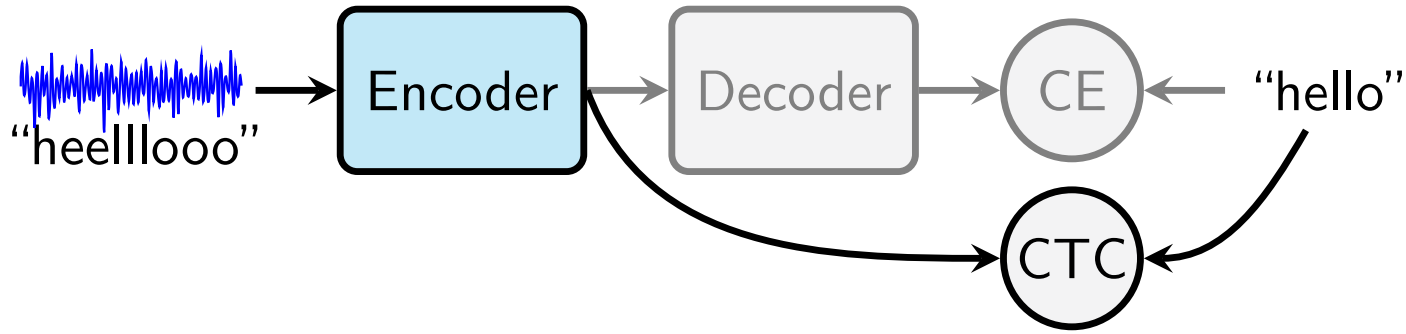


ASR pre-training with CTC. *CE is optional.*

- ASR encoder trained with CTC already learns to align speech input to text output without a decoder.

→ No loss of pre-trained alignment information in encoder during ST fine-tuning.

CTC can reduce modality gap



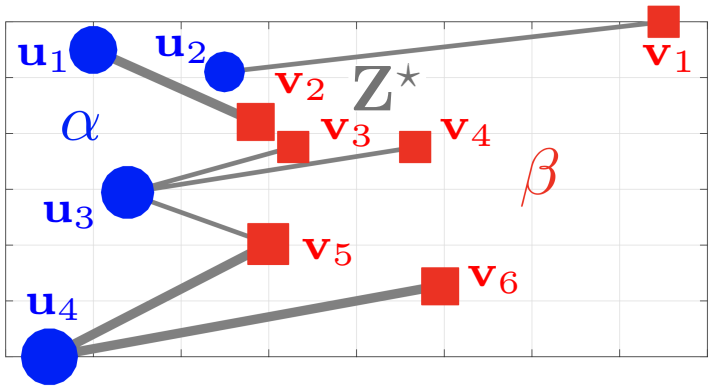
ASR pre-training with CTC. *CE is optional.*

- ASR encoder trained with CTC already learns to align speech input to text output without a decoder.

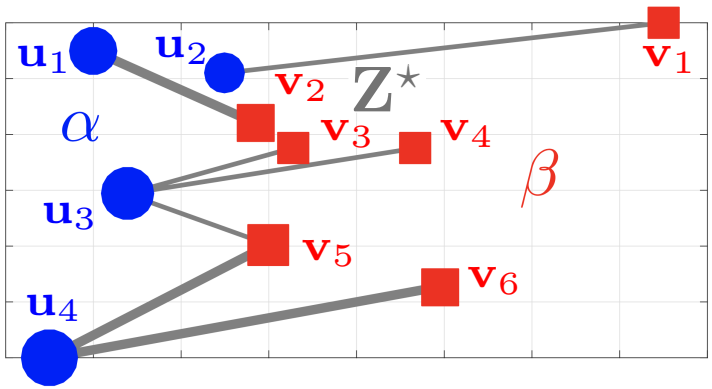
→ No loss of pre-trained alignment information in encoder during ST fine-tuning.

- Loss of alignment information due to discarding MT encoder cannot be recovered by CTC alone.

Second proposal: use optimal transport to further alleviate modality gap

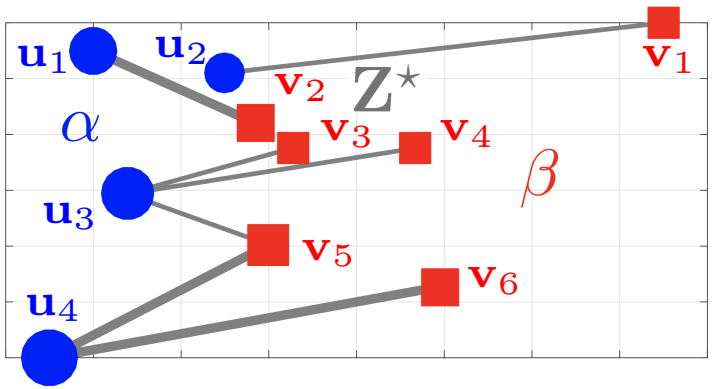


Second proposal: use optimal transport to further alleviate modality gap



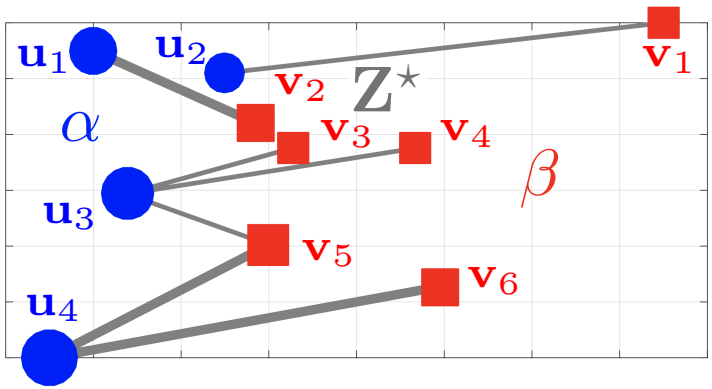
- $\mathbf{u}_1, \ldots, \mathbf{u}_m \in \mathbb{R}^d$, and $\mathbf{v}_1, \ldots, \mathbf{v}_n \in \mathbb{R}^d$: *locations* of the masses of α and β.

Second proposal: use optimal transport to further alleviate modality gap



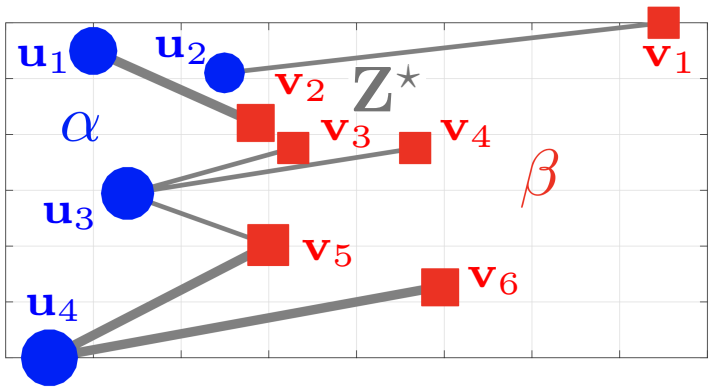
- $\mathbf{u}_1, \ldots, \mathbf{u}_m \in \mathbb{R}^d$, and $\mathbf{v}_1, \ldots, \mathbf{v}_n \in \mathbb{R}^d$: *locations* of the masses of α and β.
- $a_1, \ldots, a_m$ and $b_1, \ldots, b_n$: possitive *masses* of 2 discrete probability distributions α and β, respectively ($\sum_{i=1}^m a_i = 1$, $\sum_{j=1}^n b_j = 1$).

Second proposal: use optimal transport to further alleviate modality gap



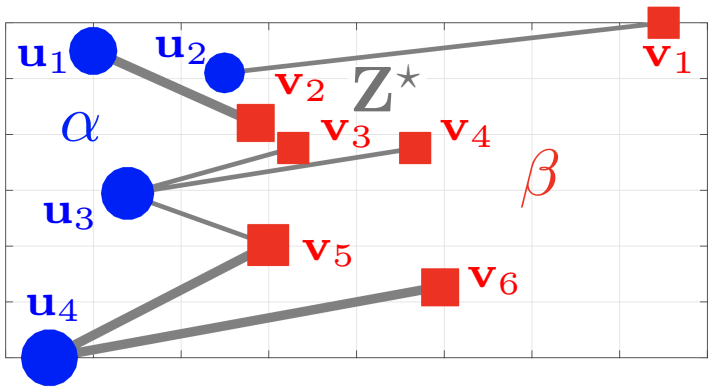
- $\mathbf{u}_1, \ldots, \mathbf{u}_m \in \mathbb{R}^d$, and $\mathbf{v}_1, \ldots, \mathbf{v}_n \in \mathbb{R}^d$: *locations* of the masses of α and β.
- $a_1, \ldots, a_m$ and $b_1, \ldots, b_n$: possitive *masses* of 2 discrete probability distributions α and β, respectively ($\sum_{i=1}^m a_i = 1$, $\sum_{j=1}^n b_j = 1$).
- $c(\mathbf{u}_i, \mathbf{v}_j)$ where $c : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}_+$: *cost* of transporting a unit of mass from $\mathbf{u}_i$ to $\mathbf{v}_j$.

Second proposal: use optimal transport to further alleviate modality gap



- $\mathbf{u}_1, \ldots, \mathbf{u}_m \in \mathbb{R}^d$, and $\mathbf{v}_1, \ldots, \mathbf{v}_n \in \mathbb{R}^d$: *locations* of the masses of α and β.
- $a_1, \ldots, a_m$ and $b_1, \ldots, b_n$: possitive *masses* of 2 discrete probability distributions α and β, respectively ($\sum_{i=1}^m a_i = 1$, $\sum_{j=1}^n b_j = 1$).
- $c(\mathbf{u}_i, \mathbf{v}_j)$ where $c : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}_+$: *cost* of transporting a unit of mass from $\mathbf{u}_i$ to $\mathbf{v}_j$.
- $Z_{ij} \geq 0$: *quantity* of mass to be transported from $\mathbf{u}_i$ to $\mathbf{v}_j$.

Second proposal: use optimal transport to further alleviate modality gap



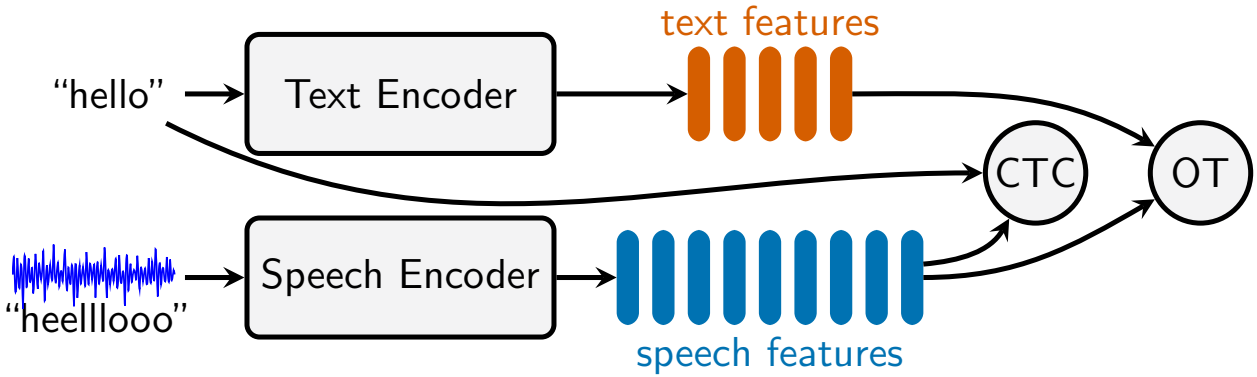
- $\mathbf{u}_1, \ldots, \mathbf{u}_m \in \mathbb{R}^d$, and $\mathbf{v}_1, \ldots, \mathbf{v}_n \in \mathbb{R}^d$: *locations* of the masses of α and β.
- $a_1, \ldots, a_m$ and $b_1, \ldots, b_n$: possitive *masses* of 2 discrete probability distributions α and β, respectively ($\sum_{i=1}^m a_i = 1$, $\sum_{j=1}^n b_j = 1$).
- $c(\mathbf{u}_i, \mathbf{v}_j)$ where $c : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}_+$: *cost* of transporting a unit of mass from $\mathbf{u}_i$ to $\mathbf{v}_j$.
- $Z_{ij} \geq 0$: *quantity* of mass to be transported from $\mathbf{u}_i$ to $\mathbf{v}_j$.

OT finds the transportation plan $\mathbf{Z}^*$ that has the minimum cost:

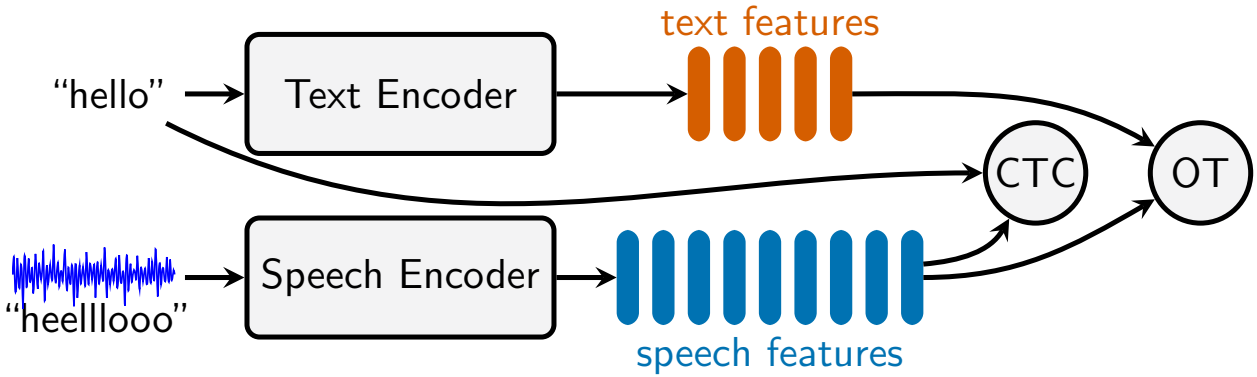
$$\min_{\mathbf{Z} \in \mathbb{R}^{m \times n}} \quad \sum_{i=1}^{m} \sum_{j=1}^{n} Z_{ij} c(\mathbf{u}_i, \mathbf{v}_j),$$

$$\text{s.t.} \quad \mathbf{Z} \geq \mathbf{0}, \ \sum_{j=1}^{n} Z_{ij} = a_i, \ \sum_{i=1}^{m} Z_{ij} = b_j.$$

Learning to align speech and text features

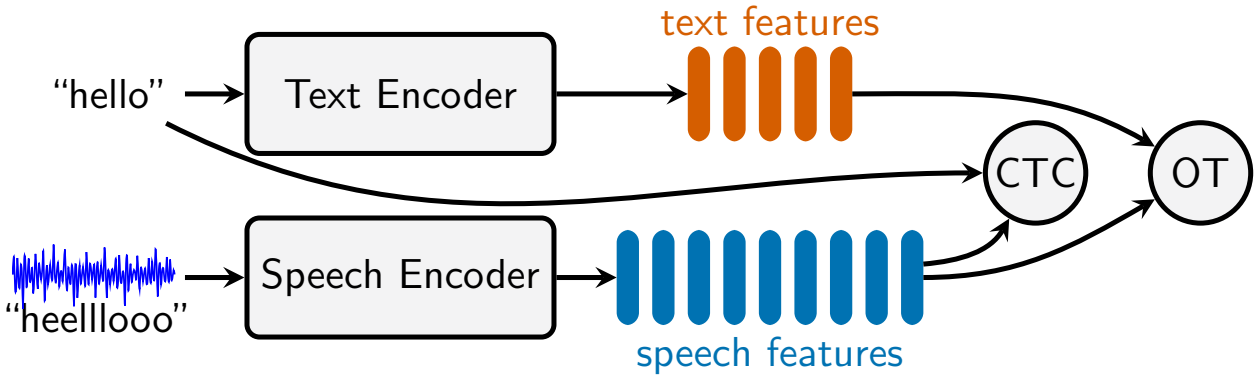


Learning to align speech and text features



Siamese network for speech-text alignment. *OT pulls speech and text features closer in Wasserstein space, while CTC further enhances speech features.*

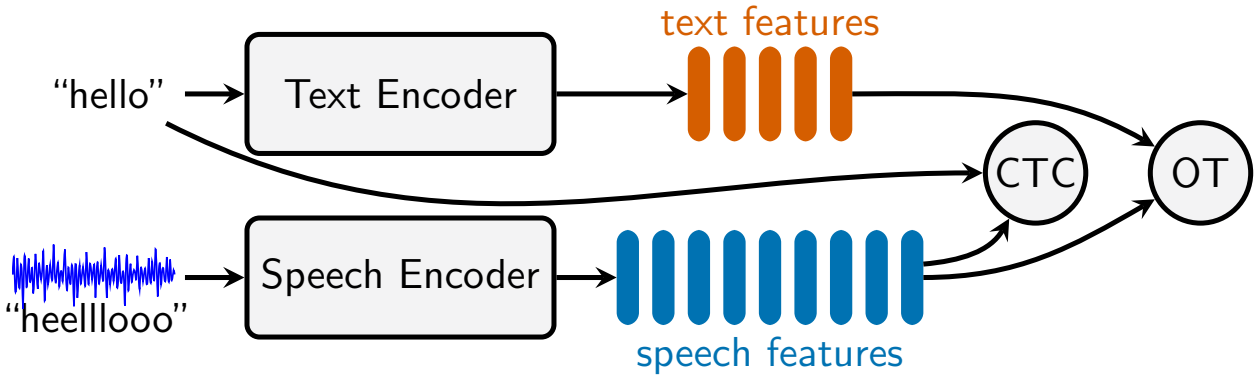
Learning to align speech and text features



Siamese network for speech-text alignment. *OT pulls speech and text features closer in Wasserstein space, while CTC further enhances speech features.*

Given speech features $\mathbf{U} = (\mathbf{u}_1, \ldots, \mathbf{u}_m)$, text features $\mathbf{V} = (\mathbf{v}_1, \ldots, \mathbf{v}_n)$ $(\mathbf{u}_i, \mathbf{v}_j \in \mathbb{R}^d)$. The OT loss is defined as:

Learning to align speech and text features



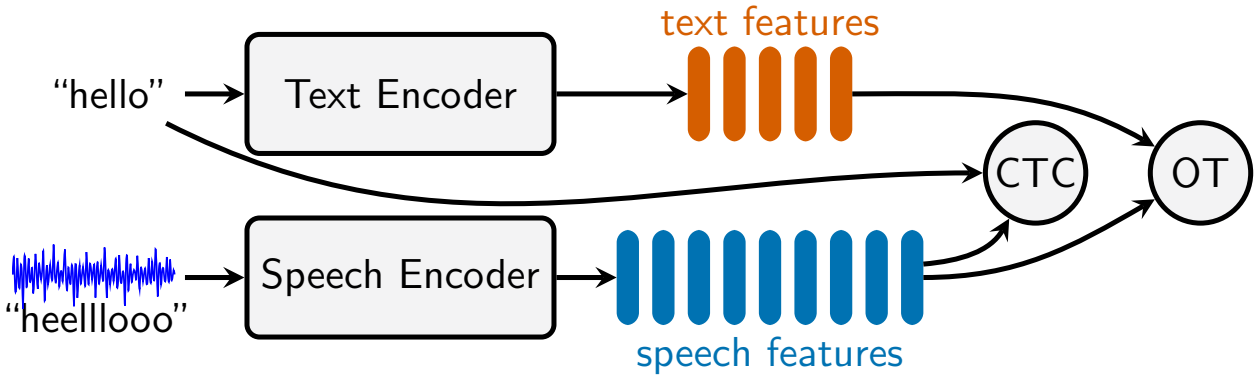
Siamese network for speech-text alignment. *OT pulls speech and text features closer in Wasserstein space, while CTC further enhances speech features.*

Given speech features $\mathbf{U} = (\mathbf{u}_1, \ldots, \mathbf{u}_m)$, text features $\mathbf{V} = (\mathbf{v}_1, \ldots, \mathbf{v}_n)$ $(\mathbf{u}_i, \mathbf{v}_j \in \mathbb{R}^d)$. The OT loss is defined as:

$$\mathbf{OT}(\mathbf{U}, \mathbf{V}) = \min_{\mathbf{Z} \in \mathbb{R}^{m \times n}} \sum_{i=1}^{m} \sum_{j=1}^{n} Z_{ij} c(\mathbf{u}_i, \mathbf{v}_j),$$

$$\text{s.t.} \quad \mathbf{Z} \geq \mathbf{0}, \ \sum_{j=1}^{n} Z_{ij} = \frac{1}{m}, \ \sum_{i=1}^{m} Z_{ij} = \frac{1}{n}.$$

Learning to align speech and text features



Siamese network for speech-text alignment. *OT pulls speech and text features closer in Wasserstein space, while CTC further enhances speech features.*

Given speech features $\mathbf{U} = (\mathbf{u}_1, \ldots, \mathbf{u}_m)$, text features $\mathbf{V} = (\mathbf{v}_1, \ldots, \mathbf{v}_n)$ $(\mathbf{u}_i, \mathbf{v}_j \in \mathbb{R}^d)$. The OT loss is defined as:

$$\mathbf{OT}(\mathbf{U}, \mathbf{V}) = \min_{\mathbf{Z} \in \mathbb{R}^{m \times n}} \sum_{i=1}^{m} \sum_{j=1}^{n} Z_{ij} c(\mathbf{u}_i, \mathbf{v}_j),$$

$$\text{s.t.} \quad \mathbf{Z} \geq \mathbf{0}, \ \sum_{j=1}^{n} Z_{ij} = \frac{1}{m}, \ \sum_{i=1}^{m} Z_{ij} = \frac{1}{n}.$$

$\mathbf{Z}^*$ can be seen as an alignment map between 2 sequences.

Positional encoding for OT

Positional encoding for OT

- **Motivation:** OT loss ignores sequence orders, while our encoder inputs are *monotonically* aligned.

Positional encoding for OT

- **Motivation:** OT loss ignores sequence orders, while our encoder inputs are *monotonically* aligned.
- **Idea:** integrating normalized positions $s_i = \frac{i-1}{m-1}$ and $t_j = \frac{j-1}{n-1}$ into cost function:

$$c(\mathbf{u}_i, \mathbf{v}_j) = \left(\|\mathbf{u}_i - \mathbf{v}_j\|_p^p + \gamma^p |s_i - t_j|^p \right)^{1/p}$$
$$= \|\mathbf{u}'_i - \mathbf{v}'_j\|_p$$

where $\mathbf{u}'_i = [\mathbf{u}_i; \gamma s_i]$ and $\mathbf{v}'_j = [\mathbf{v}_j; \gamma t_j]$.

Positional encoding for OT

- **Motivation:** OT loss ignores sequence orders, while our encoder inputs are *monotonically* aligned.
- **Idea:** integrating normalized positions $s_i = \frac{i-1}{m-1}$ and $t_j = \frac{j-1}{n-1}$ into cost function:

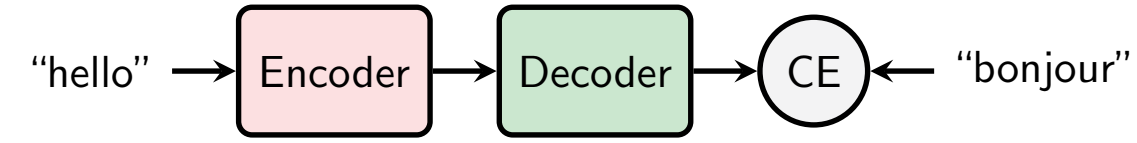
$$c(\mathbf{u}_i, \mathbf{v}_j) = \left(\|\mathbf{u}_i - \mathbf{v}_j\|_p^p + \gamma^p |s_i - t_j|^p \right)^{1/p}$$
$$= \|\mathbf{u}'_i - \mathbf{v}'_j\|_p$$

  where $\mathbf{u}'_i = [\mathbf{u}_i; \gamma s_i]$ and $\mathbf{v}'_j = [\mathbf{v}_j; \gamma t_j]$.
- **Intuition:** Transport from $\mathbf{u}_1$ to $\mathbf{v}_1$ (or from $\mathbf{u}_m$ to $\mathbf{v}_n$) should induce a low cost while transporting from $\mathbf{u}_1$ to $\mathbf{v}_n$ should induce a high cost.

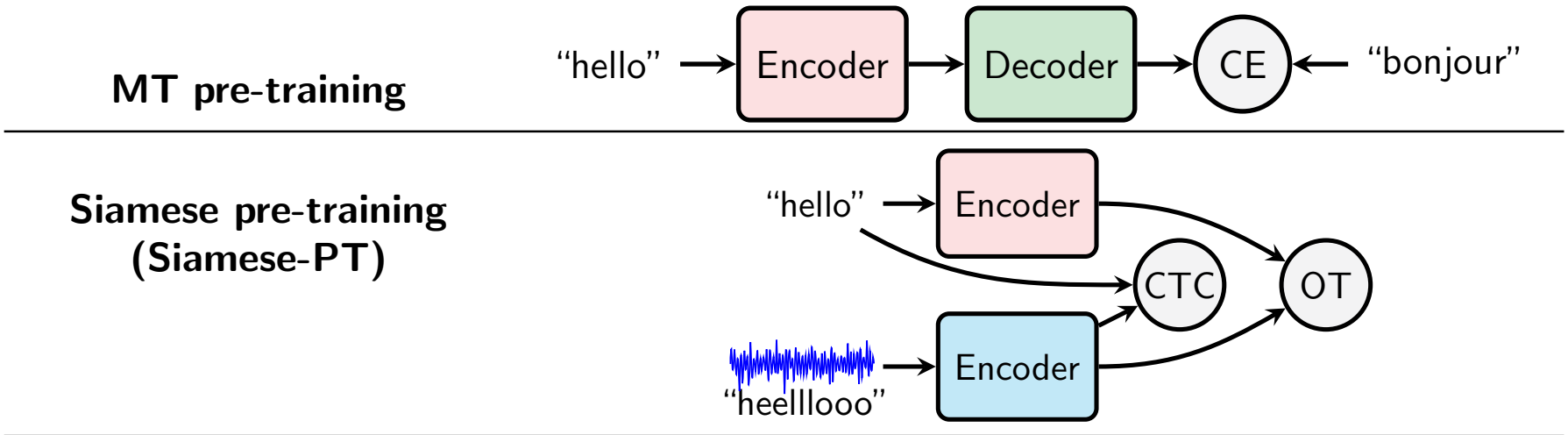
Proposed Siamese pre-training recipe for speech translation

Proposed Siamese pre-training recipe for speech translation

MT pre-training

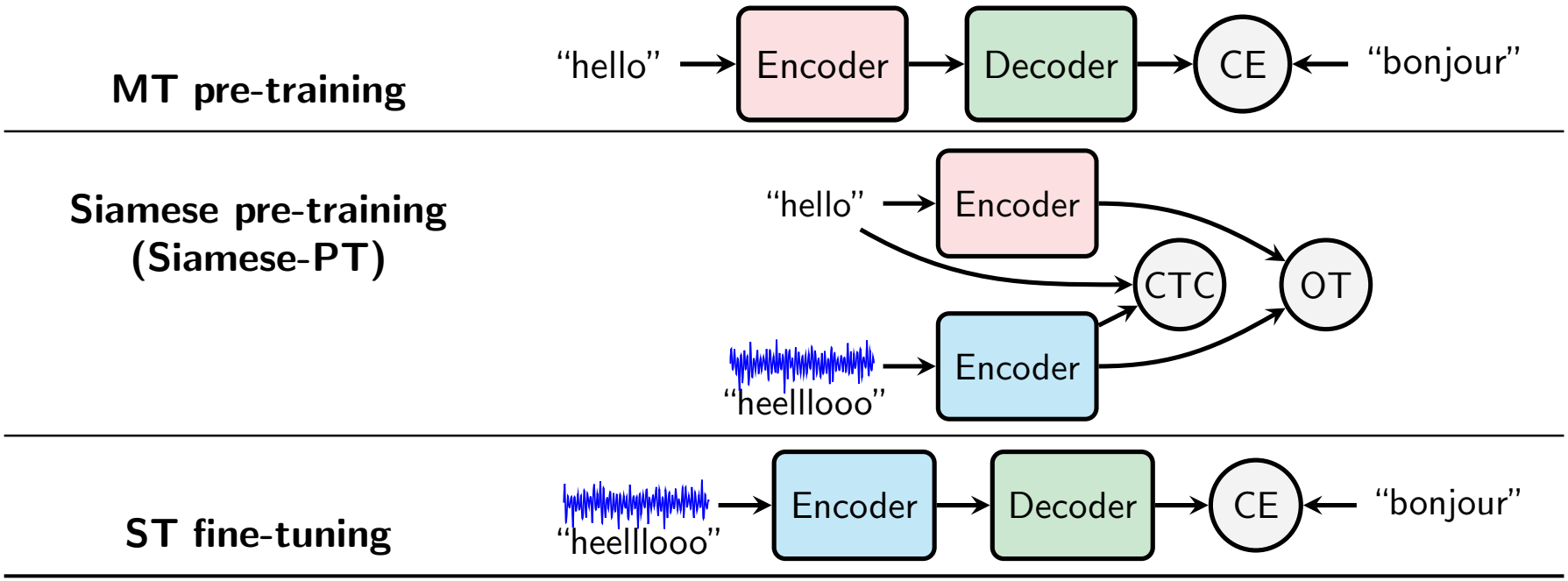


Proposed Siamese pre-training recipe for speech translation



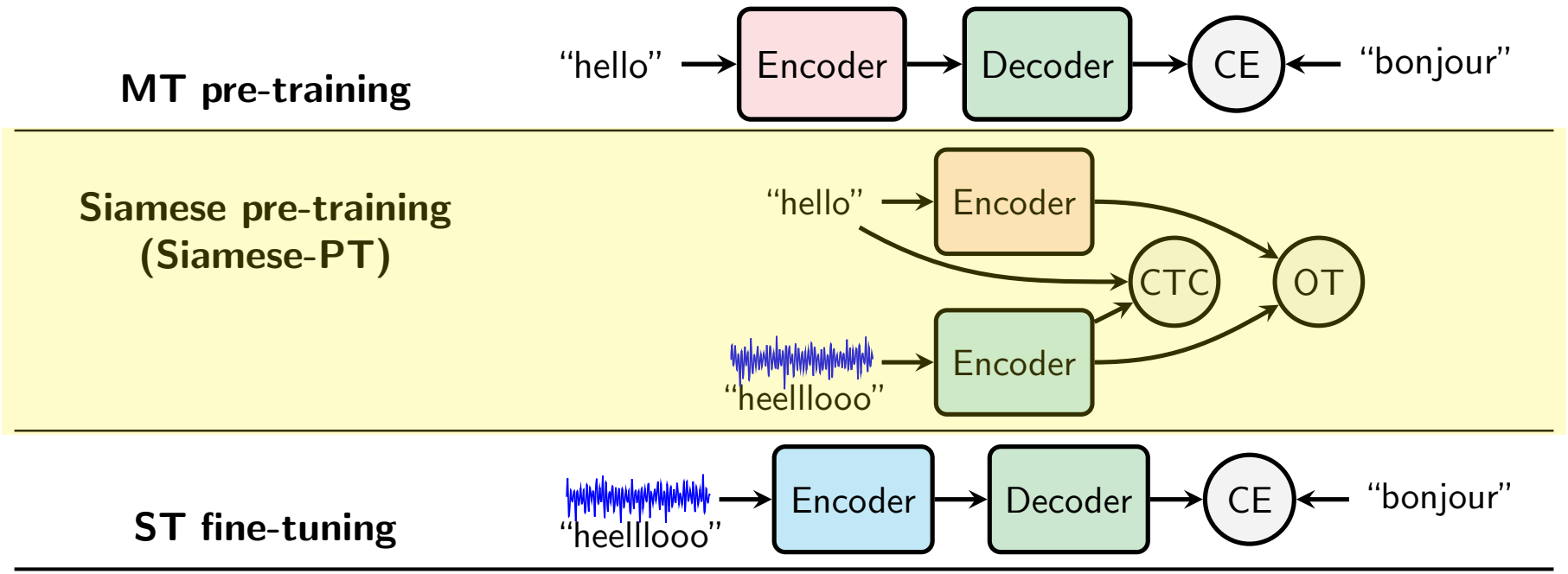
Pre-trained MT encoder is used by OT in Siamese-PT step.

Proposed Siamese pre-training recipe for speech translation



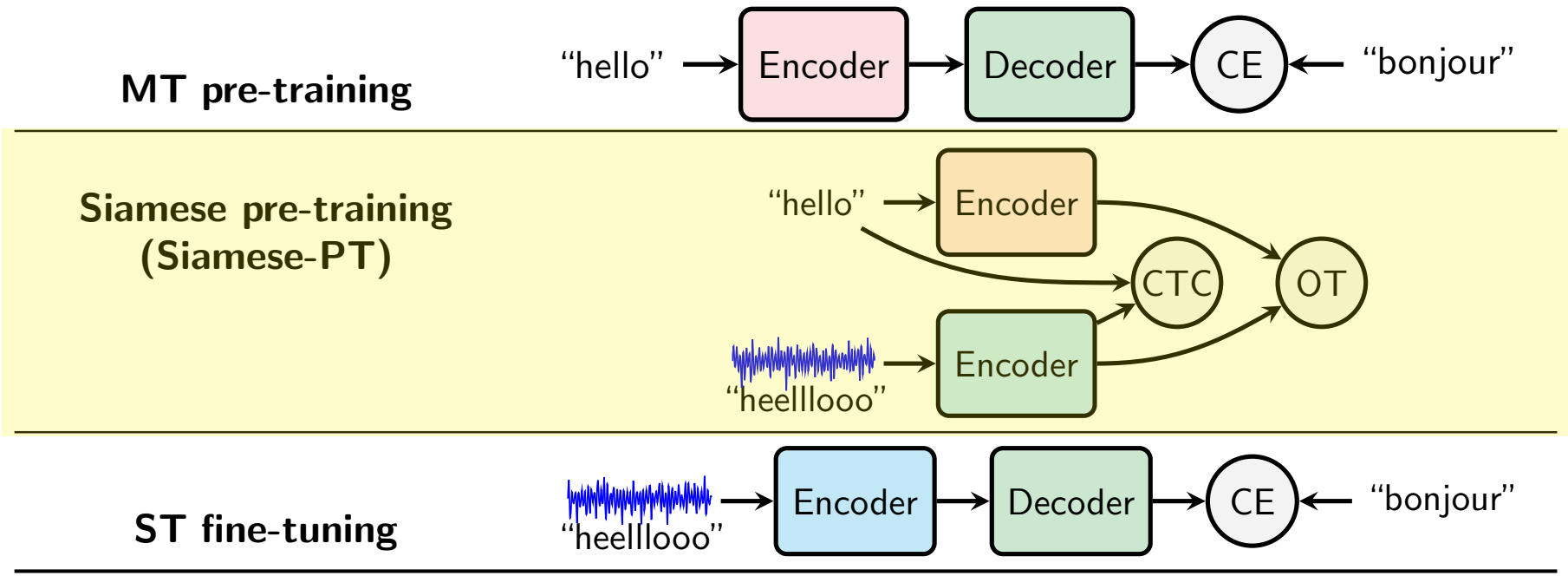
Pre-trained MT encoder is used by OT in Siamese-PT step.

Proposed Siamese pre-training recipe for speech translation



Pre-trained MT encoder is used by OT in Siamese-PT step.

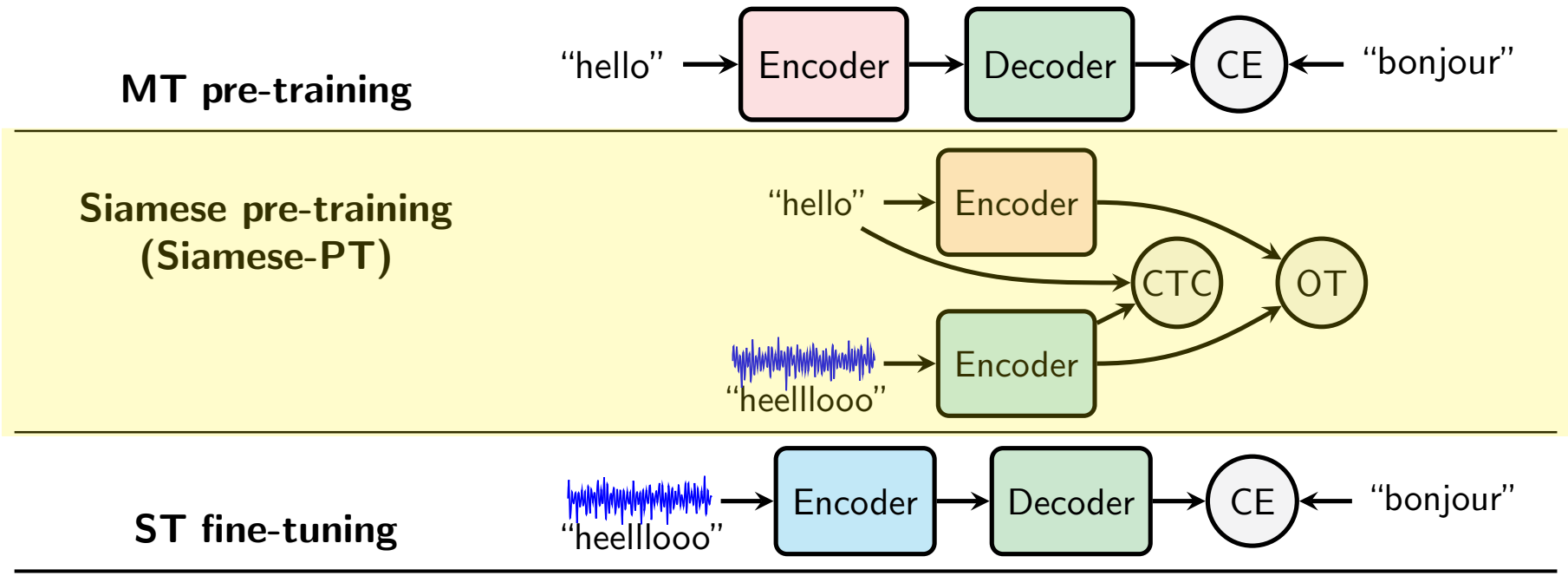
Proposed Siamese pre-training recipe for speech translation



Pre-trained MT encoder is used by OT in Siamese-PT step.

- All pre-trained components are used.

Proposed Siamese pre-training recipe for speech translation



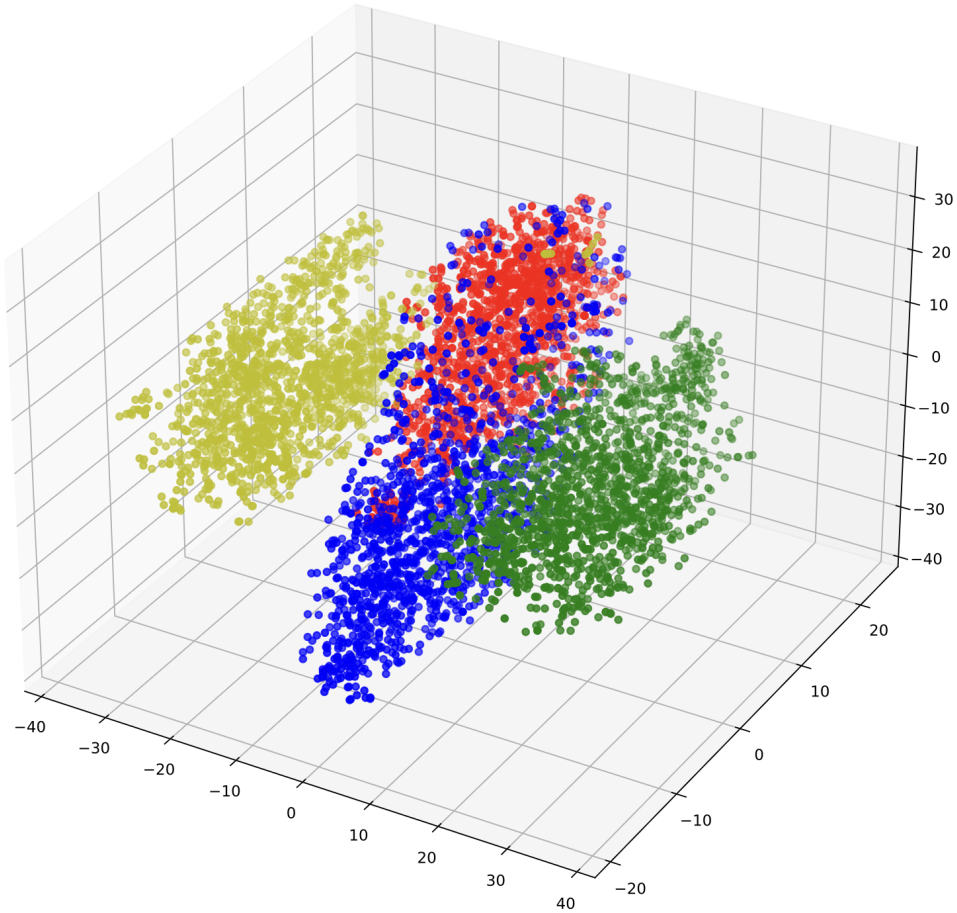
Pre-trained MT encoder is used by OT in Siamese-PT step.

- All pre-trained components are used.
- Optimal transport reduces modality gap by aligning speech and text features.

Qualitative results: Modality gap visualization

t-SNE dimensionality reduction [Van der Maaten and Hinton, 2008] of encoder outputs:

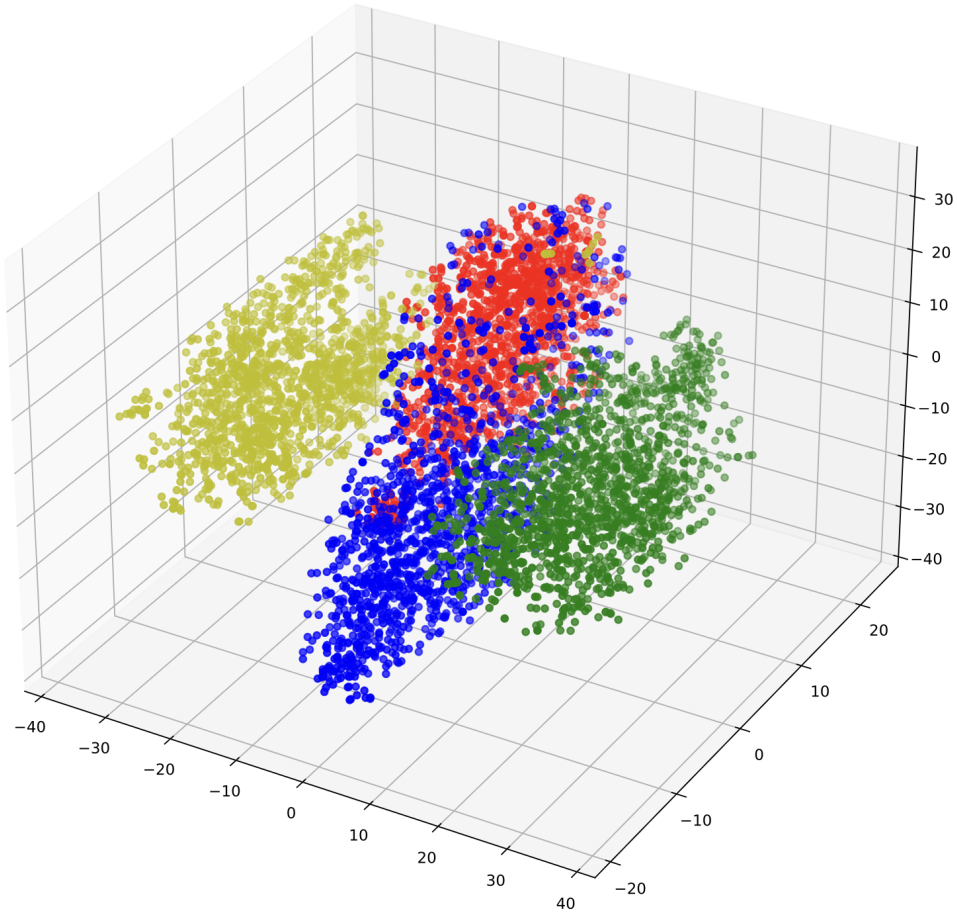
Text encoder
Speech encoder (CE)
Speech encoder (CTC)
Speech encoder (CTC + OT)



Qualitative results: Modality gap visualization

t-SNE dimensionality reduction [Van der Maaten and Hinton, 2008] of encoder outputs:

Text encoder
Speech encoder (CE)
Speech encoder (CTC)
Speech encoder (CTC + OT)



The results support our claims: **CTC can reduce modality gap and optimal transport can improve even further**.

(Reminder: modality gap is discrepancy between speech and text features.)

Experimental results

OT vs. other distances

Metric	**Length-match**	**En→De**	**En→Fr**
(CTC alone)	-	24.08 (17.7)	29.91 (17.3)
Euclidean	average	24.41 (18.1)	29.86 (17.3)
	attention	24.30 (18.6)	29.81 (19.2)
	interpolation	23.94 (19.3)	29.77 (17.8)
KL-diverg.	attention	24.56 (18.3)	30.10 (**17.1**)
	interpolation	24.26 (18.1)	29.96 (17.4)
Adversarial	-	23.73 (20.6)	29.98 (19.6)
OT	-	**24.74** (**17.6**)	**30.31** (**17.1**)

Table: BLEU and WER (in parentheses) for different distance functions in Siamese pre-training

Experimental results

OT vs. other distances

Metric	**Length-match**	**En→De**	**En→Fr**
(CTC alone)	-	24.08 (17.7)	29.91 (17.3)
Euclidean	average	24.41 (18.1)	29.86 (17.3)
	attention	24.30 (18.6)	29.81 (19.2)
	interpolation	23.94 (19.3)	29.77 (17.8)
KL-diverg.	attention	24.56 (18.3)	30.10 (**17.1**)
	interpolation	24.26 (18.1)	29.96 (17.4)
Adversarial	-	23.73 (20.6)	29.98 (19.6)
OT	-	**24.74** (**17.6**)	**30.31** (**17.1**)

Table: BLEU and WER (in parentheses) for different distance functions in Siamese pre-training

Experimental results

OT vs. other distances

Metric	Length-match	En→De	En→Fr
(CTC alone)	-	24.08 (17.7)	29.91 (17.3)
Euclidean	average	24.41 (18.1)	29.86 (17.3)
	attention	24.30 (18.6)	29.81 (19.2)
	interpolation	23.94 (19.3)	29.77 (17.8)
KL-diverg.	attention	24.56 (18.3)	30.10 (**17.1**)
	interpolation	24.26 (18.1)	29.96 (17.4)
Adversarial	-	23.73 (20.6)	29.98 (19.6)
OT	-	**24.74** (**17.6**)	**30.31** (**17.1**)

Table: BLEU and WER (in parentheses) for different distance functions in Siamese pre-training

Experimental results

OT vs. other distances

Metric	Length-match	En→De	En→Fr
(CTC alone)	-	24.08 (17.7)	29.91 (17.3)
Euclidean	average	24.41 (18.1)	29.86 (17.3)
	attention	24.30 (18.6)	29.81 (19.2)
	interpolation	23.94 (19.3)	29.77 (17.8)
KL-diverg.	attention	24.56 (18.3)	30.10 (**17.1**)
	interpolation	24.26 (18.1)	29.96 (17.4)
Adversarial	-	23.73 (20.6)	29.98 (19.6)
OT	-	**24.74** (**17.6**)	**30.31** (**17.1**)

Table: BLEU and WER (in parentheses) for different distance functions in Siamese pre-training

Experimental results

OT vs. other distances

Metric	Length-match	En→De	En→Fr
(CTC alone)	-	24.08 (17.7)	29.91 (17.3)
Euclidean	average	24.41 (18.1)	29.86 (17.3)
	attention	24.30 (18.6)	29.81 (19.2)
	interpolation	23.94 (19.3)	29.77 (17.8)
KL-diverg.	attention	24.56 (18.3)	30.10 (**17.1**)
	interpolation	24.26 (18.1)	29.96 (17.4)
Adversarial	-	23.73 (20.6)	29.98 (19.6)
OT	-	**24.74** (**17.6**)	**30.31** (**17.1**)

Table: BLEU and WER (in parentheses) for different distance functions in Siamese pre-training

Experimental results

Siamese-PT vs. existing supervised pre-training approaches

Method	**En→X**	**X**→En**
Wang et al. [2020b]	19.4	24.5
CE	19.2	24.6
CTC	19.8	24.7
CTC+CE	19.7	24.5
Siamese-PT	**21.5**	**25.5**

Table: Translation performance in BLEU on CoVoST-2

Experimental results

Siamese-PT vs. existing supervised pre-training approaches

Method	**En→X**	**X**→En**
Wang et al. [2020b]	19.4	24.5
CE	19.2	24.6
CTC	19.8	24.7
CTC+CE	19.7	24.5
Siamese-PT	**21.5**	**25.5**

Table: Translation performance in BLEU on CoVoST-2

Experimental results

Siamese-PT vs. existing supervised pre-training approaches

Method	**En→X**	**X**→En**
Wang et al. [2020b]	19.4	24.5
CE	19.2	24.6
CTC	19.8	24.7
CTC+CE	19.7	24.5
Siamese-PT	**21.5**	**25.5**

Table: Translation performance in BLEU on CoVoST-2

Experimental results

Siamese-PT vs. existing supervised pre-training approaches

Method	**En→X**	**X**→En**
Wang et al. [2020b]	19.4	24.5
CE	19.2	24.6
CTC	19.8	24.7
CTC+CE	19.7	24.5
Siamese-PT	**21.5**	**25.5**

Table: Translation performance in BLEU on CoVoST-2

Experimental results

Siamese-PT vs. existing supervised pre-training approaches

Method	**En→X**	**X**→En**
Wang et al. [2020b]	19.4	24.5
CE	19.2	24.6
CTC	19.8	24.7
CTC+CE	19.7	24.5
Siamese-PT	**21.5**	**25.5**

Table: Translation performance in BLEU on CoVoST-2

Experimental results

Siamese-PT vs. existing supervised pre-training approaches

Method	**En→X**	**X**→En**
Wang et al. [2020b]	19.4	24.5
CE	19.2	24.6
CTC	19.8	24.7
CTC+CE	19.7	24.5
Siamese-PT	**21.5**	**25.5**

Table: Translation performance in BLEU on CoVoST-2

Experimental results

Siamese-PT vs. existing supervised pre-training approaches

Method	**En→X**	**X**→En**
Wang et al. [2020b]	19.4	24.5
CE	19.2	24.6
CTC	19.8	24.7
CTC+CE	19.7	24.5
Siamese-PT	**21.5**	**25.5**

Table: Translation performance in BLEU on CoVoST-2

Experimental results

Siamese-PT vs. existing supervised pre-training approaches

Method	**External data**	**avg. BLEU**
FAIRSEQ S2T [Wang et al., 2020a]	-	26.5
ESPNET-ST [Inaguma et al., 2020]	-	25.1
NEURST [Zhao et al., 2021]	-	24.9
XSTNet [Ye et al., 2021]	✓	28.8
Chimera [Han et al., 2021]	✓	27.4
STEMM [Fang et al., 2022]	✓	28.4
ConST [Ye et al., 2022]	✓	29.4
CE pre-training	-	28.8
CTC pre-training	-	29.2
CTC+CE pre-training	-	29.1
Siamese-PT	-	**29.8**

Table: Translation performance in BLEU on MuST-C.

Experimental results

Siamese-PT vs. existing supervised pre-training approaches

Method	External data	avg. BLEU
FAIRSEQ S2T [Wang et al., 2020a]	-	26.5
ESPNET-ST [Inaguma et al., 2020]	-	25.1
NEURST [Zhao et al., 2021]	-	24.9
XSTNet [Ye et al., 2021]	✓	28.8
Chimera [Han et al., 2021]	✓	27.4
STEMM [Fang et al., 2022]	✓	28.4
ConST [Ye et al., 2022]	✓	29.4
CE pre-training	-	28.8
CTC pre-training	-	29.2
CTC+CE pre-training	-	29.1
Siamese-PT	-	**29.8**

Table: Translation performance in BLEU on MuST-C.

Experimental results

Siamese-PT vs. existing supervised pre-training approaches

Method	External data	avg. BLEU
FAIRSEQ S2T [Wang et al., 2020a]	-	26.5
ESPNET-ST [Inaguma et al., 2020]	-	25.1
NEURST [Zhao et al., 2021]	-	24.9
XSTNet [Ye et al., 2021]	✓	28.8
Chimera [Han et al., 2021]	✓	27.4
STEMM [Fang et al., 2022]	✓	28.4
ConST [Ye et al., 2022]	✓	29.4
CE pre-training	-	28.8
CTC pre-training	-	29.2
CTC+CE pre-training	-	29.1
Siamese-PT	-	**29.8**

Table: Translation performance in BLEU on MuST-C.

Experimental results

Siamese-PT vs. existing supervised pre-training approaches

Method	External data	avg. BLEU
FAIRSEQ S2T [Wang et al., 2020a]	-	26.5
ESPNET-ST [Inaguma et al., 2020]	-	25.1
NEURST [Zhao et al., 2021]	-	24.9
XSTNet [Ye et al., 2021]	✓	28.8
Chimera [Han et al., 2021]	✓	27.4
STEMM [Fang et al., 2022]	✓	28.4
ConST [Ye et al., 2022]	✓	29.4
CE pre-training	-	28.8
CTC pre-training	-	29.2
CTC+CE pre-training	-	29.1
Siamese-PT	-	**29.8**

Table: Translation performance in BLEU on MuST-C.

Experimental results

Application to multi-task learning

ASR pre-training method	**En-De**
CE (reported in Tang et al. [2021])	26.74
CE (reproduced)	26.78
CTC	27.04
CTC+CE	26.69
Siamese-PT	**27.20**

Table: Results of the multi-task learning system of Tang et al. [2021], using different ASR pre-training methods.

Experimental results

Application to multi-task learning

ASR pre-training method	**En-De**
CE (reported in Tang et al. [2021])	26.74
CE (reproduced)	26.78
CTC	27.04
CTC+CE	26.69
Siamese-PT	**27.20**

Table: Results of the multi-task learning system of Tang et al. [2021], using different ASR pre-training methods.

Experimental results

Application to multi-task learning

ASR pre-training method	**En-De**
CE (reported in Tang et al. [2021])	26.74
CE (reproduced)	26.78
CTC	27.04
CTC+CE	26.69
Siamese-PT	**27.20**

Table: Results of the multi-task learning system of Tang et al. [2021], using different ASR pre-training methods.

Contribution 3: Conclusion

We have shown that:

- Encoder trained with CTC is stronger than the one trained with encoder-decoder-CE.
- Our Siamese pre-training helps reduce modality gap without any changes in the ST model.
- Optimal transport is very effective for learning to align sequences of features from different modalities.

Thesis conclusion & discussion

Thesis conclusion & discussion

Three major contributions, spanning two primary ST research areas: **model architectures** and **training techniques**.

Thesis conclusion & discussion

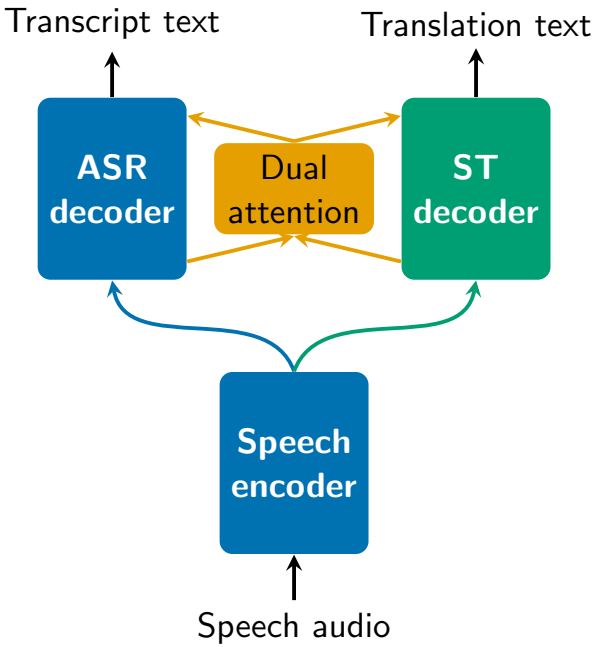
Three major contributions, spanning two primary ST research areas: **model architectures** and **training techniques**.

Proposed methods are general and flexible:

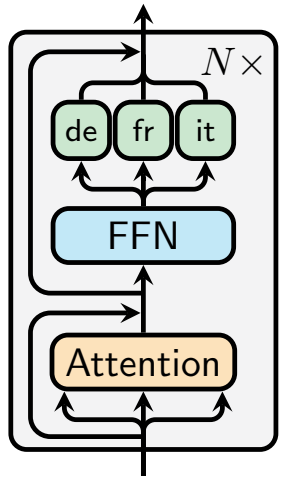
- *Dual-decoder Transformer* can be applied to more general problems where two tasks can help each other
- *Siamese network with Optimal Transport* can be used to learn to align two sequences of different modalities
- *Adapter tuning* can be used to connect existing pre-trained models from different modalities.

Future work: Three ideas in a unified framework

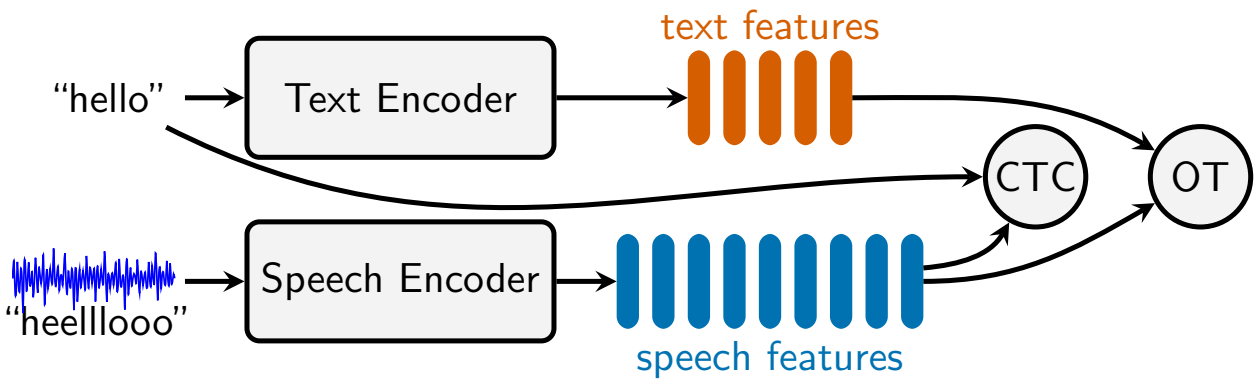
Future work: Three ideas in a unified framework



Dual-decoder Transformer

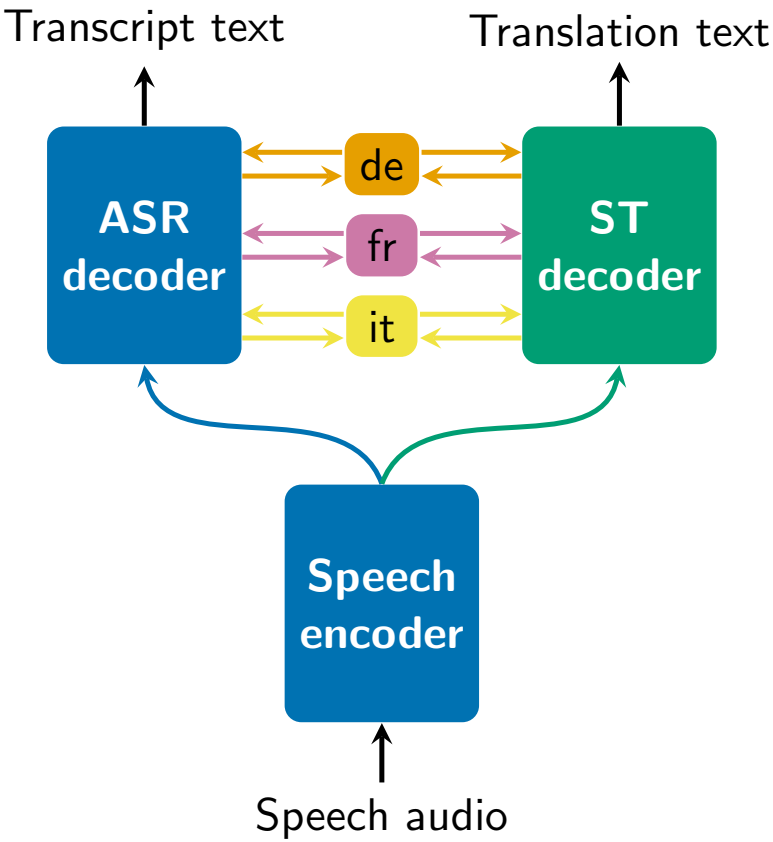


Adapters



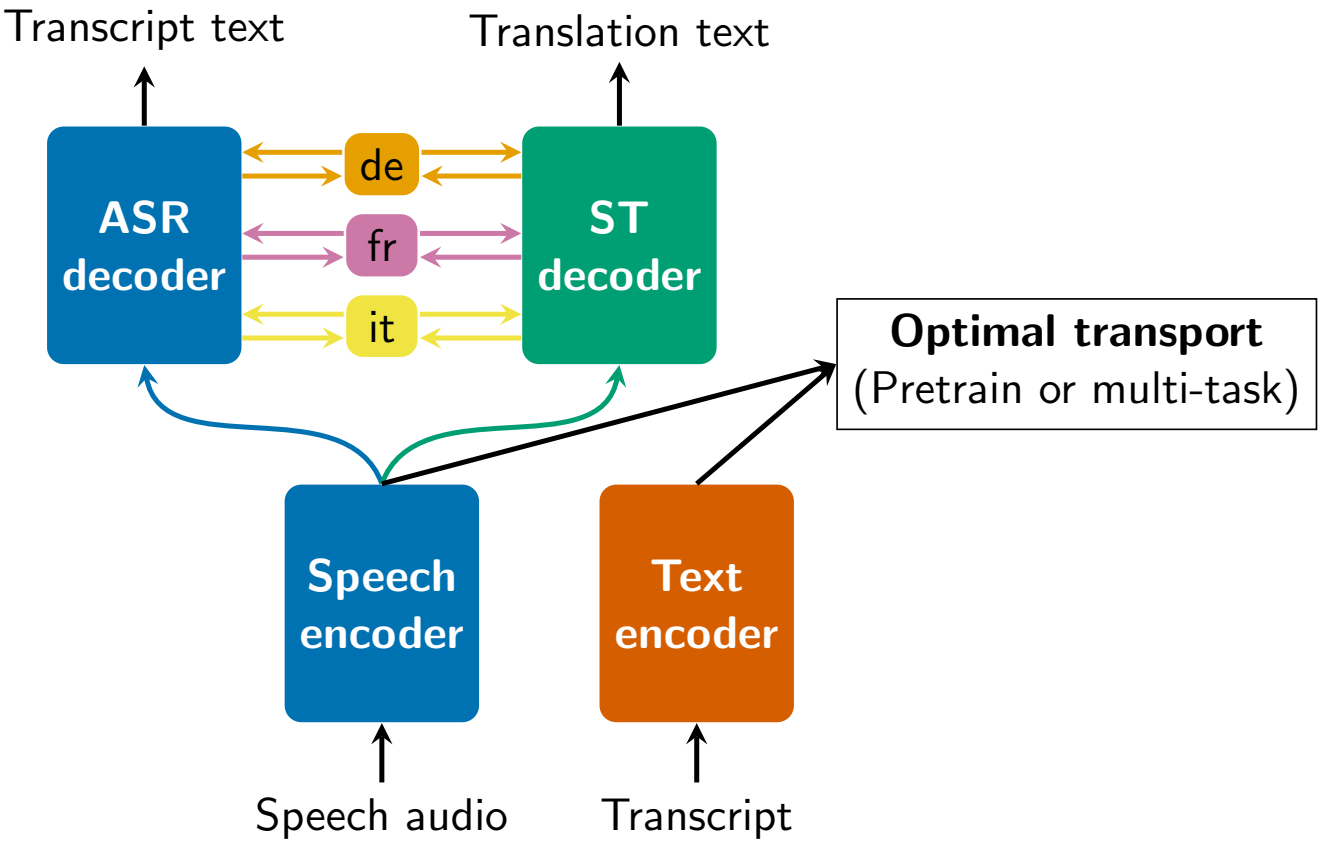
Optimal transport

Future work: Three ideas in a unified framework



This is work in progress.

Future work: Three ideas in a unified framework



This is work in progress.

Thank you for your attention!

Thank you for your attention!

Thank you for your attention!

Thank you for your attention!

Thank you for your attention!

Thank you for your attention!

Thank you for your attention!

Thank you for your attention!

Thank you for your attention!

Thank you for your attention!

Thank you for your attention!

Thank you for your attention!