DEVIEW 2021

Efficient Multilingual Neural Machine Translation

Alexandre Berard, Laurent Besacier, Vassilina Nikoulina NAVER LABS Europe

Other Contributors:

Stephane Clinchant, Matthias Galle NAVER LABS Europe

Kweonwoo Jung, Dain Lee NAVER Corp. (Papago)

Asa Cooper Stickland University of Edinburgh

Ahmet Ustun University of Groningen

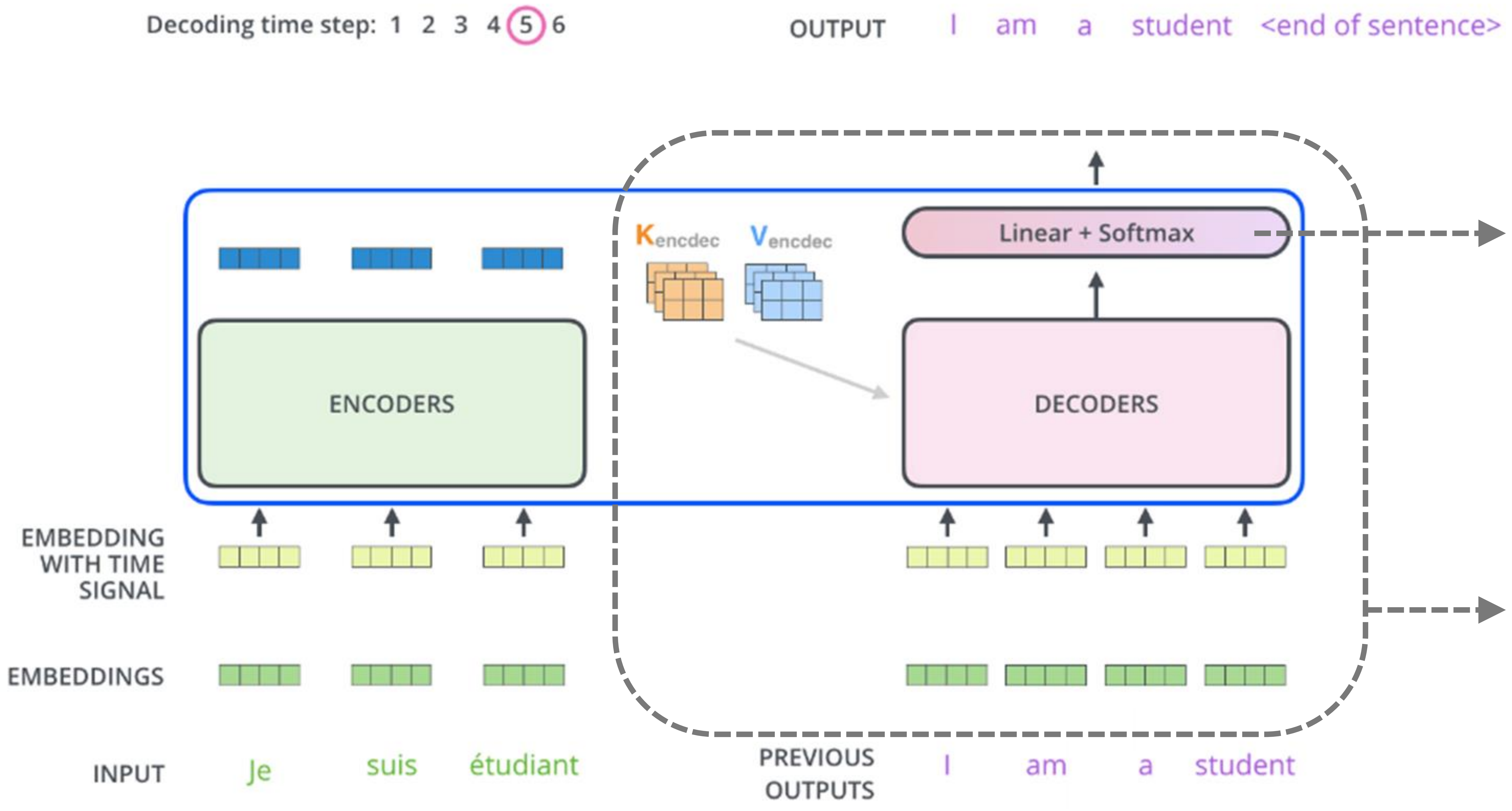
NAVER LABS Europe

papago

CONTENTS

1.Introduction to multilingual neural machine translation (MNMT)

1.1 Encoder-decoder



~15% of parameters

Grows with vocabulary size

Most costly part

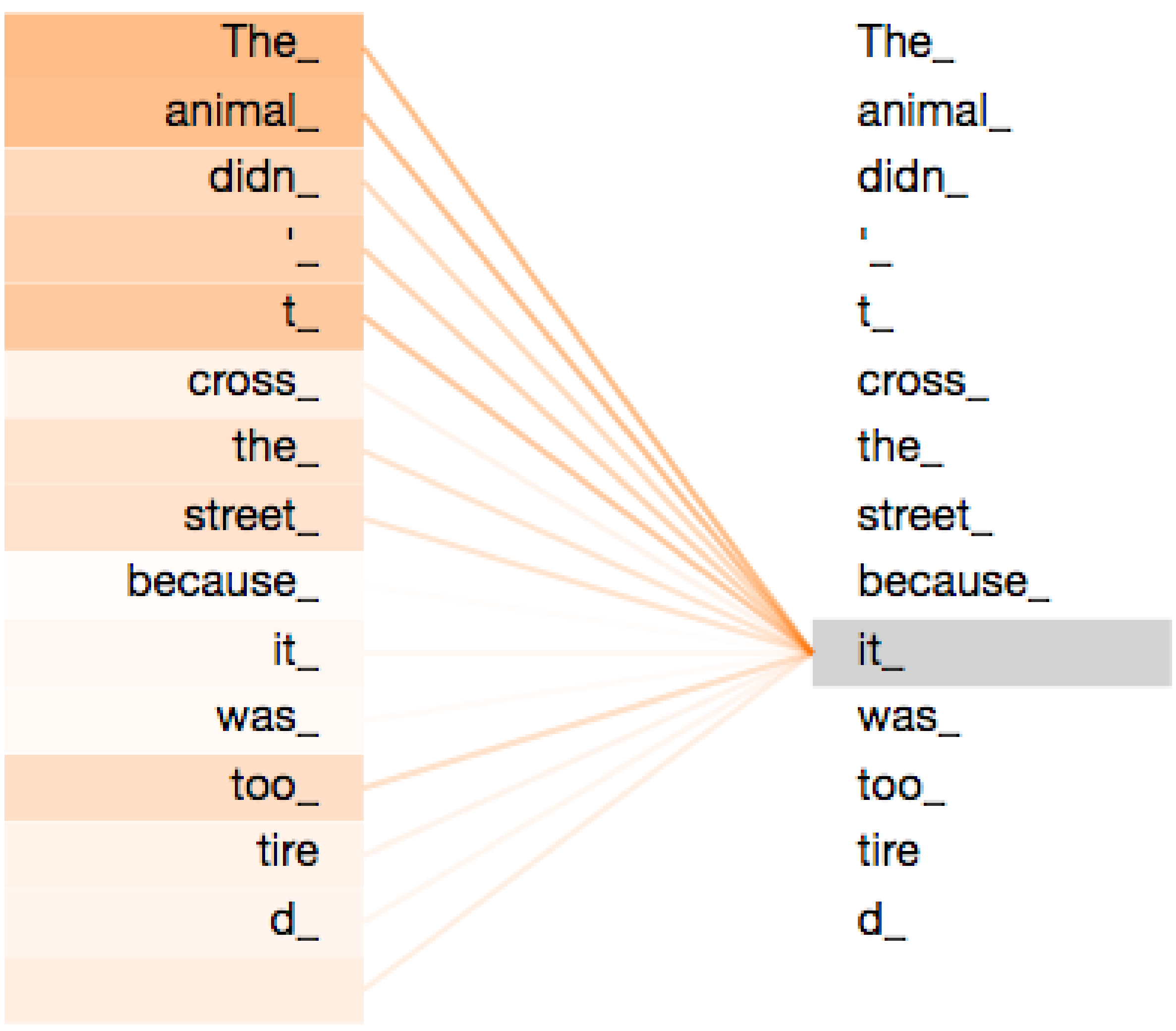
Called for each new token

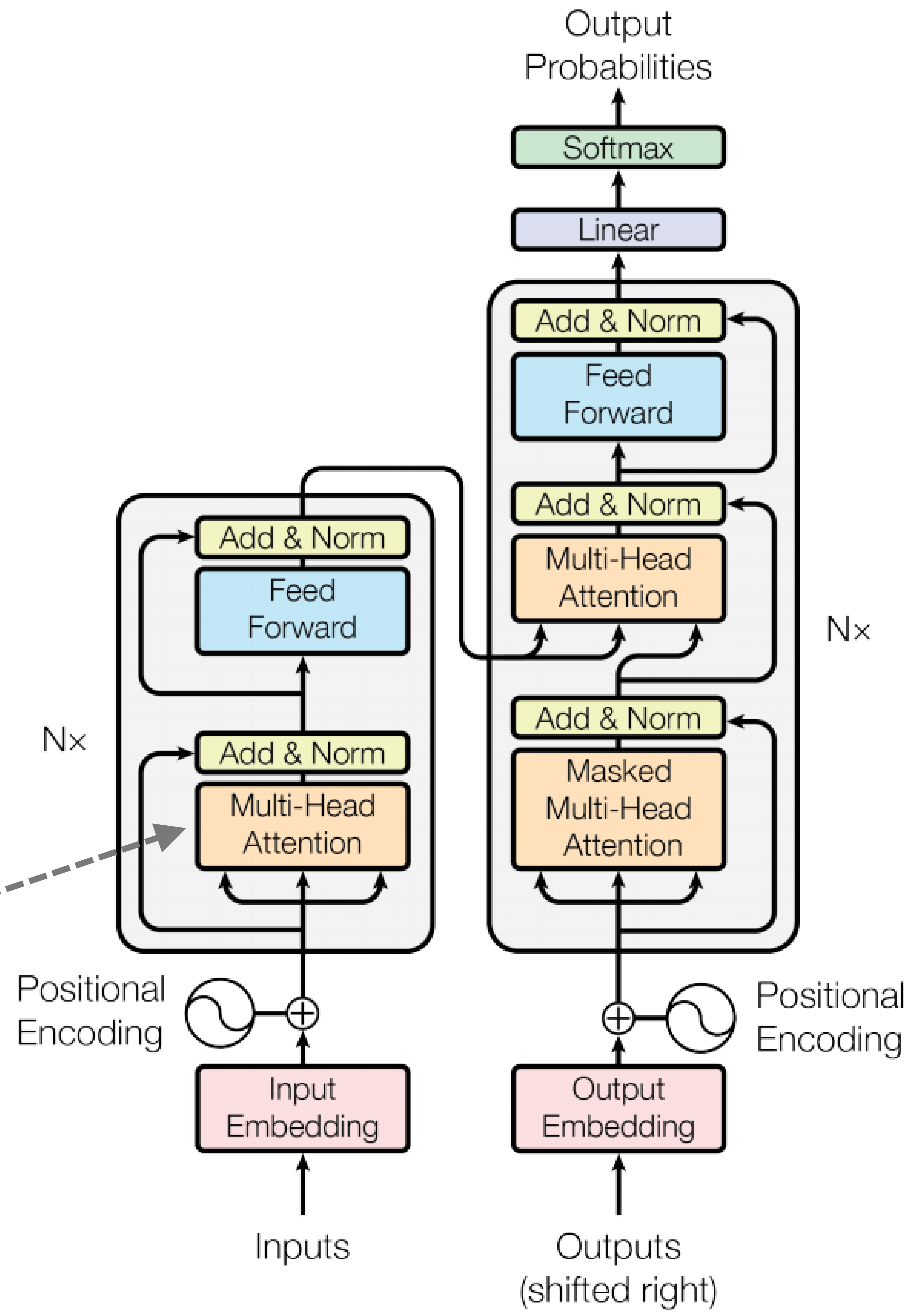
From *https://jalammar.github.io/illustrated-transformer*

1.1 Transformer

State-of-the-art model in NMT

Vaswani et al. (2017)





1.2 Pre-processing

Tokenize text sequences into sequences of *wordpieces*

고맙습니다. ➔ _고 | 맙 | 습니다 | . ➔ 147 | 1809 | 13 | 1009

Thank you. ➔ _Thank | _you | . ➔ 663 | 54 | 1029

Typically with the *Byte Pair Encoding* algorithm

1.2 Pre-processing

Tokenize text sequences into sequences of *wordpieces*

고맙습니다. → ▁고 | 맙 | 습니다 | . → 147 | 1809 | 13 | 1009

Thank you. → ▁Thank | ▁you | . → 663 | 54 | 1029

Typically with the *Byte Pair Encoding* algorithm

Vocab ID	Wordpiece
13	습니다
145	▁안
147	▁고
1009	.
1017	하
1809	맙
1872	녕

Korean vocabulary

Vocab ID	Wordpiece
28	▁m
38	or
54	▁you
346	▁G
488	ood
663	▁Thank
1029	.

English vocabulary

1.3 Machine translation evaluation

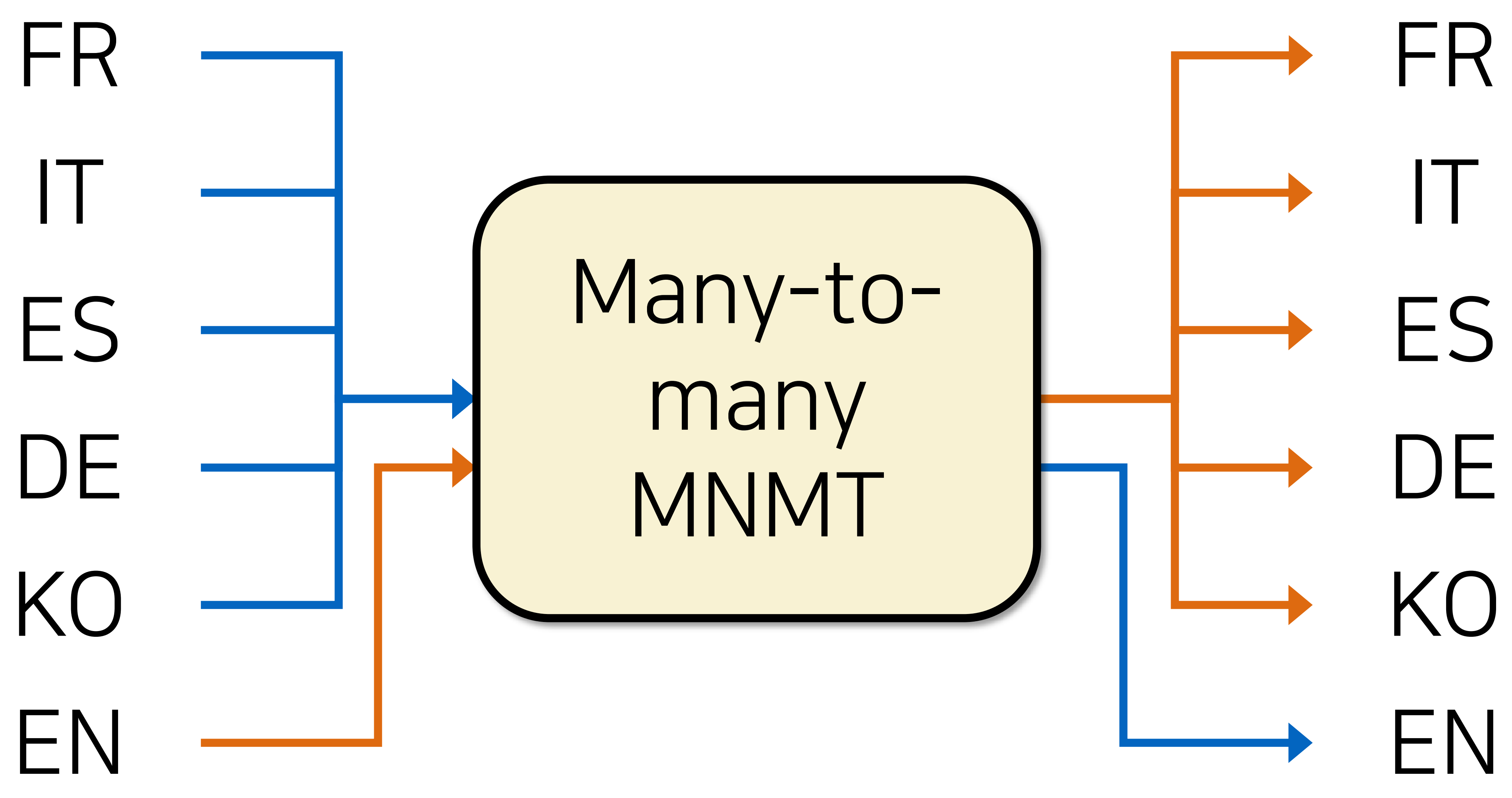
DEVIEW 2021

BLEU (Papineni et al., 2002)

- Default evaluation metric in MT
- Based on *precision* of matched N-grams
- Not perfect: complementary evaluation is often helpful

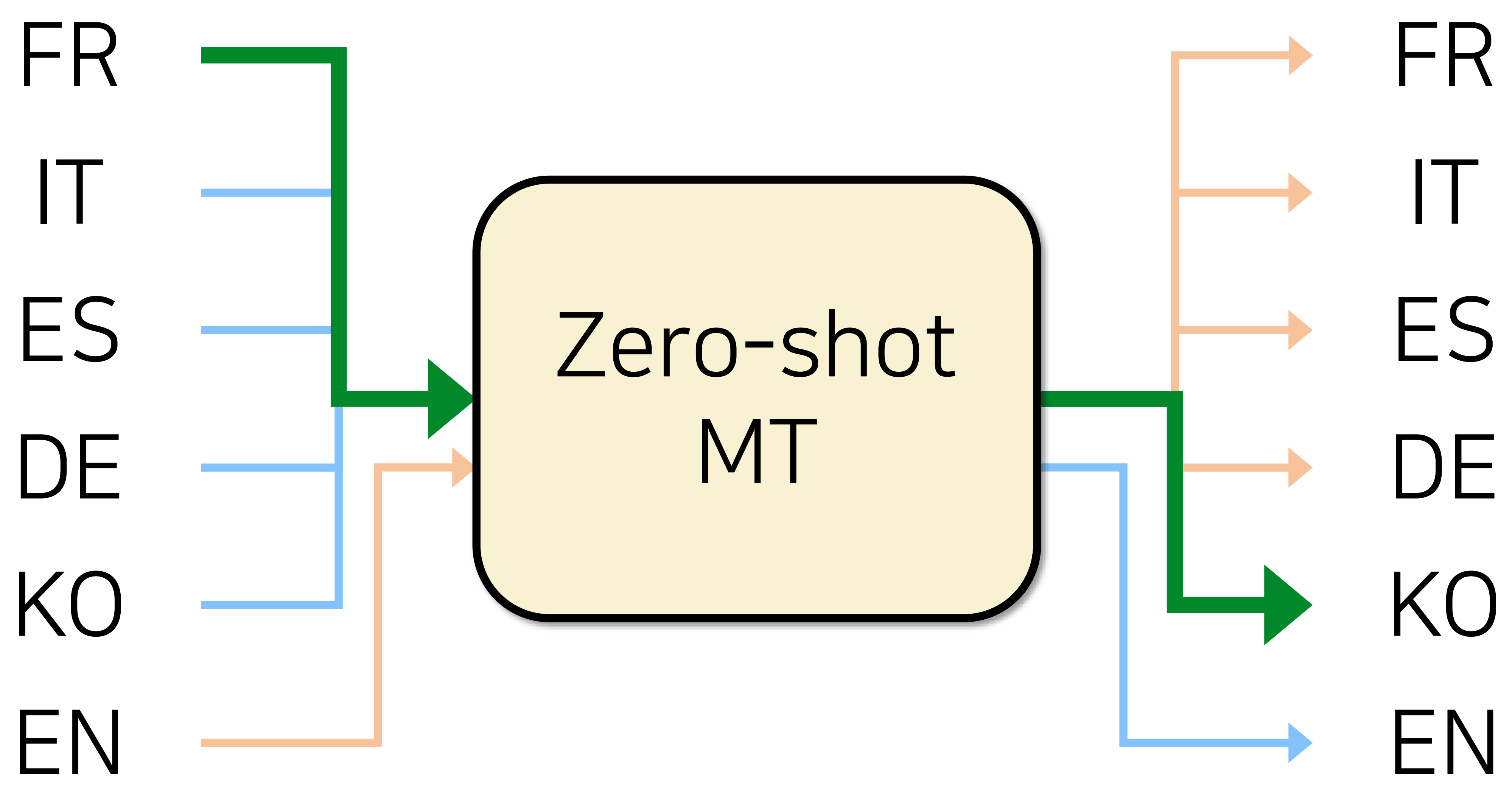
枪手被警方击毙。	(Source Original)	
the gunman was shot to death by the police .	(Reference Translation)	
the gunman was police kill .	#1	
wounded police jaya of	#2	
the gunman was shot dead by the police .	#3	
the gunman arrested by police kill .	#4	
the gunmen were killed .	#5	
the gunman was shot to death by the police .	#6	*Best BLEU Score*

1.4 Multilingual NMT (MNMT)



- One single model for multiple source and target languages
- Shared vocabulary and embeddings
- Choose target language with source-side tag: <2FR>, <2IT>, etc.
- Often trained on *English-centric* data: FR-EN, EN-KO, but no FR-KO

1.4 Multilingual NMT (MNMT)



Zero-shot translation

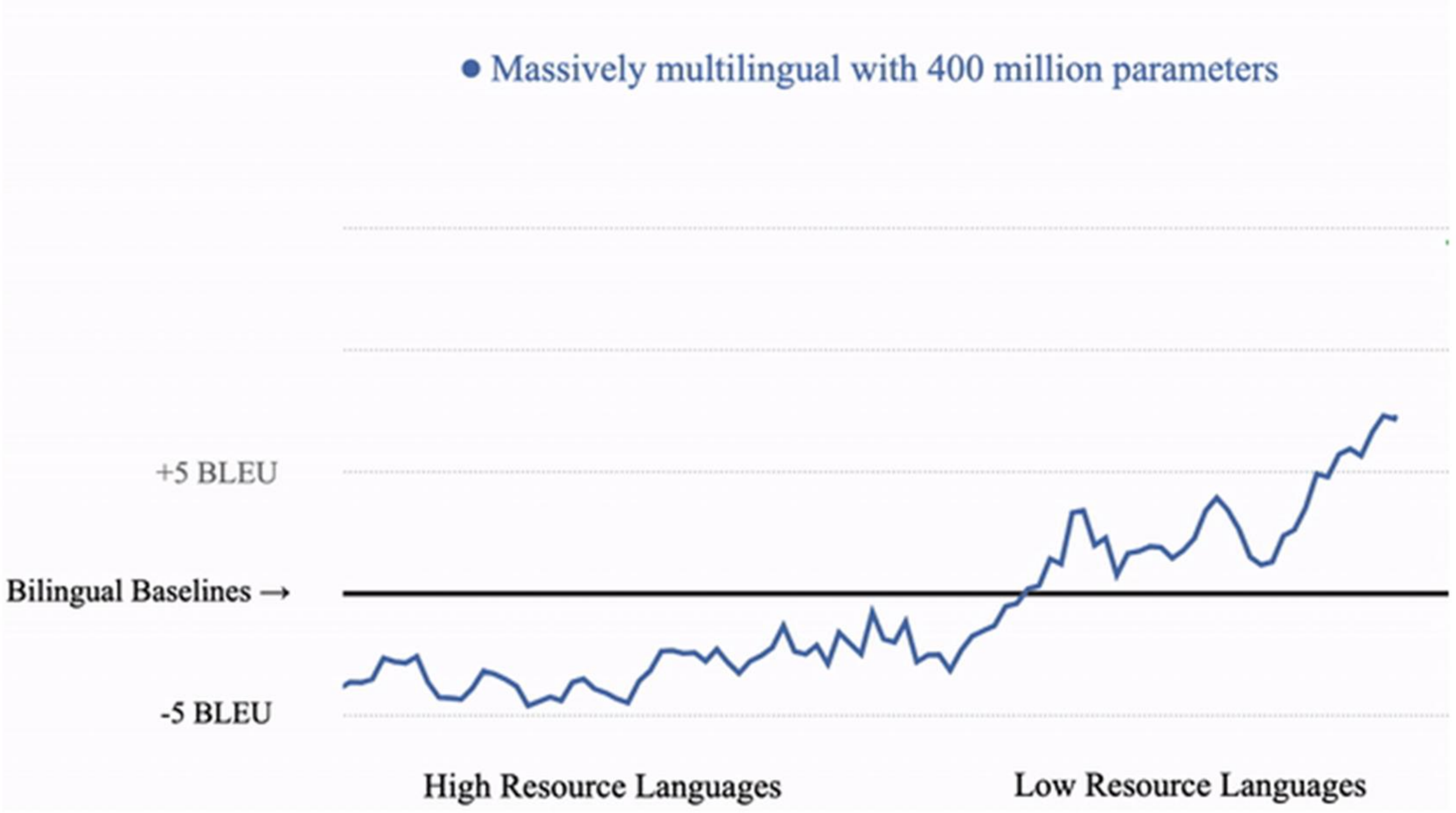
Translate in a language pair that was not seen at training
But whose source and target language are known

<2KO> | _Merci | _beau | coup | . ➔ _고 | 맙 | 습니다 | .

1.5 Challenges of MNMT

Multilingual NMT is convenient in production

- single model for all language pairs
- knowledge transfer



Translation quality improvement of a single massively multilingual model as we increase the capacity (number of parameters) compared to 103 individual bilingual baselines.

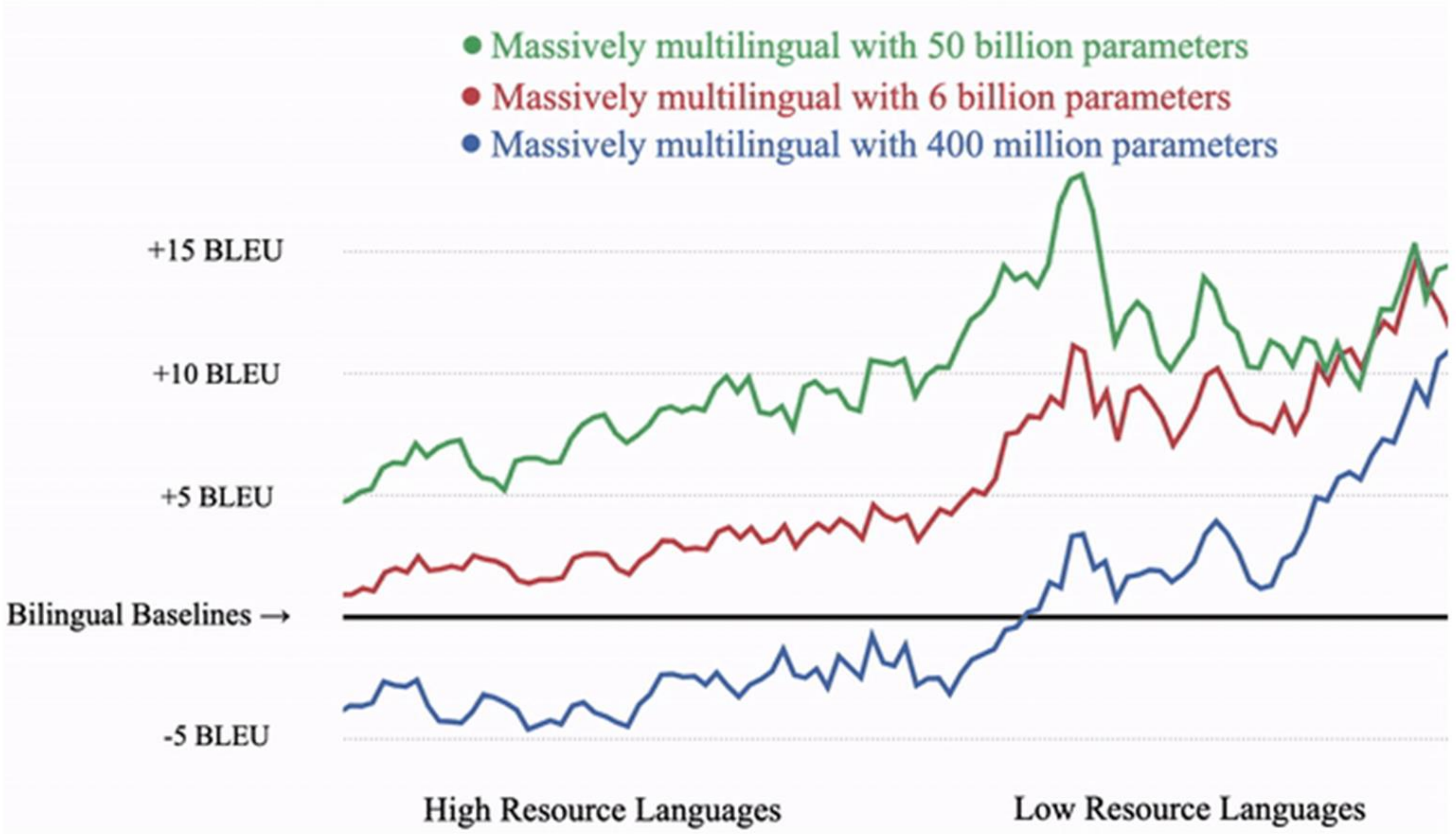
Arivazhagan et al. (2019)

Google's Massively MNMT paper

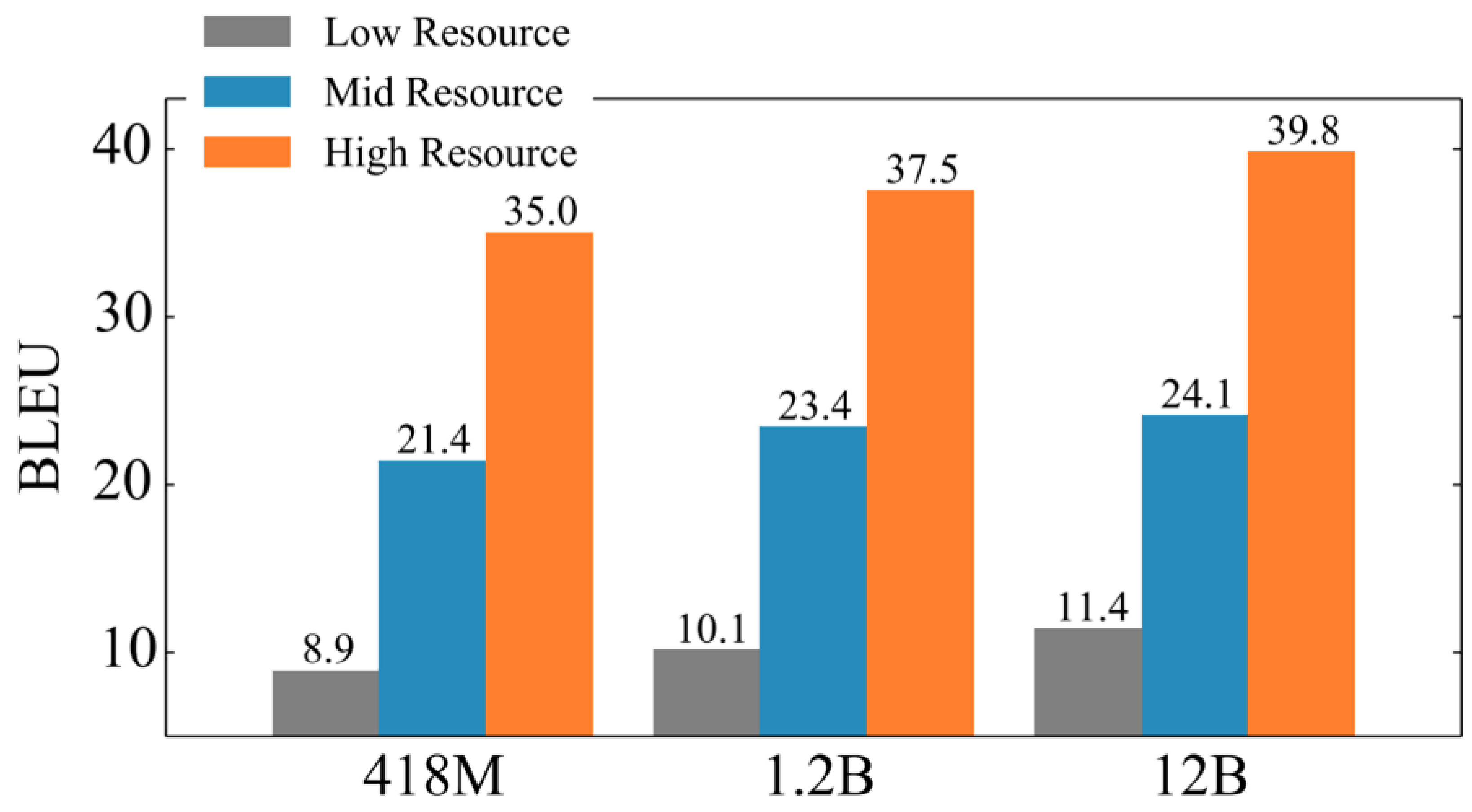
1.5 Challenges of MNMT

Multilingual NMT is convenient in production, but it requires bigger models

- slower at inference
- costly to train



Translation quality improvement of a single massively multilingual model as we increase the capacity (number of parameters) compared to 103 individual bilingual baselines.



Fan et al. (2020)

Facebook AI's M2M-100

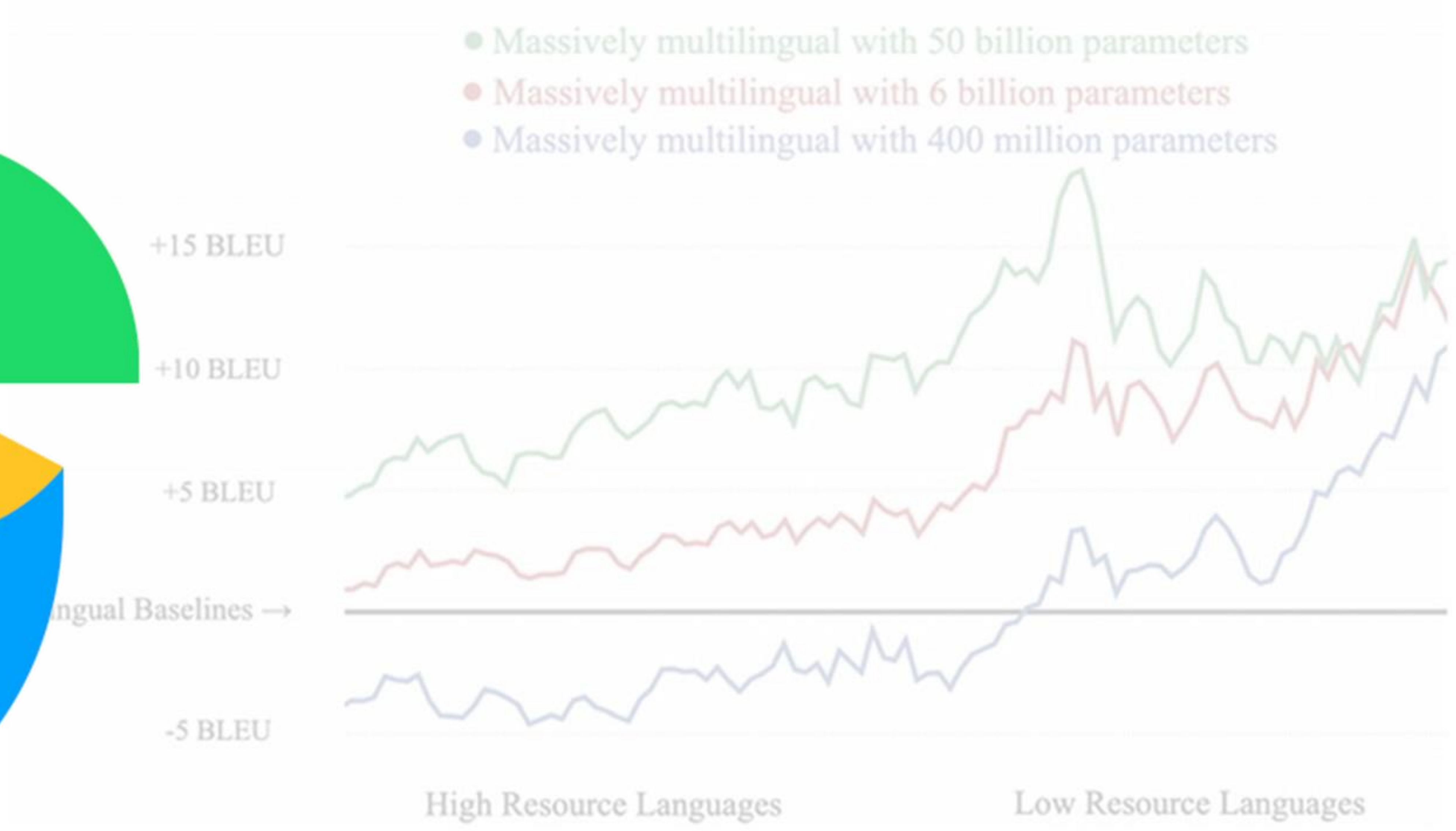
Arivazhagan et al. (2019)

Google's Massively MNMT paper

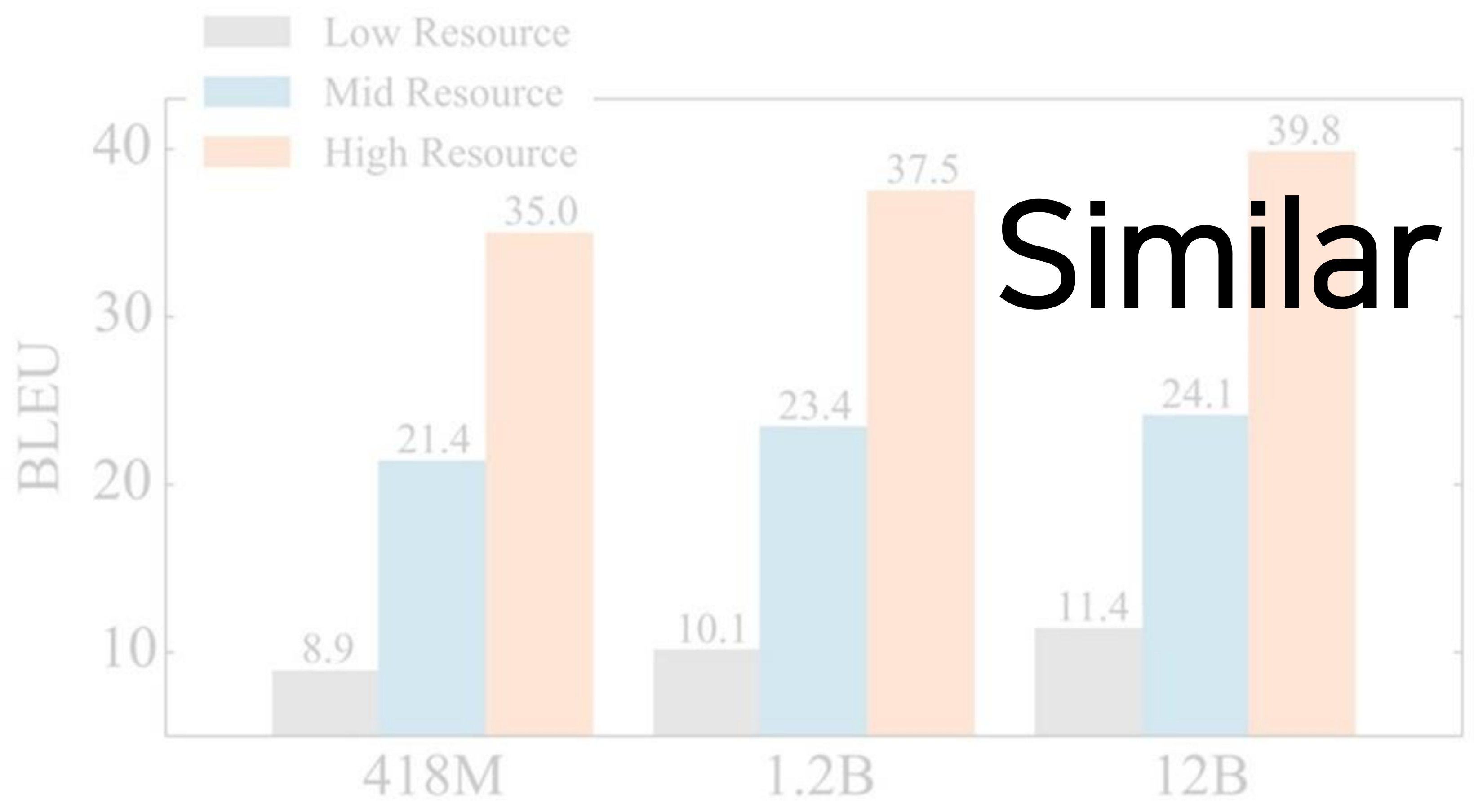
1.5 Challenges of MNMT

Multilingual NMT is convenient in production, but it requires bigger models

- slower at inference
- costly to train



Translation quality improvement of a single massively multilingual model as we increase the capacity (number of parameters) compared to 103 individual bilingual baselines.



Similar findings at NAVER Papago

Fan et al. (2020)

Facebook AI's M2M-100

Arivazhagan et al. (2019)

Google's Massively MNMT paper

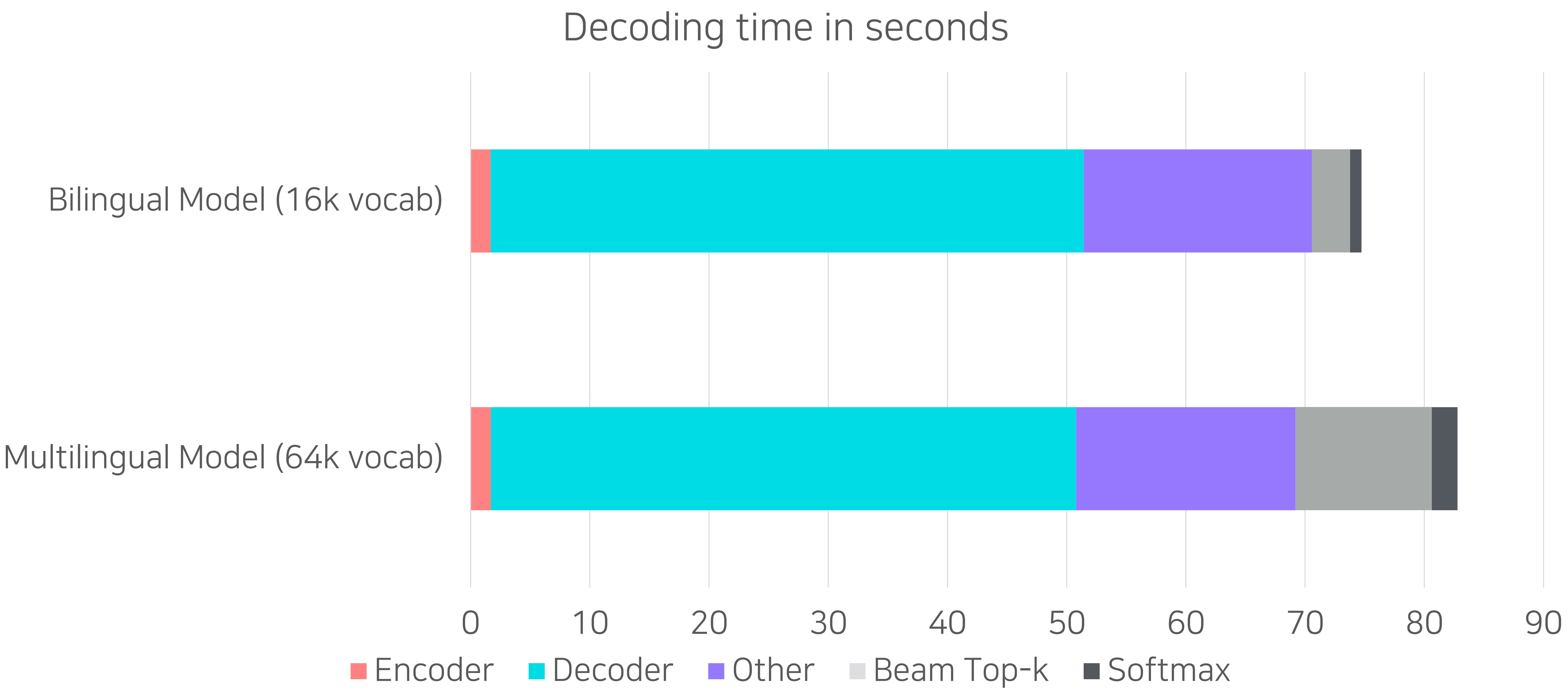
2. Speeding up inference

Efficient Inference for Multilingual Neural Machine Translation

A. Berard, D. Lee, S. Clinchant, K. Jung and V. Nikoulina

EMNLP 2021

2.1 Introduction



- Most time is spent in the *decoder*
- Encoder time is negligible
- *Softmax* and *beam search* times are higher for the multilingual model

2.2 Techniques

Faster decoder

- Deep encoder / shallow decoder(s)
- Hybrid model with shallow RNN decoder

2.2 Techniques

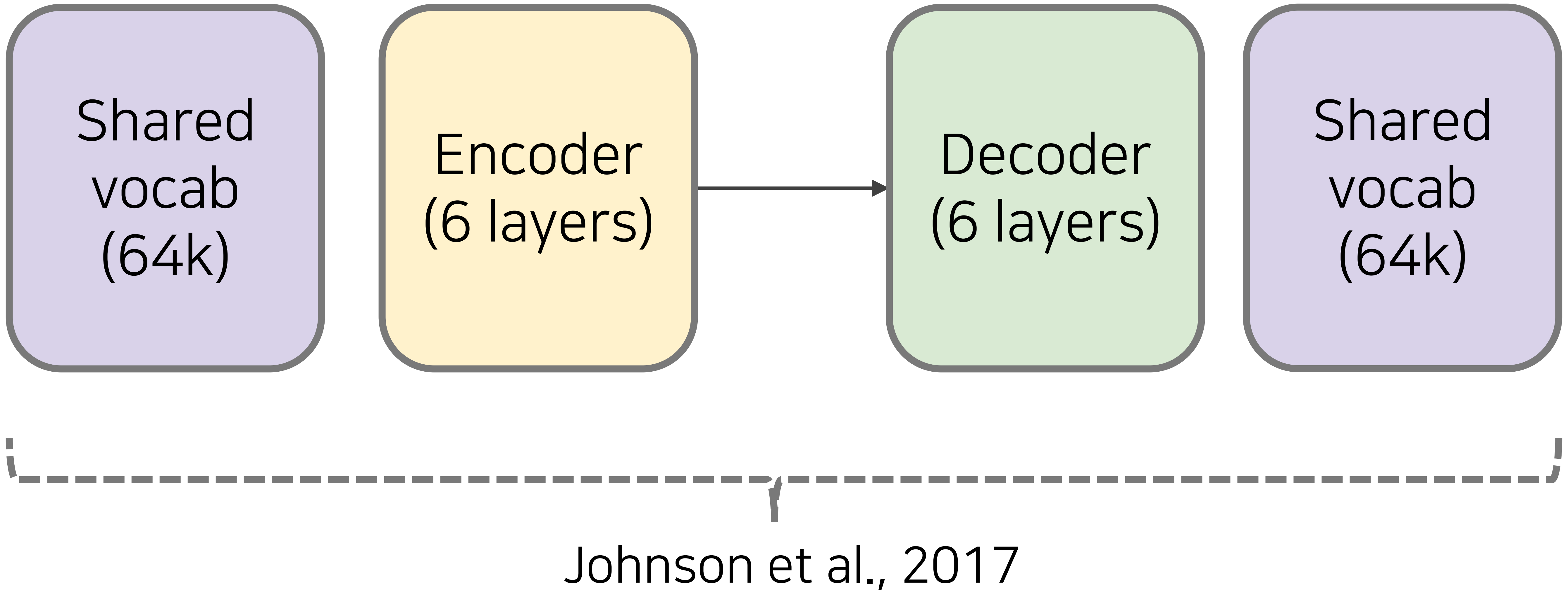
Faster decoder

- Deep encoder / shallow decoder(s)
- Hybrid model with shallow RNN decoder

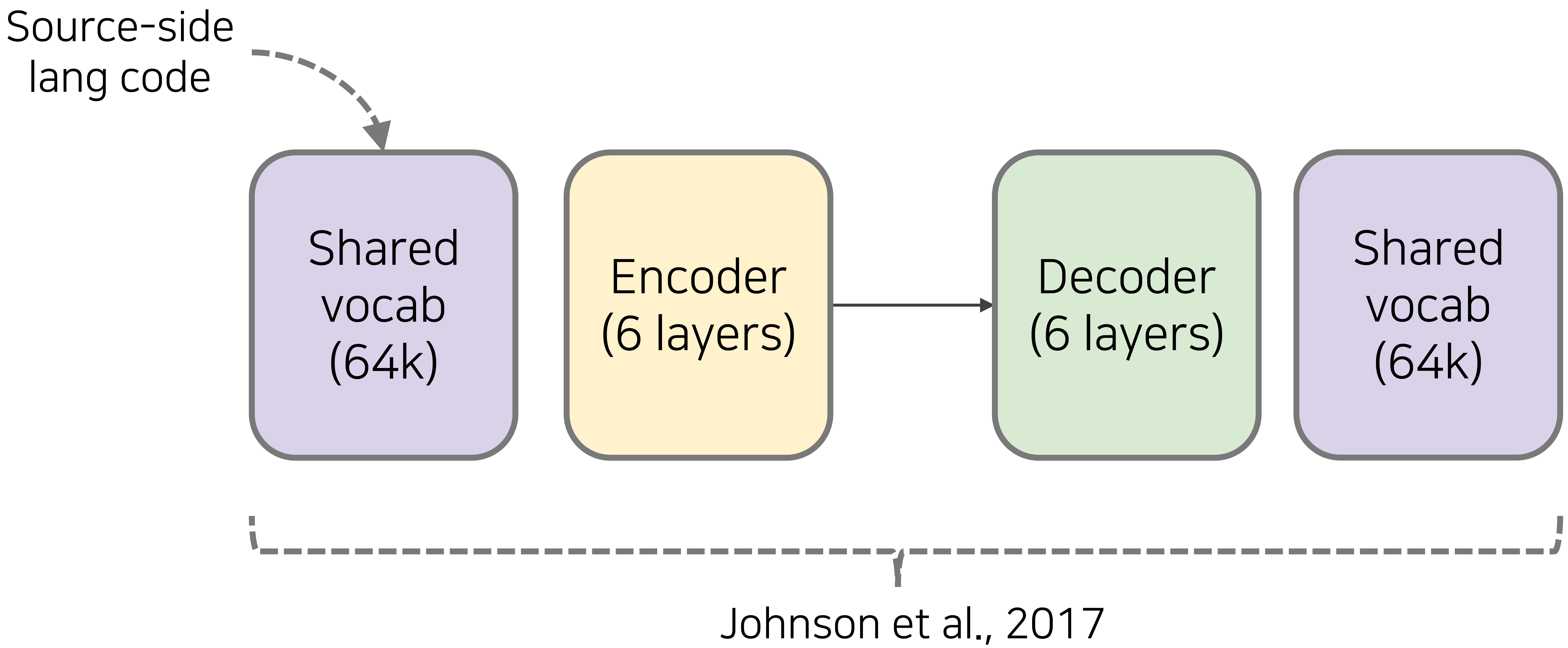
Reducing softmax / beam search cost

- Language-specific vocabulary filtering

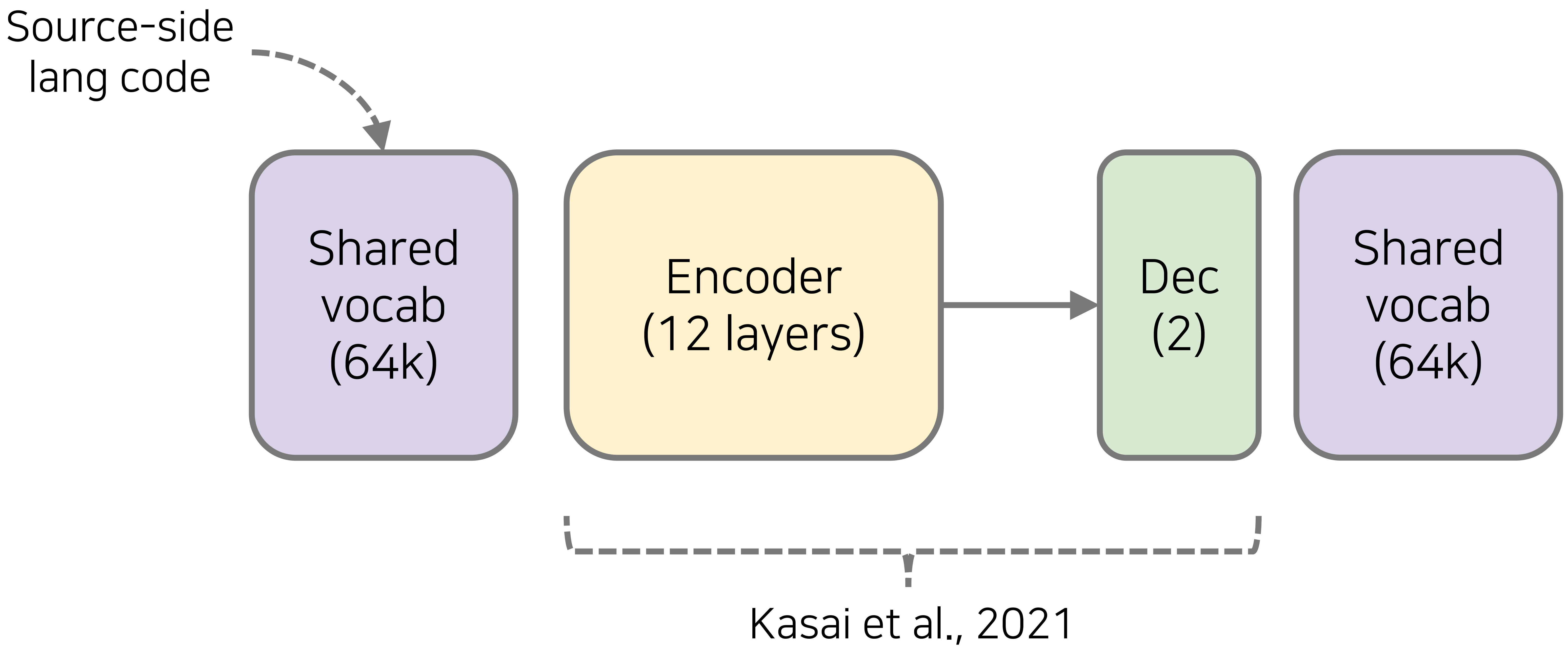
2.2 Techniques: baseline model



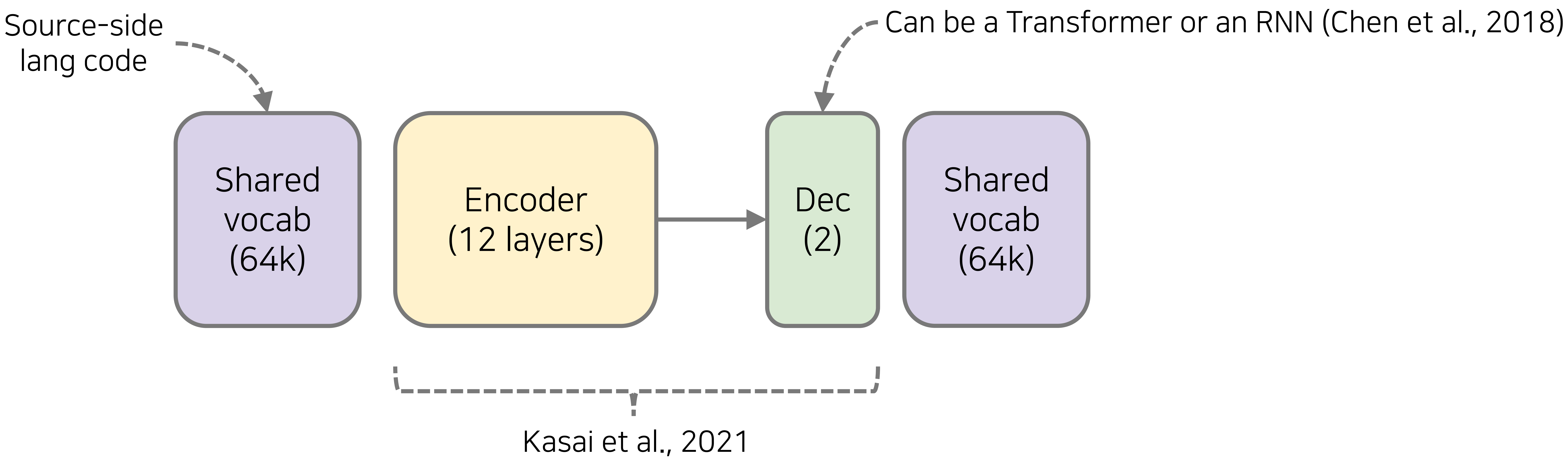
2.2 Techniques: baseline model



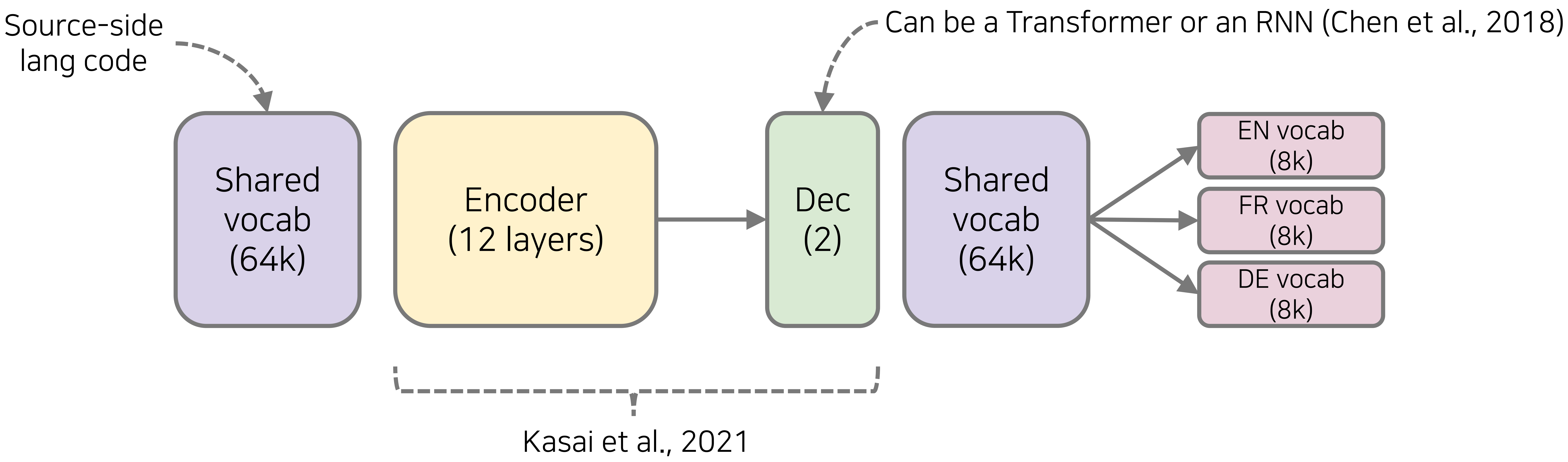
2.2 Techniques: fast MNMT model



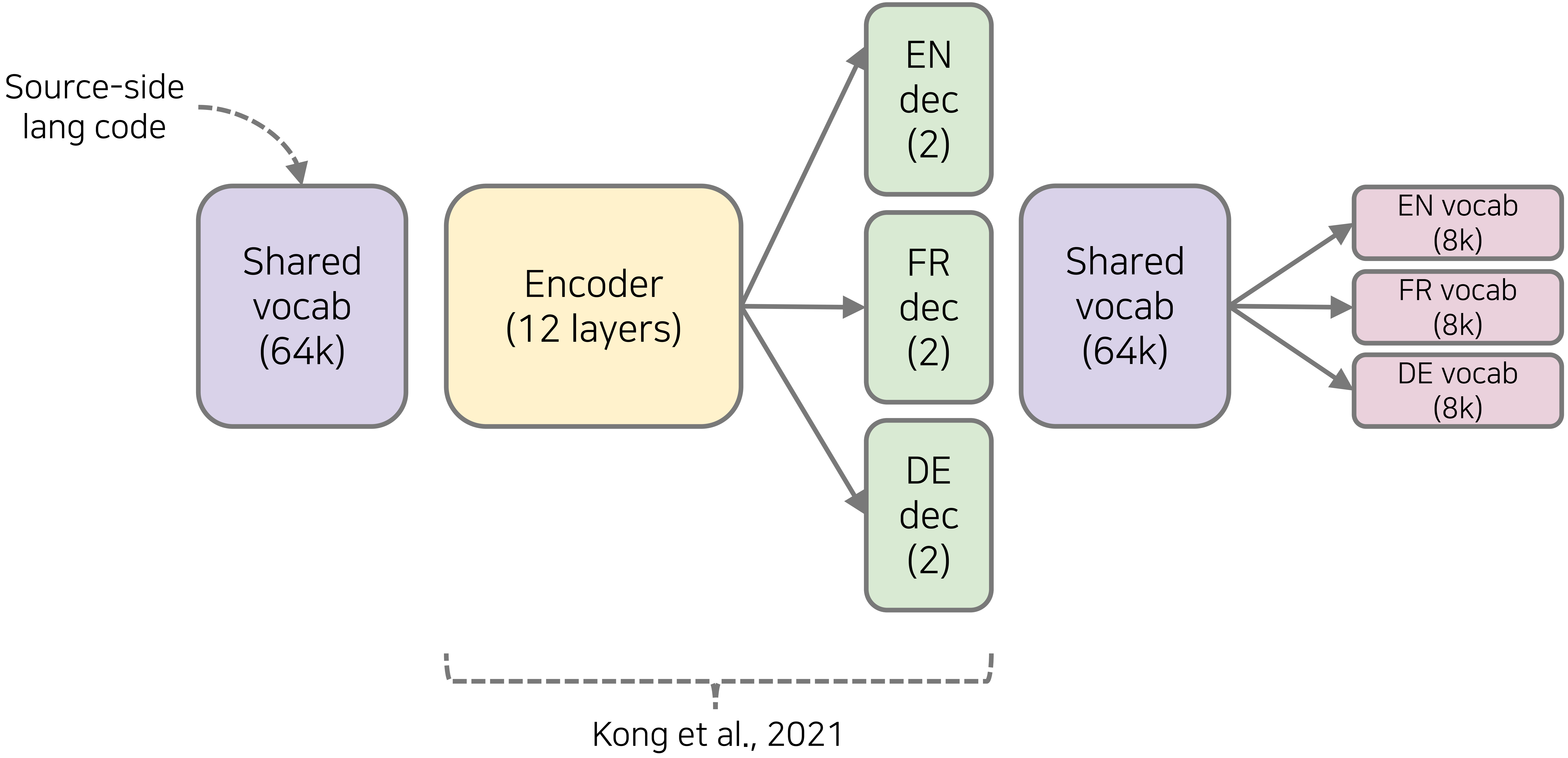
2.2 Techniques: fast MNMT model



2.2 Techniques: fast MNMT model



2.2 Techniques: multi-decoder model



2.2 Techniques: language-specific vocabulary filtering

1. Tokenize German training data with shared BPE
2. Count token frequencies
3. Build German-specific vocab with top 8k tokens (subset of shared vocab)

2.2 Techniques: language-specific vocabulary filtering

1. Tokenize German training data with shared BPE
2. Count token frequencies
3. Build German-specific vocab with top 8k tokens (subset of shared vocab)

Test-time filtering:

- Filter target embedding matrix to only keep German tokens

2.2 Techniques: language-specific vocabulary filtering

1. Tokenize German training data with shared BPE
2. Count token frequencies
3. Build German-specific vocab with top 8k tokens (subset of shared vocab)

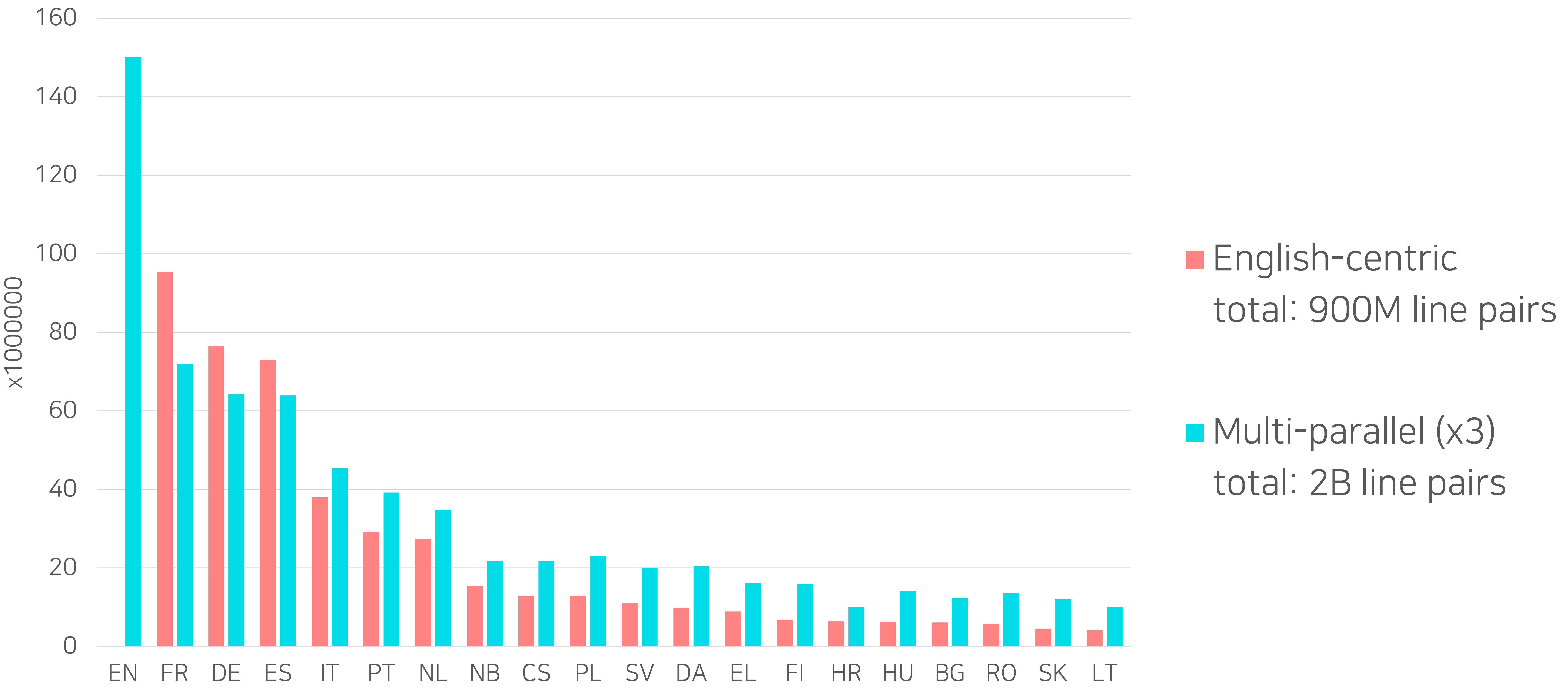
Test-time filtering:

- Filter target embedding matrix to only keep German tokens

Train-time filtering:

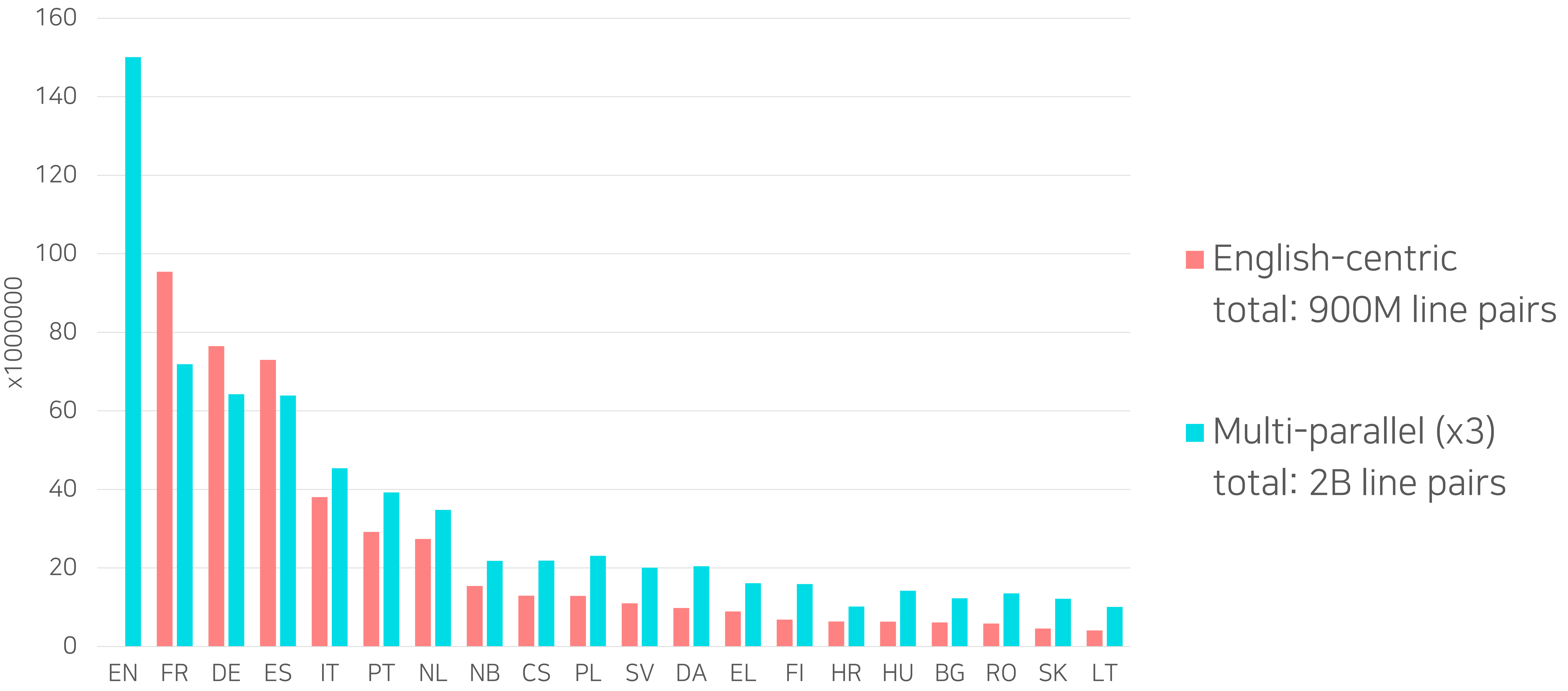
- Continue training with shared vocab, but force target tokenization to only generate German tokens

2.3 Experiments: ParaCrawl Top 20



6 language families, 3 scripts

2.3 Experiments: ParaCrawl Top 20



6 language families, 3 scripts

Test set: FLORES (in all 380 directions)

2.3 Experiments: training tricks

- 2-stage training: English-centric → multi-parallel

2.3 Experiments: training tricks

- 2-stage training: English-centric → multi-parallel
- Initialize 12-2 model with 6-6 model (only for TED Talks experiments)

2.3 Experiments: training tricks

- 2-stage training: English-centric → multi-parallel
- Initialize 12-2 model with 6-6 model (only for TED Talks experiments)
- Initialize Hybrid model with Transformer

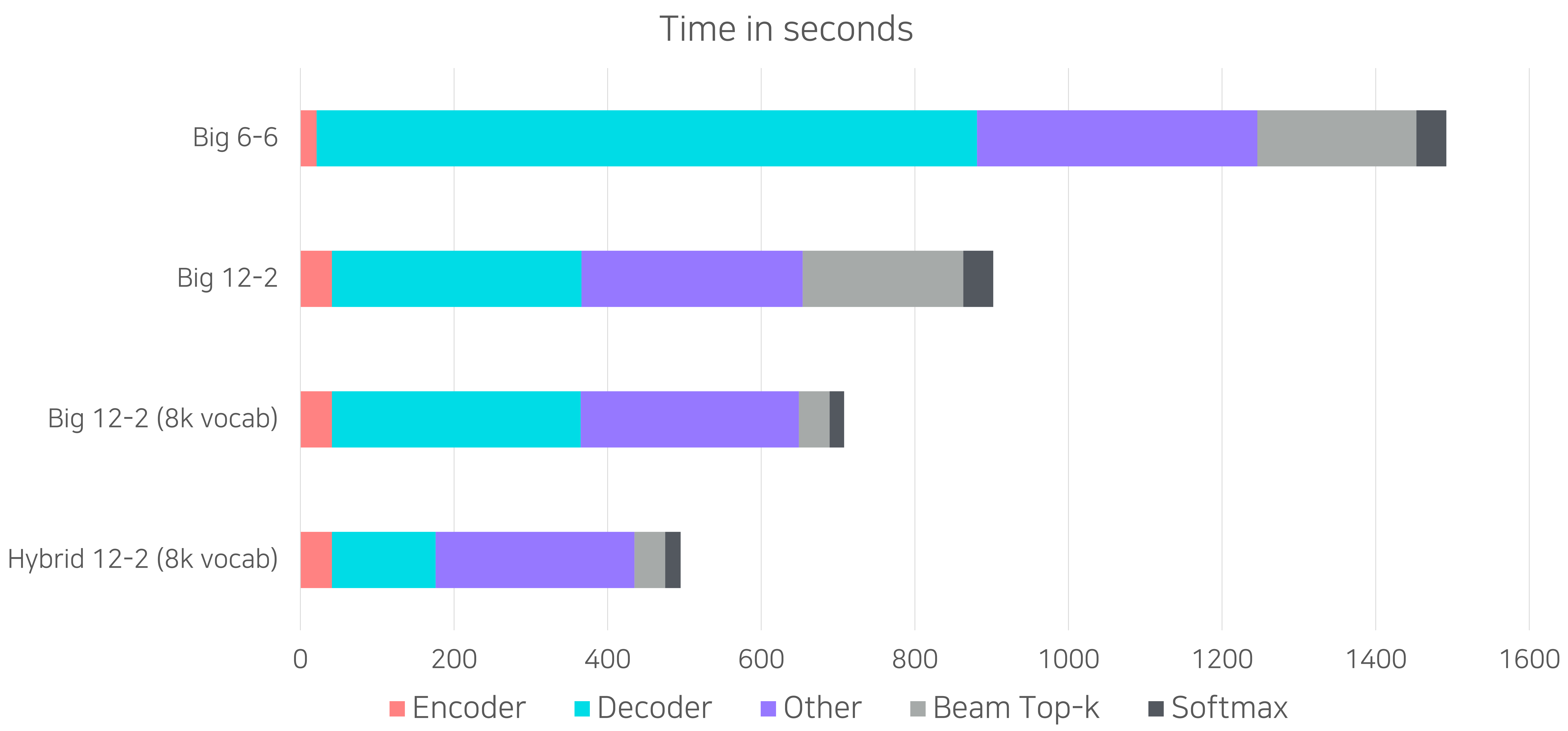
2.3 Experiments: training tricks

- 2-stage training: English-centric → multi-parallel
- Initialize 12-2 model with 6-6 model (only for TED Talks experiments)
- Initialize Hybrid model with Transformer
- Initialize multi-decoder model with single-decoder model

2.3 Experiments: training tricks

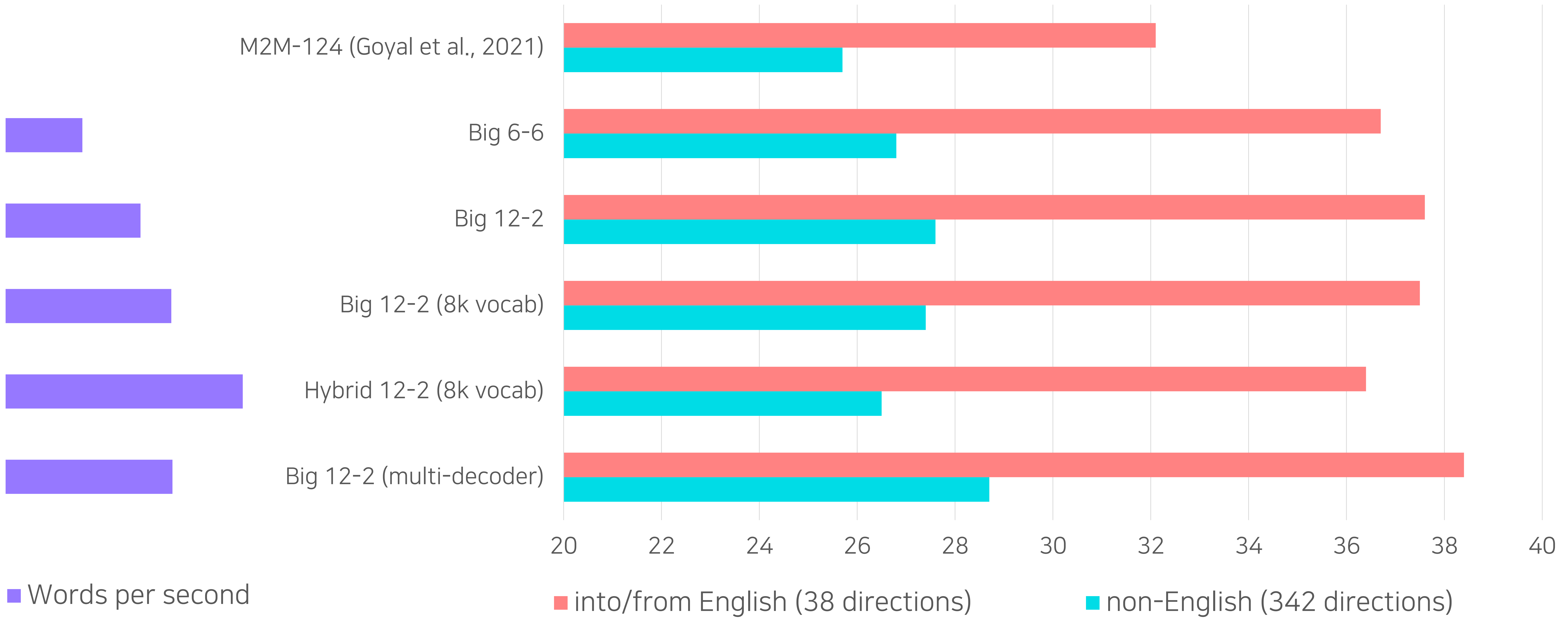
- 2-stage training: English-centric → multi-parallel
- Initialize 12-2 model with 6-6 model (only for TED Talks experiments)
- Initialize Hybrid model with Transformer
- Initialize multi-decoder model with single-decoder model
- Language codes must be on the source side
    - For zero-shot translation
    - For the 12-2 architectures

2.4 Results: inference time

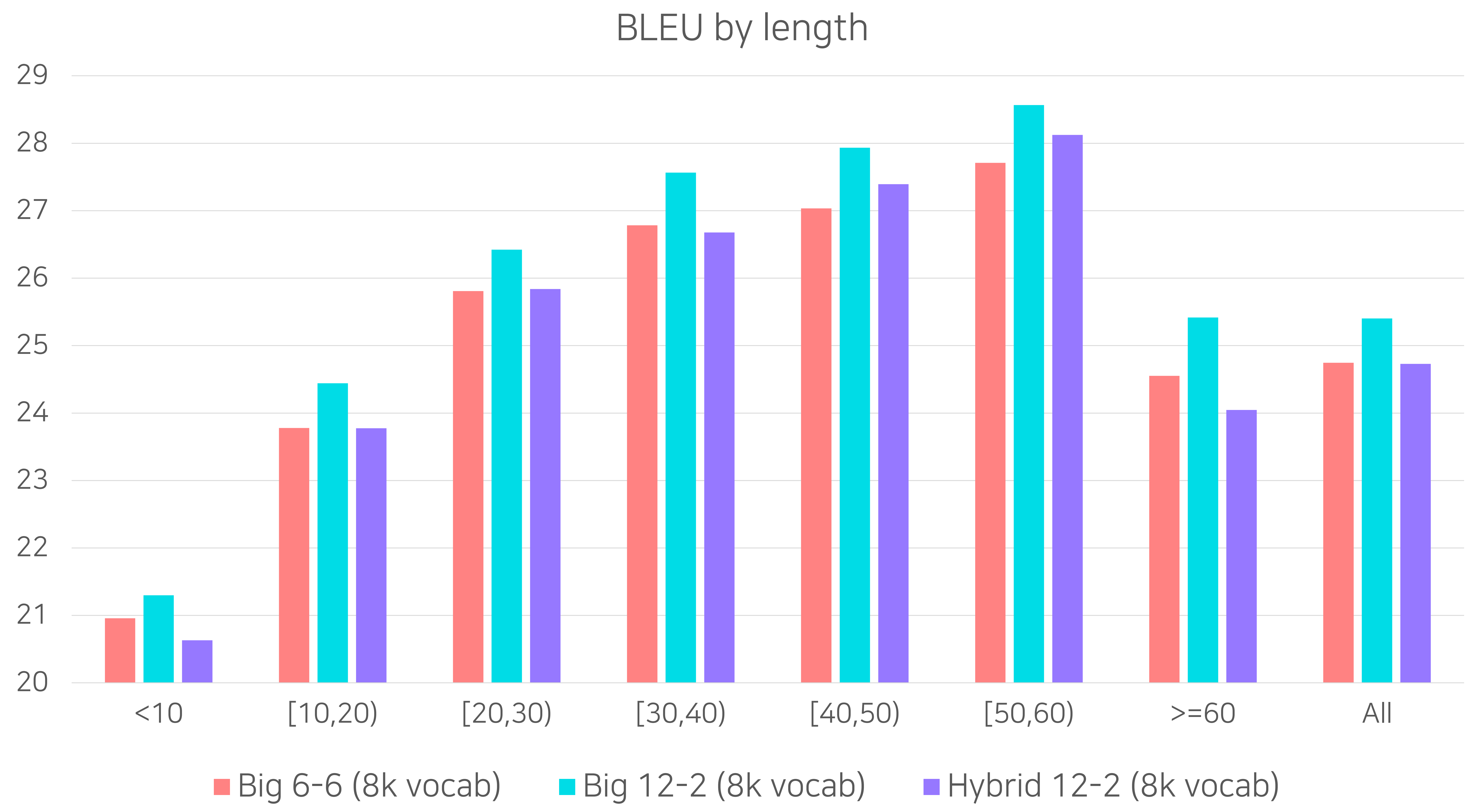


Beam size: 5 Batch size: 64 lines Device: V100x1

2.4 Results: BLEU vs speed



2.4 Results: robustness to length



2.4 Results: robustness to noise

Model	BLEU consistency (UNK)	BLEU consistency (Char)
Big 6-6	73.3	54.2
Big 12-2	**76.4**	**56.1**
Big 12-2 (8k vocab)	73.7	55.5
Hybrid 12-2 (8k vocab)	75.0	55.3

- UNK: unknown symbol inserted at the beginning, middle or end
- Char: 3 random char-level operations (del, ins, sub, swap)
- BLEU consistency: BLEU between translations of clean and noisy inputs

2.5 Conclusion

- 12-2 > 6-6 for multilingual MT (faster and even better quality)
- Lang-specific vocab filtering improves speed
- RNN decoder: very good speed/BLEU tradeoff
- New MNMT setup on ParaCrawl

3. Efficient domain adaptation

Multilingual Domain Adaptation for NMT:
Decoupling Language and Domain Information with Adapters

A. Cooper Stickland, A. Berard and V. Nikoulina
WMT 2021

3.1 Introduction

A single model for many domains and languages

How can we adapt an MNMT model to a new domain in a *parameter-efficient* way?

3.1 Introduction

A single model for many domains and languages

How can we adapt an MNMT model to a new domain in a *parameter-efficient* way?

Covering high and low resource languages

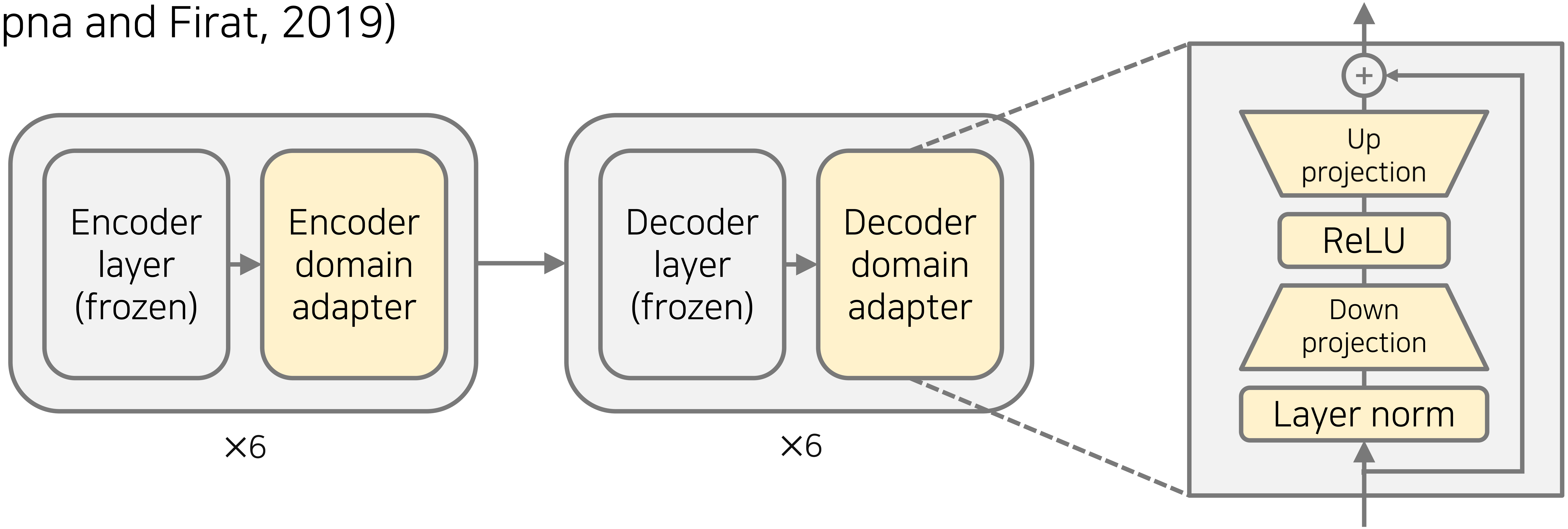
How can we do multilingual domain adaptation with *incomplete* in-domain data?

3.2 Adapter layers

Adapter layers are lightweight modules inserted in-between layers.

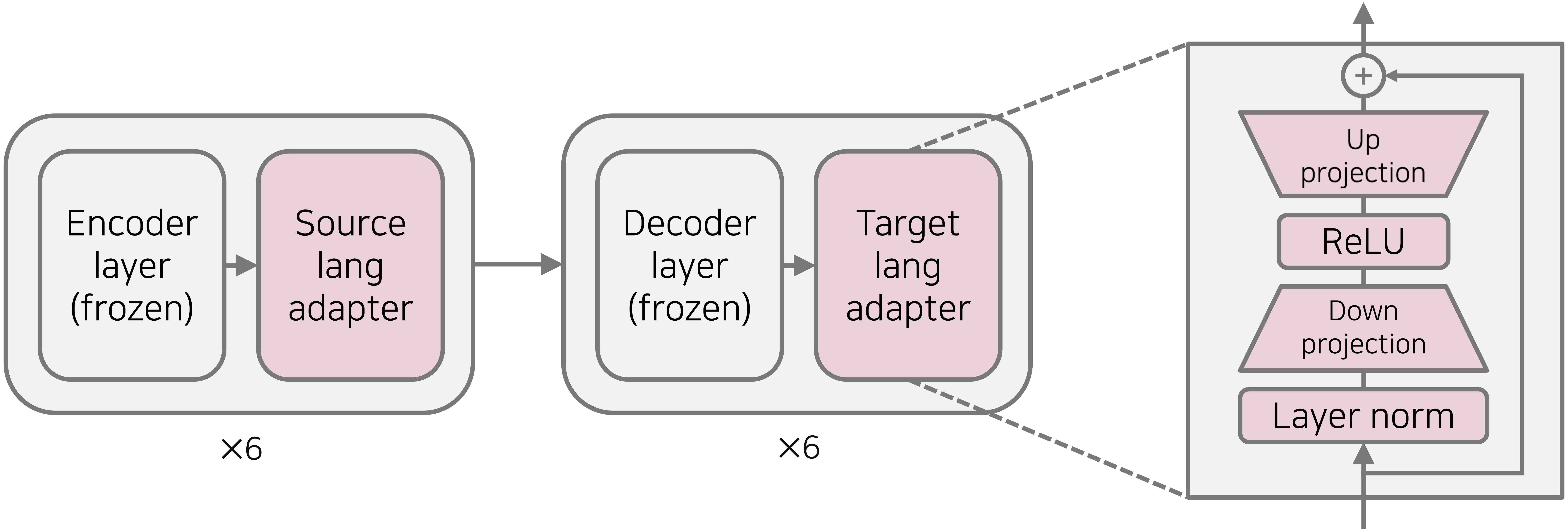
They can be trained to specialize to **language pairs** or **domains**

(Bapna and Firat, 2019)



3.2 Adapter layers

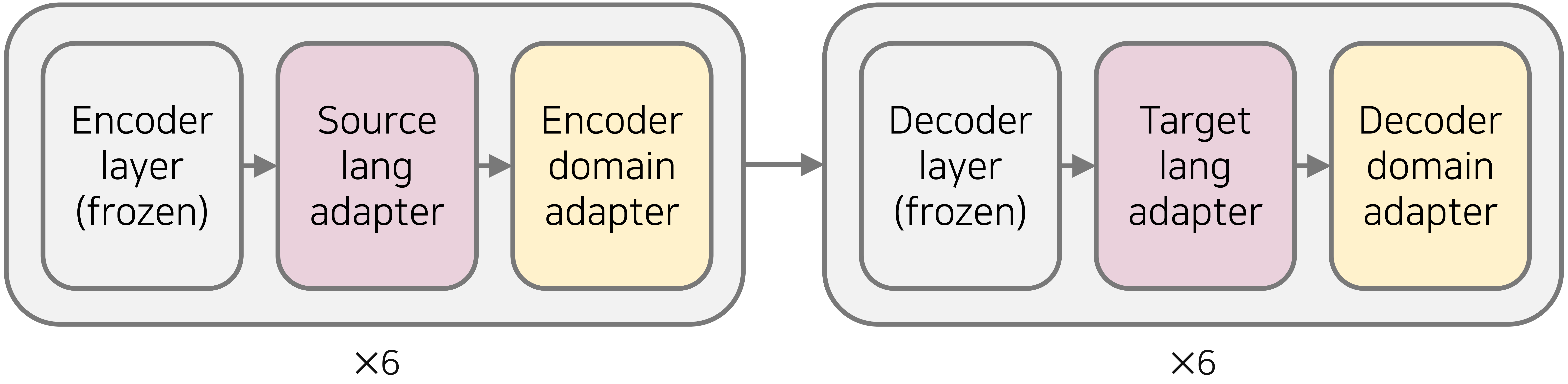
Language adapters are specific to one language and can be composed to perform **zero-shot machine translation** (Philip et al., 2020)



3.2 Adapter layers

Can we stack **domain** and **language adapters?**

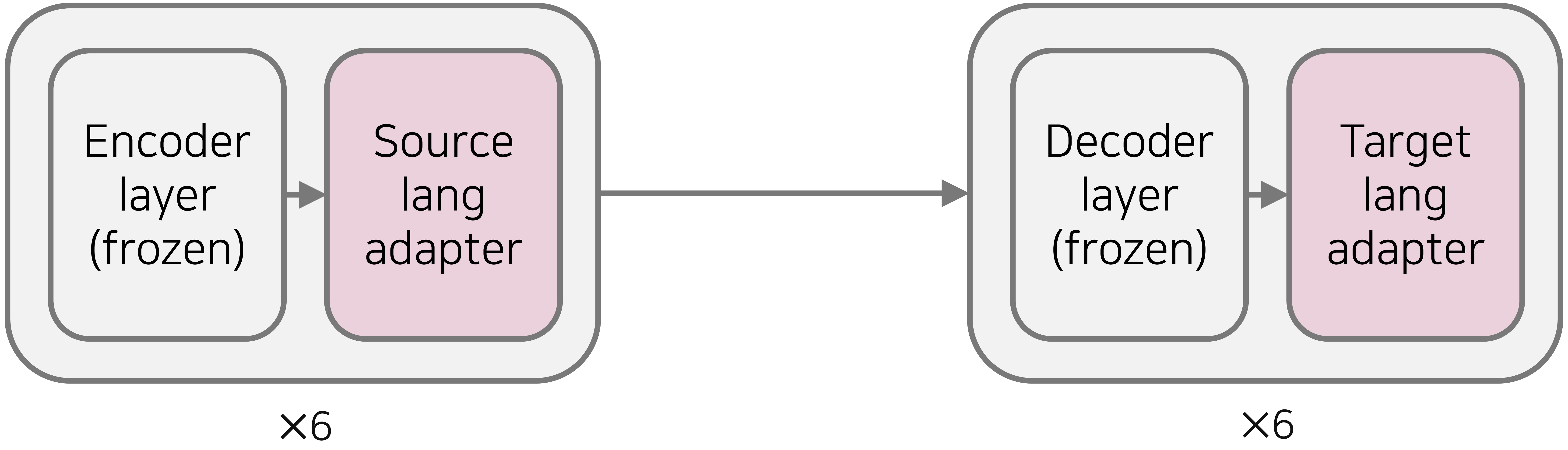
Pfeiffer et al. (2020) propose a similar approach for classification tasks.



3.3 Technique

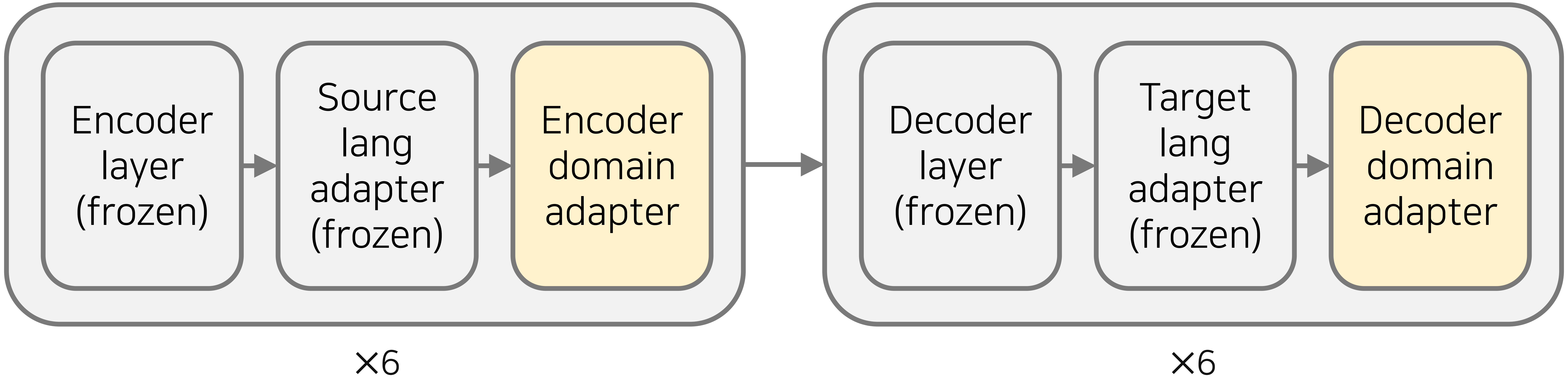
Train a baseline Transformer on English-centric ParaCrawl data

3.3 Technique



Train language adapters on multi-parallel ParaCrawl data

3.3 Technique



Train domain adapters on in-domain data

3.4 Experiments

Baseline MNMT model

- Trained on ParaCrawl Top 12: {FR, DE, ES, IT, PT, NL, NO, CS, PL, SV, DA} ↔ EN
- With *language adapters* trained on multi-parallel data (12x11 language pairs)

3.4 Experiments

Baseline MNMT model

- Trained on ParaCrawl Top 12: {FR, DE, ES, IT, PT, NL, NO, CS, PL, SV, DA} ↔ EN
- With *language adapters* trained on multi-parallel data (12x11 language pairs)

Domain adaptation

- In-domain data for Medical domain (+ Quran, IT and Ted Talks)
- Fine-tune on a subset of languages (EN, FR, DE, CS)
- Evaluate on all 132 language pairs

3.4 Experiments

Baseline MNMT model

- Trained on ParaCrawl Top 12: {FR, DE, ES, IT, PT, NL, NO, CS, PL, SV, DA} ↔ EN
- With *language adapters* trained on multi-parallel data (12x11 language pairs)

Domain adaptation

- In-domain data for Medical domain (+ Quran, IT and Ted Talks)
- Fine-tune on a subset of languages (EN, FR, DE, CS)
- Evaluate on all 132 language pairs

Models

- Stacking domain and language adapters (at all layers, encoder-only or decoder only)
- (Vanilla fine-tuning and domain tags)

3.4 Experiments

Baseline MNMT model

- Trained on ParaCrawl Top 12: {FR, DE, ES, IT, PT, NL, NO, CS, PL, SV, DA} ↔ EN
- With *language adapters* trained on multi-parallel data (12x11 language pairs)

Domain adaptation

- In-domain data for Medical domain (+ Quran, IT and Ted Talks)
- Fine-tune on a subset of languages (EN, FR, DE, CS)
- Evaluate on all 132 language pairs

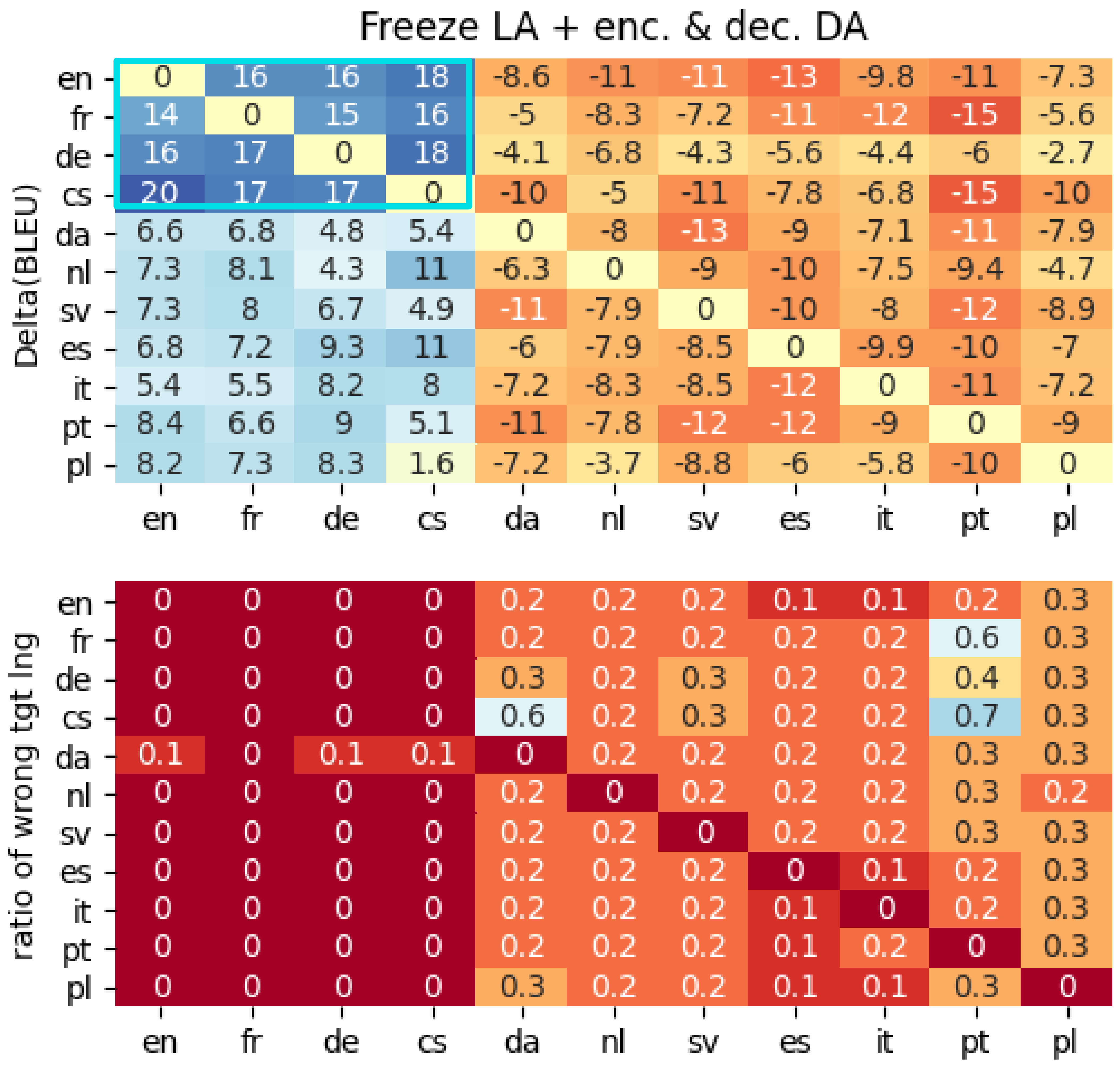
Models

- Stacking domain and language adapters (at all layers, encoder-only or decoder only)
- (Vanilla fine-tuning and domain tags)

Terminology

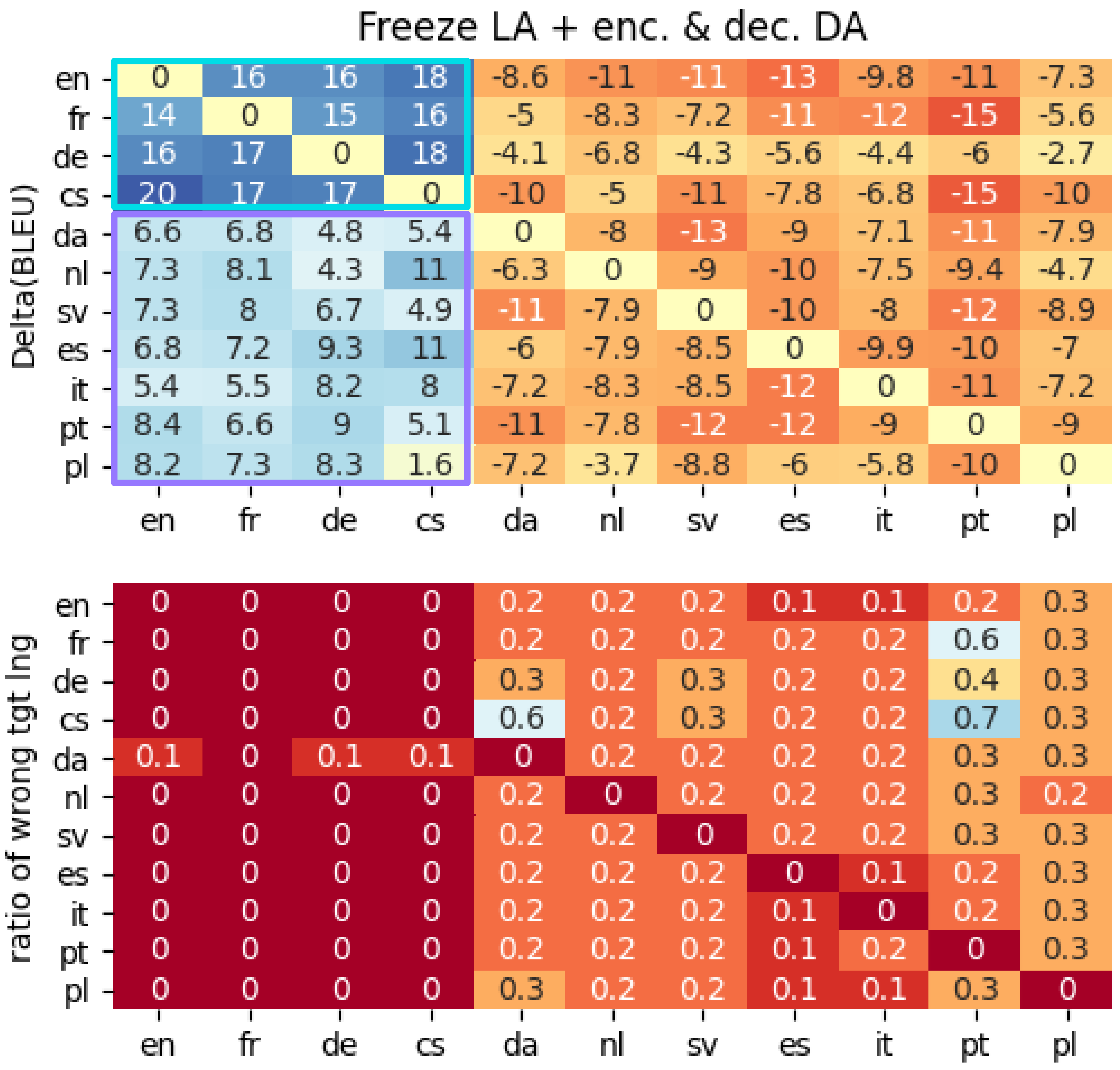
- in = languages with in-domain data (EN, FR, DE, CS)
- out = languages without in-domain data (ES, IT, PT, NL, NO, PL, SV, DA)

3.5 Vanilla adapter stacking (all layers)



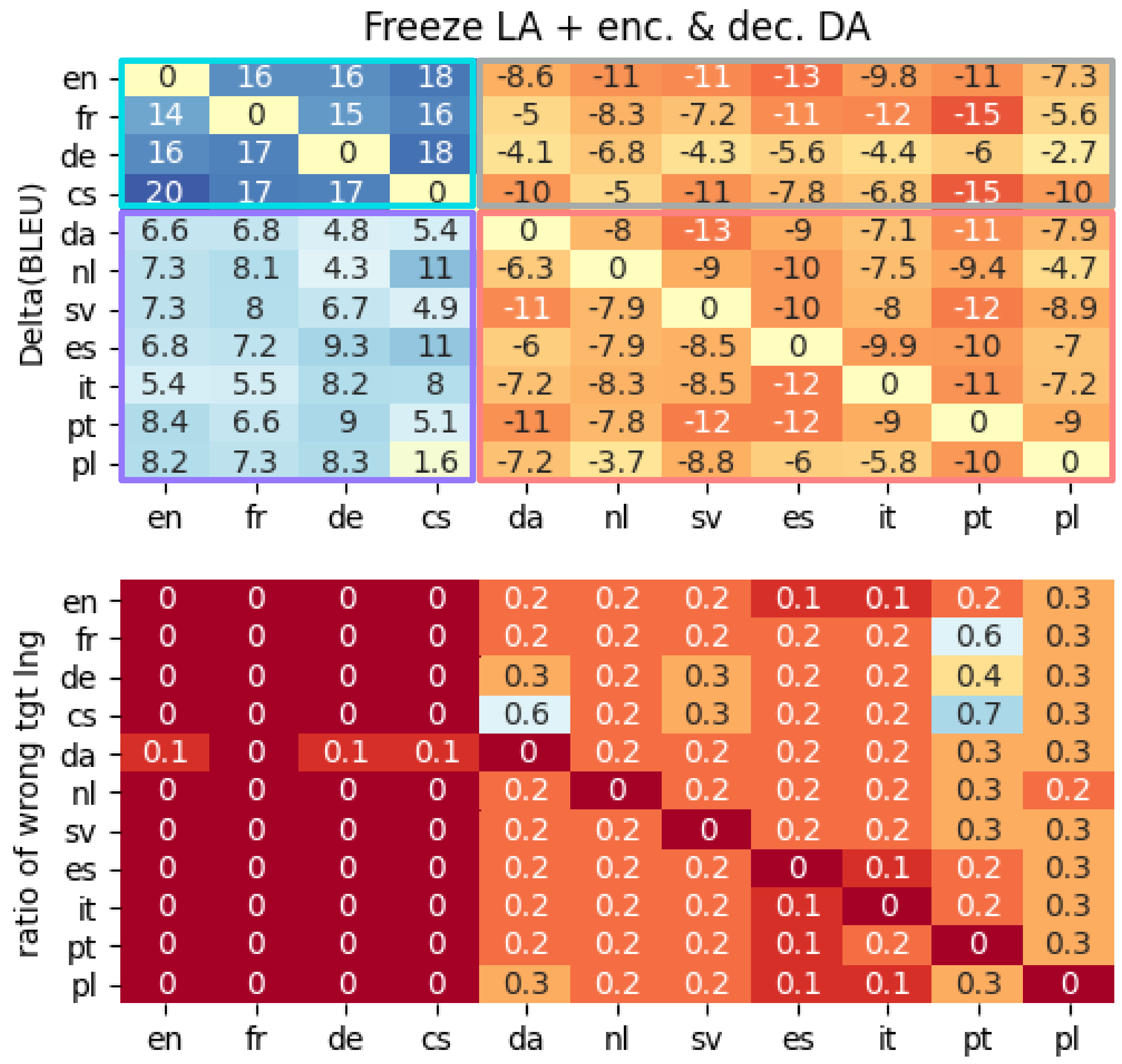
- Big improvements for in-in language pairs

3.5 Vanilla adapter stacking (all layers)



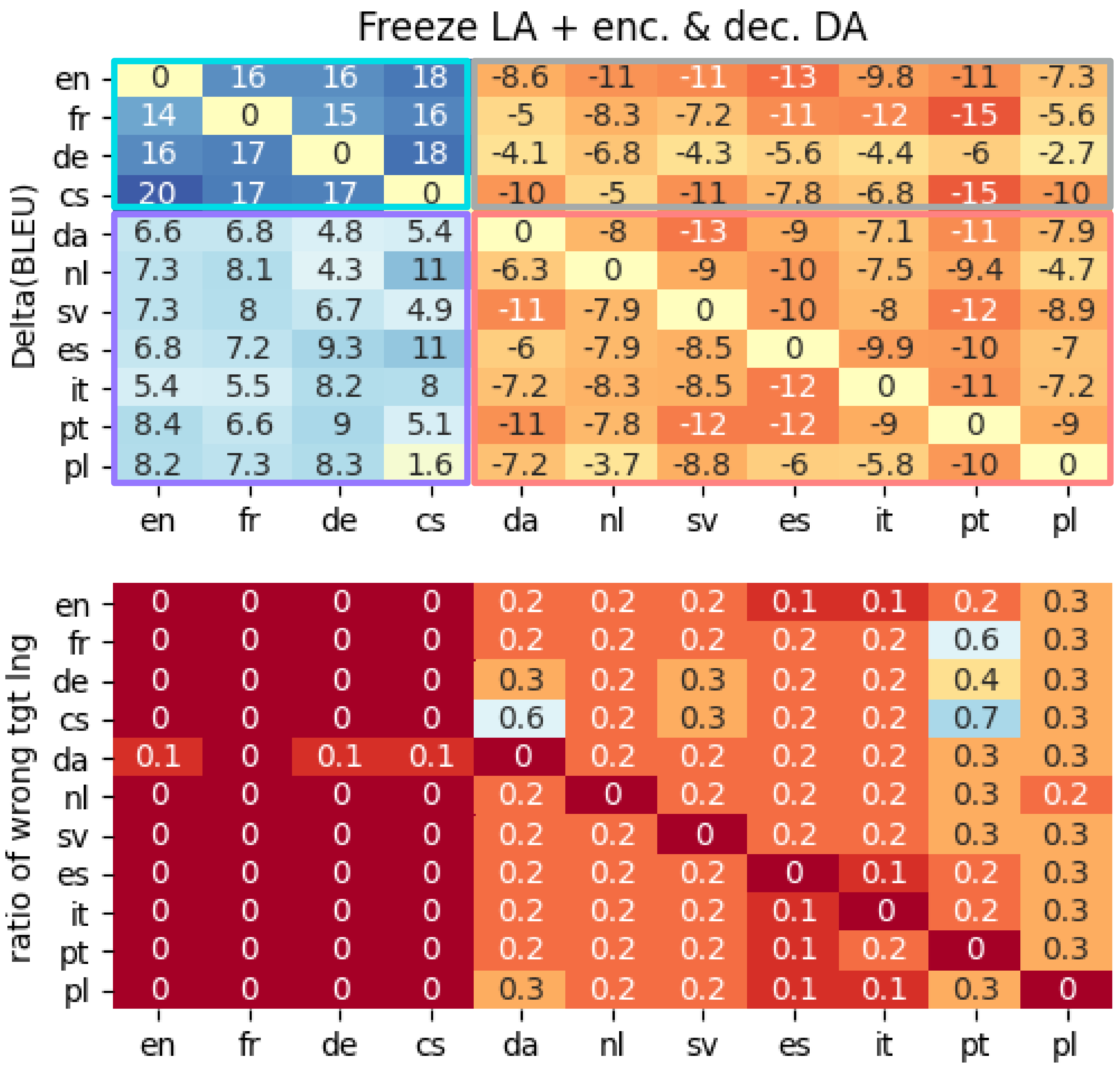
- Big improvements for in-in language pairs
- Some improvements for out-in language pairs

3.5 Vanilla adapter stacking (all layers)



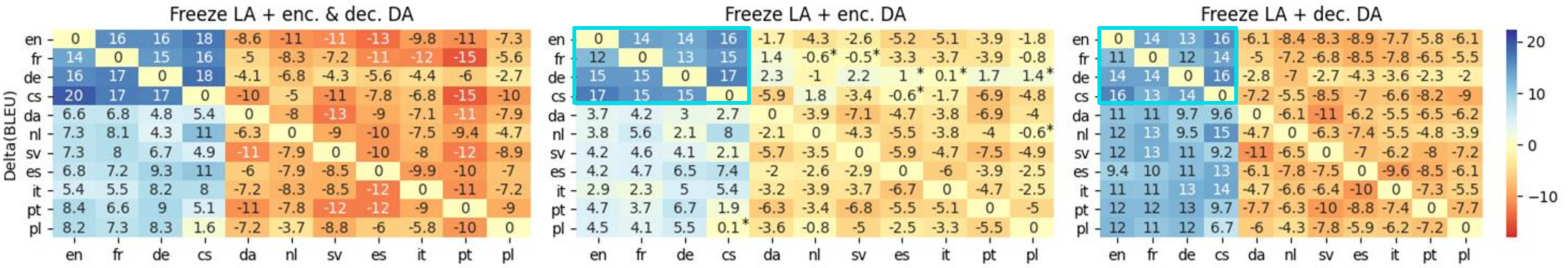
- Big improvements for in-in language pairs
- Some improvements for out-in language pairs
- Degradation for in-out and out-out language pairs
  - Partly due to generation in the wrong language

3.5 Vanilla adapter stacking (all layers)



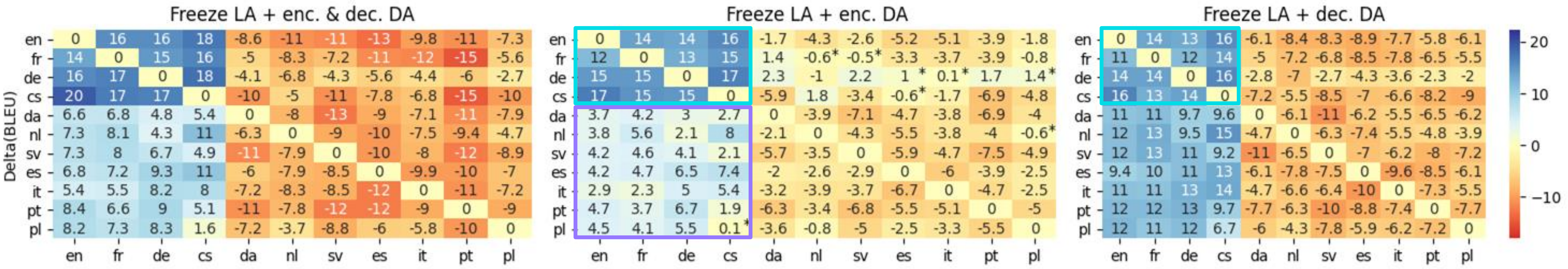
- Big improvements for in-in language pairs
- Some improvements for out-in language pairs
- Degradation for in-out and out-out language pairs
  - Partly due to generation in the wrong language
- Domain adapters seem to “erase out” language knowledge from the model
- Hard to decouple language knowledge from domain knowledge

3.6 Encoder-only or decoder-only domain adapters



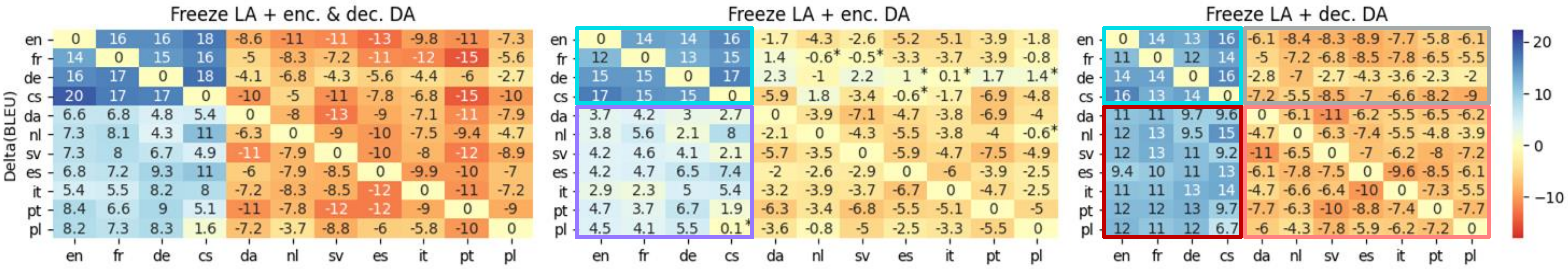
- Both: slight decrease in in-in translations (due to lower DA capacity)

3.6 Encoder-only or decoder-only domain adapters



- Both: slight decrease in in-in translations (due to lower DA capacity)
- Encoder only: less off-target translation, but lower out-in performance

3.6 Encoder-only or decoder-only domain adapters



- Both: slight decrease in in-in translations (due to lower DA capacity)
- Encoder only: less off-target translation, but lower out-in performance
- Decoder only: better out-in performance, but worse out-out and in-out performance

3.7 Regularization and data augmentation

DADrop: domain adapter drop

- Randomly drop adapters during training
- Motivation: reduce “language overfitting” effect

3.7 Regularization and data augmentation

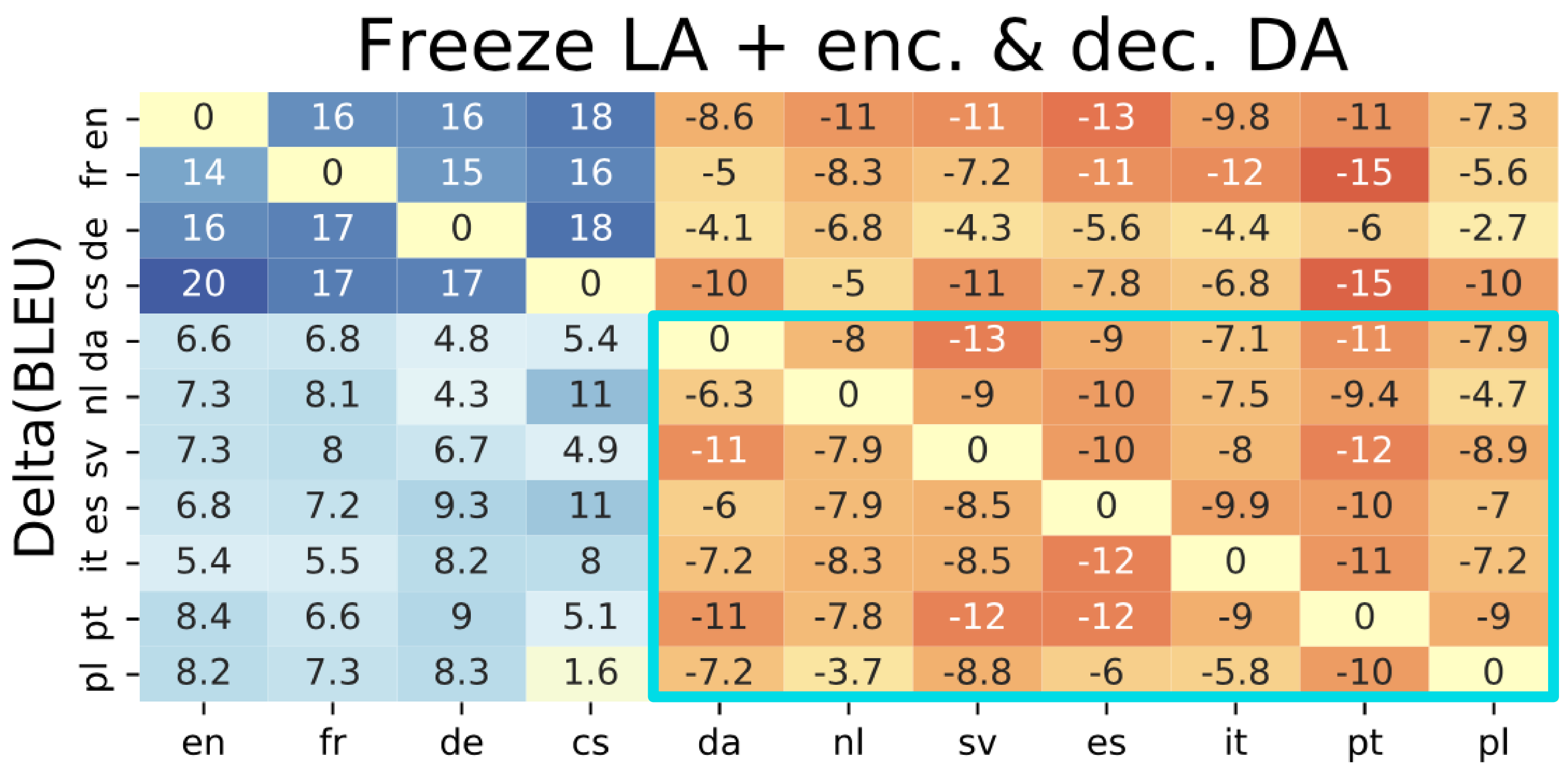
DADrop: domain adapter drop

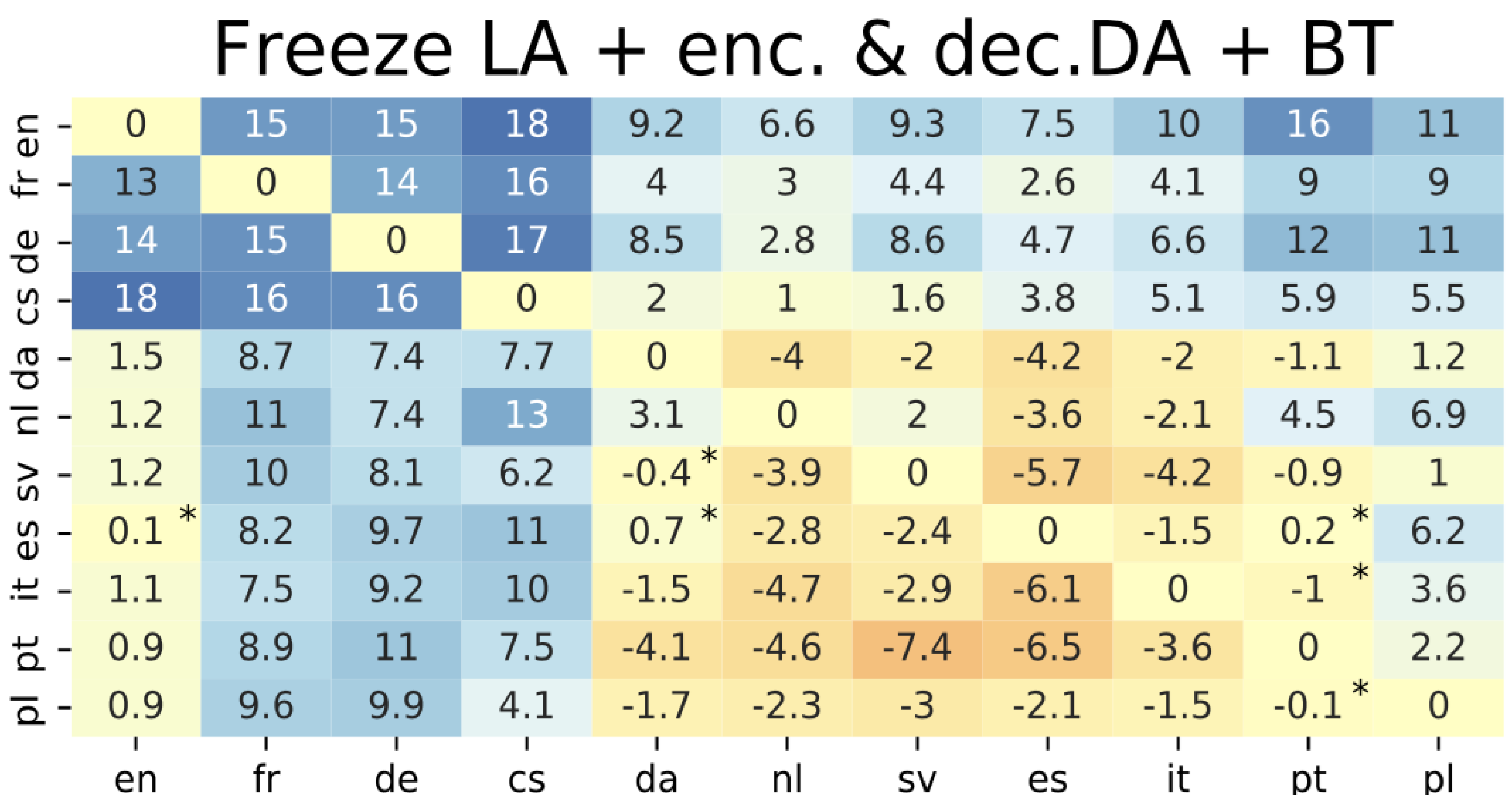
- Randomly drop adapters during training
- Motivation: reduce “language overfitting” effect

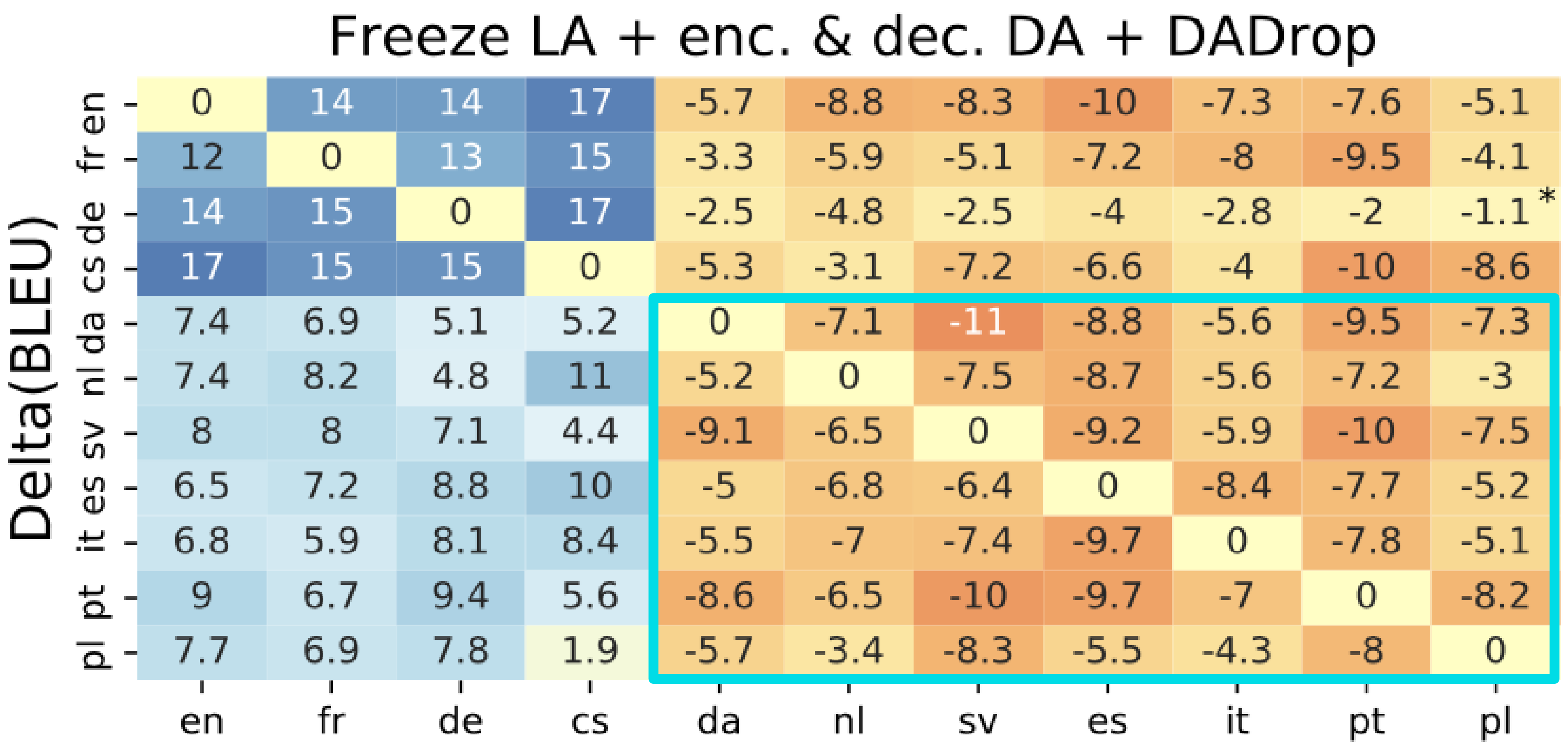
Back-translation

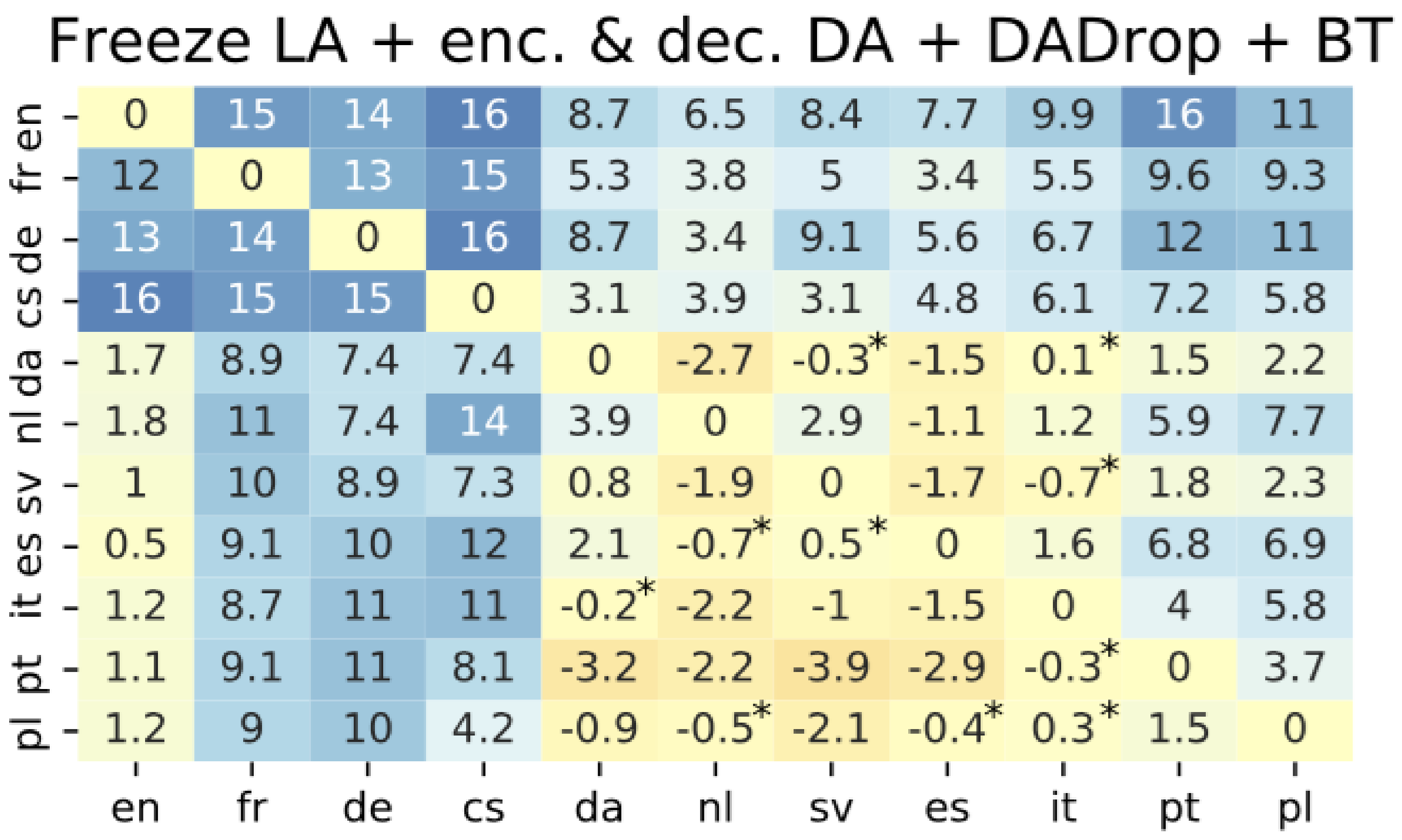
Use baseline model to back-translate in-domain data from and into English for out languages

3.7 Regularization and data augmentation





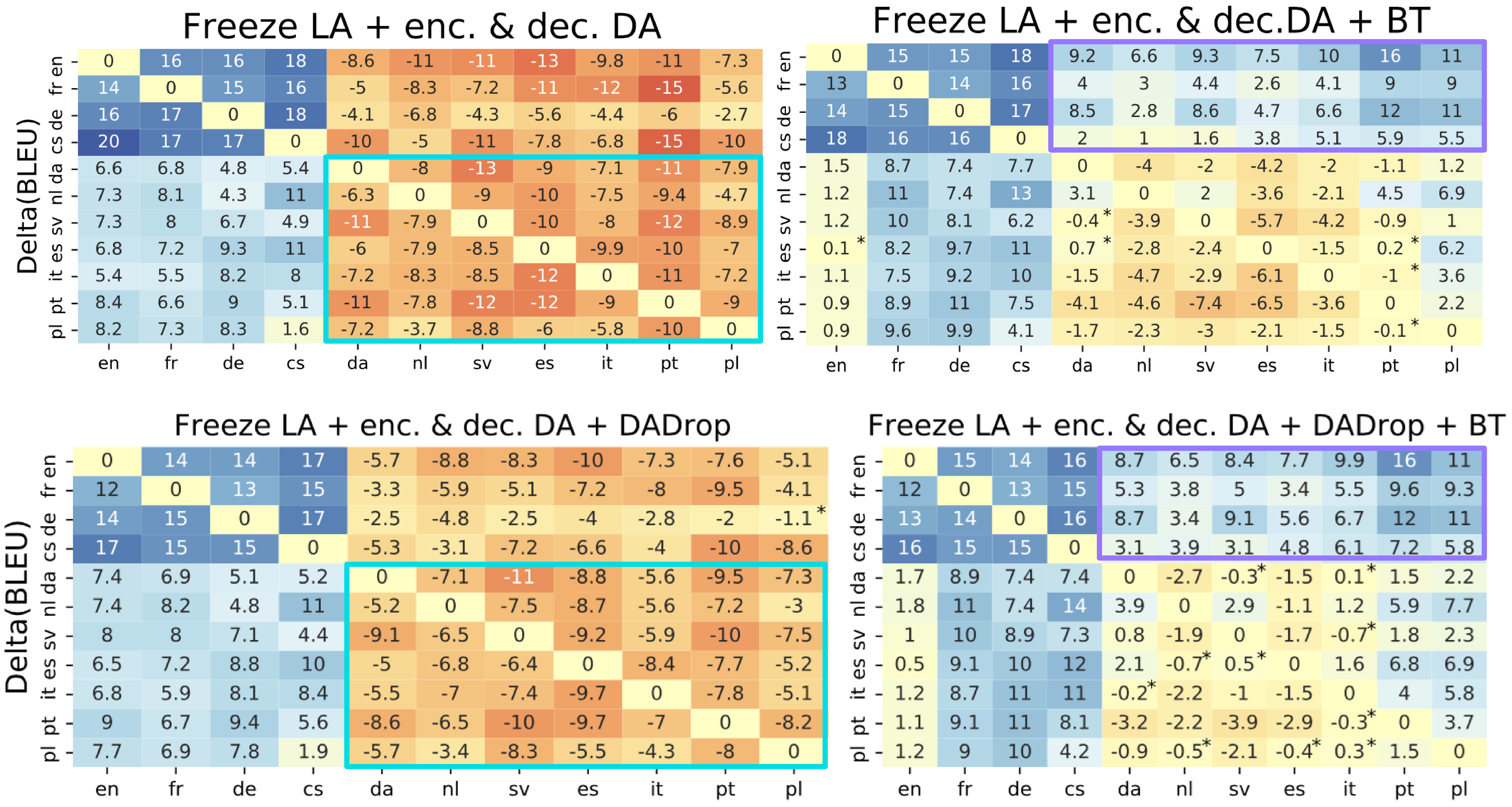




DADrop

- Improves out-out translation
- But off-target translations persist

3.7 Regularization and data augmentation



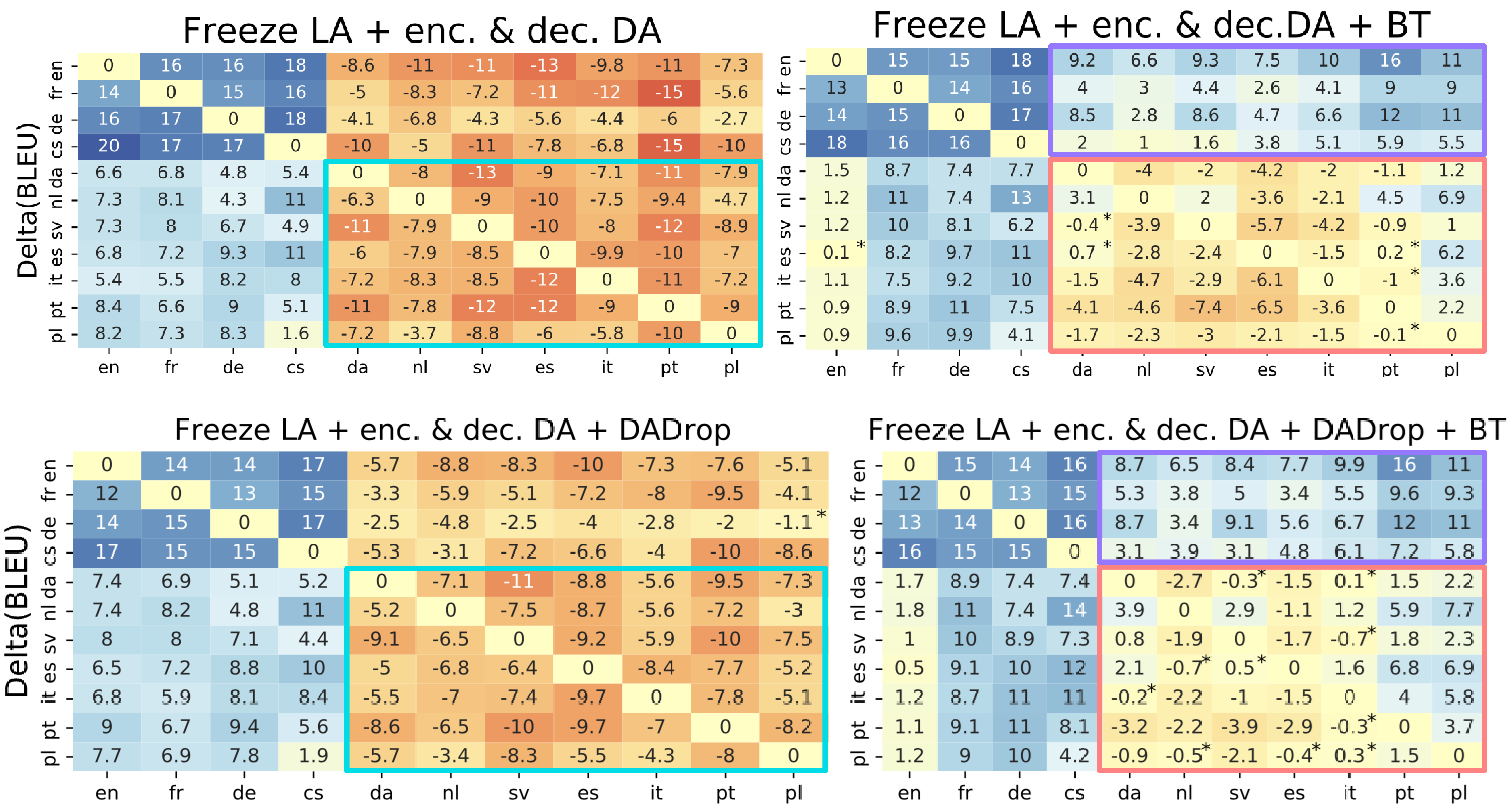
DADrop

- Improves out-out translation
- But off-target translations persist

Back-translation (BT)

- Solves off-target translation
- Improves in-out translation

3.7 Regularization and data augmentation



DADrop

- Improves out-out translation
- But off-target translations persist

Back-translation (BT)

- Solves off-target translation
- Improves in-out translation
- Effect of DA for out-out is small

3.8 Conclusion

It is hard to properly decouple language knowledge from domain knowledge

- Contrary to Pfeiffer et al. (2020), who use encoder-only classification models
- Generation tasks require good language-specific representations
- Encoder-only or decoder-only adapters have useful properties

3.8 Conclusion

It is hard to properly decouple language knowledge from domain knowledge

- Contrary to Pfeiffer et al. (2020), who use encoder-only classification models
- Generation tasks require good language-specific representations
- Encoder-only or decoder-only adapters have useful properties

Regularization and data augmentation techniques can help

DEVIEW 2021

4. Learning new languages efficiently

Continual Learning in Multilingual NMT via Language-Specific Embeddings

Alexandre Berard
WMT 2021

4.1 Introduction

Given an existing MNMT model

{FR, DE, EN} → {FR, DE, EN}

4.1 Introduction

Given an existing MNMT model

{FR, DE, EN} → {FR, DE, EN}

How can we efficiently add a new source language?

{FR, DE, EN, EL} → {FR, DE, EN}

4.1 Introduction

Given an existing MNMT model

{FR, DE, EN} → {FR, DE, EN}

How can we efficiently add a new source language?

{FR, DE, EN, EL} → {FR, DE, EN}

or a new target language?

{FR, DE, EN} → {FR, DE, EN, EL}

4.1 Introduction

Our (self-imposed) constraints:

- No re-training on the initial language pairs

4.1 Introduction

Our (self-imposed) constraints:

- No re-training on the initial language pairs
- No performance drop on the initial language pairs

4.1 Introduction

Our (self-imposed) constraints:

- No re-training on the initial language pairs
- No performance drop on the initial language pairs
- Good zero-shot performance (train on EL→EN, evaluate on EL→FR)

4.1 Introduction

Our (self-imposed) constraints:

- No re-training on the initial language pairs
- No performance drop on the initial language pairs
- Good zero-shot performance (train on EL→EN, evaluate on EL→FR)
- No significant increase in model size

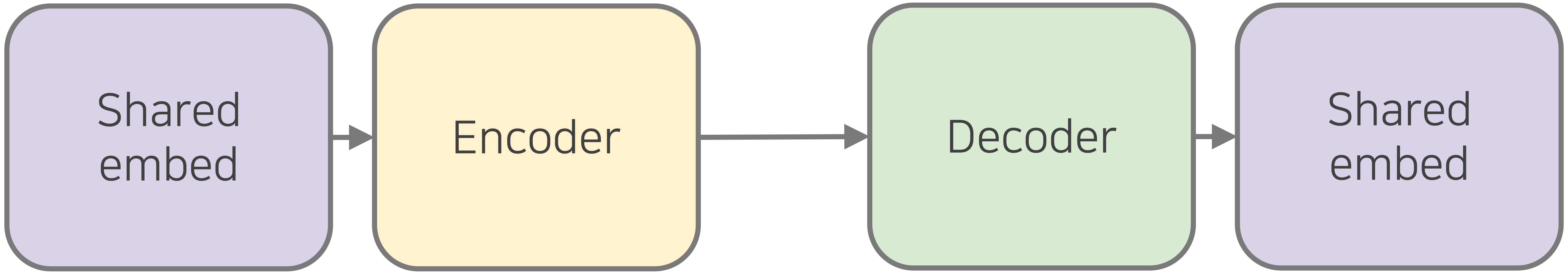
4.1 Introduction

Our (self-imposed) constraints:

- No re-training on the initial language pairs
- No performance drop on the initial language pairs
- Good zero-shot performance (train on EL→EN, evaluate on EL→FR)
- No significant increase in model size
- Fast training and inference

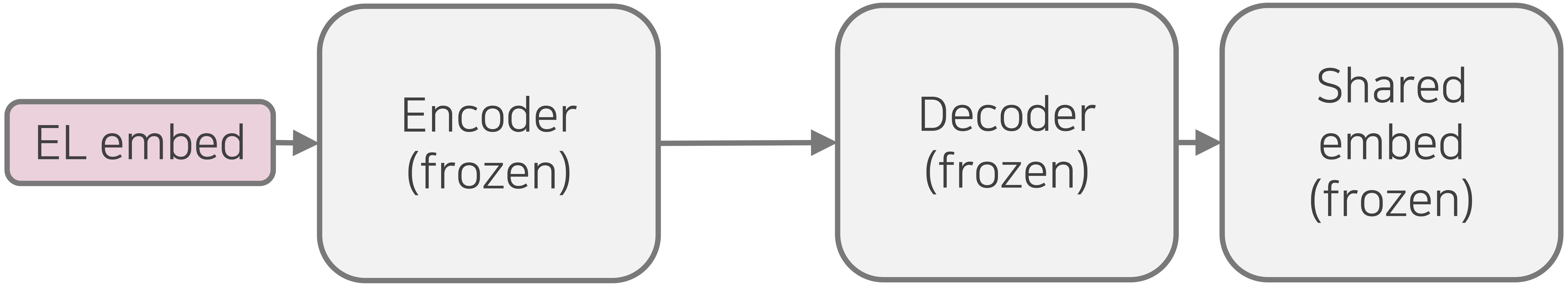
4.2 Technique: initial MNMT model

Train on many-to-many data (English-centric or multi-parallel)

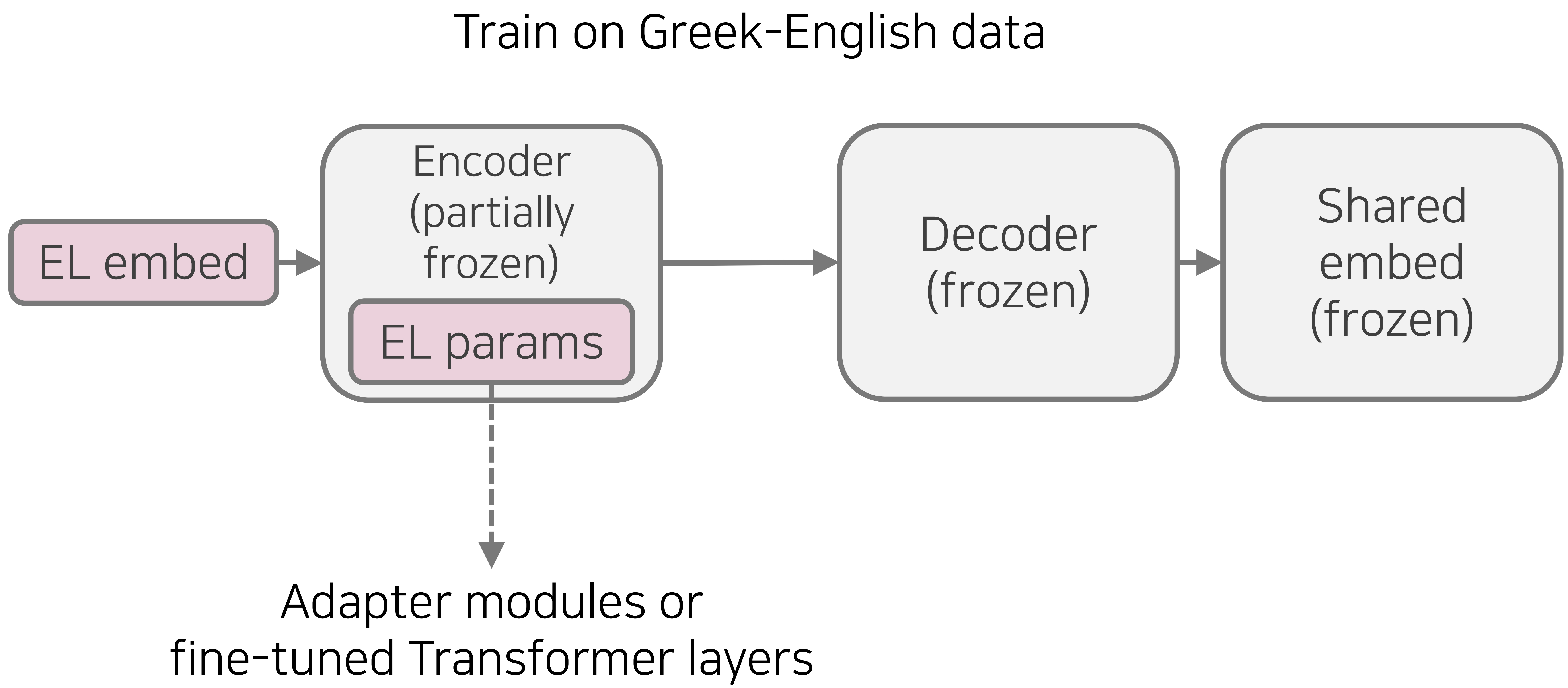


4.2 Technique: add a new source language

Train on Greek-English data

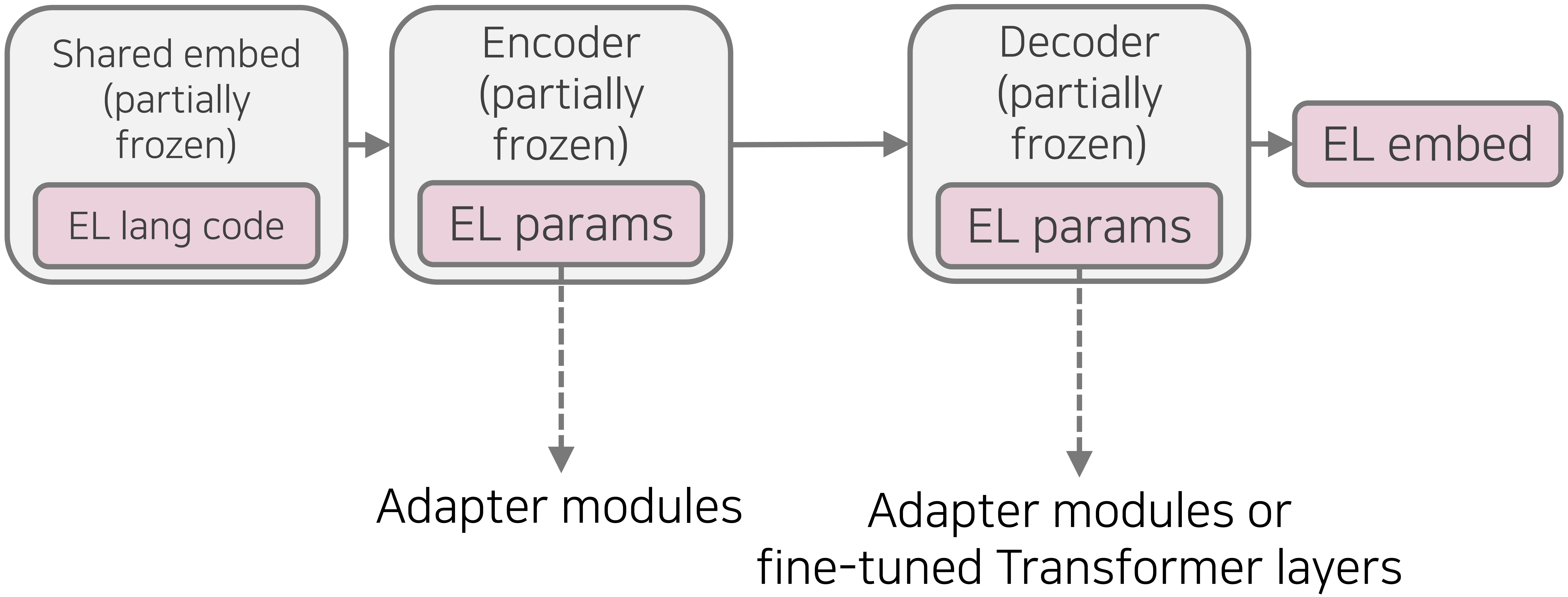


4.2 Technique: add a new source language



4.2 Technique: add a new target language

Train on English-Greek data



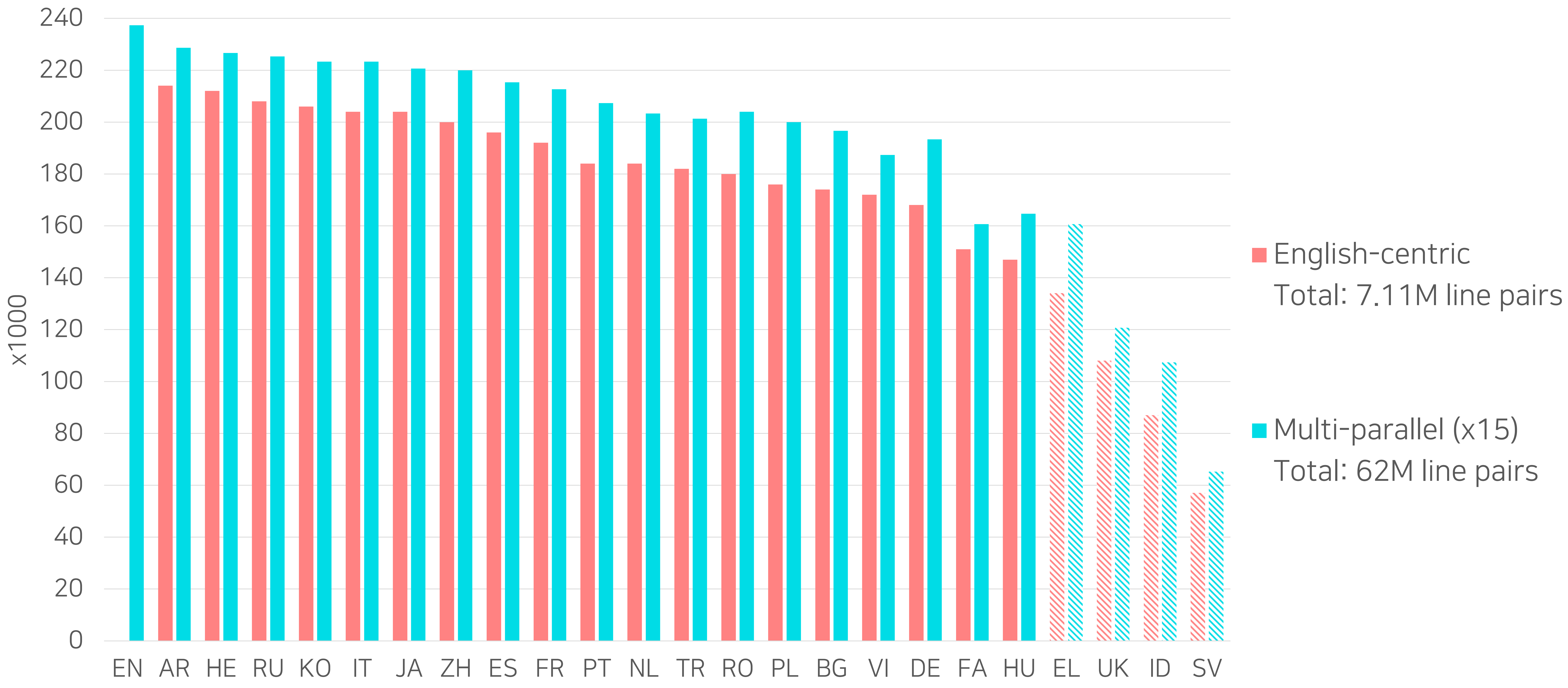
4.3 Experiments: TED Talks Top 20

Initial model: Transformer Base

4.3 Experiments: TED Talks Top 20

Initial model: Transformer Base

4.3 Experiments: TED Talks Top 20



Initial model: Transformer Base

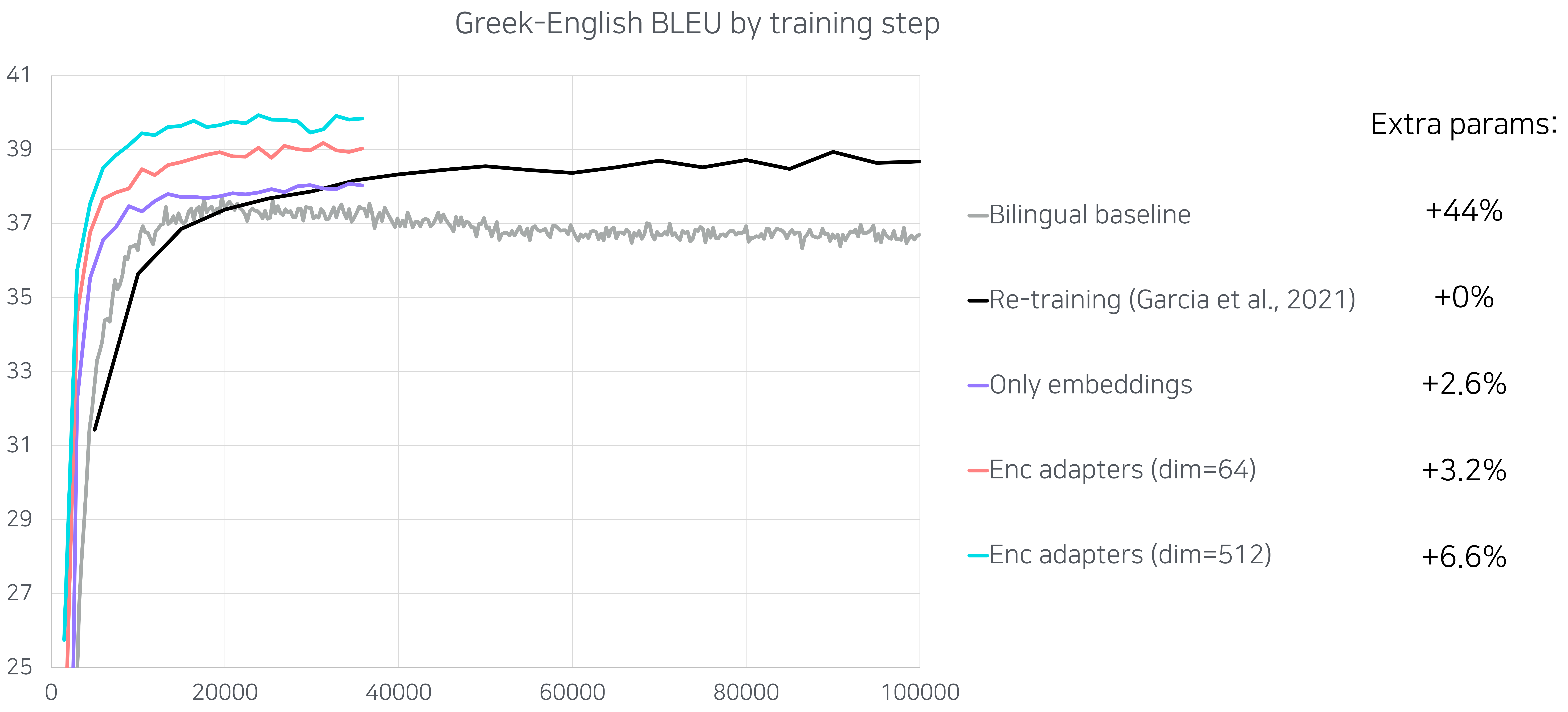
4.4 Results: new source language

Extra params:

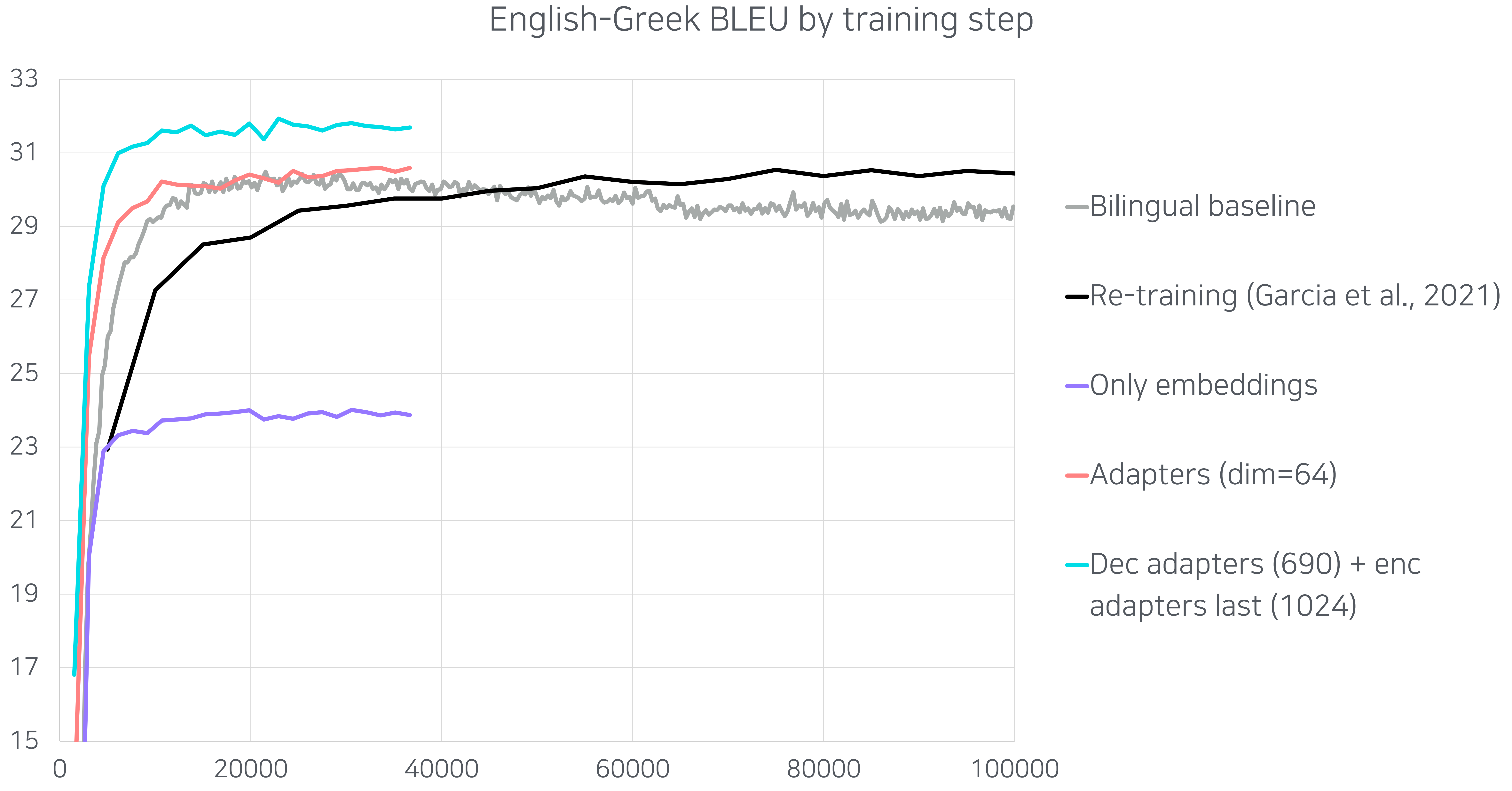
+44%

+0%

4.4 Results: new source language



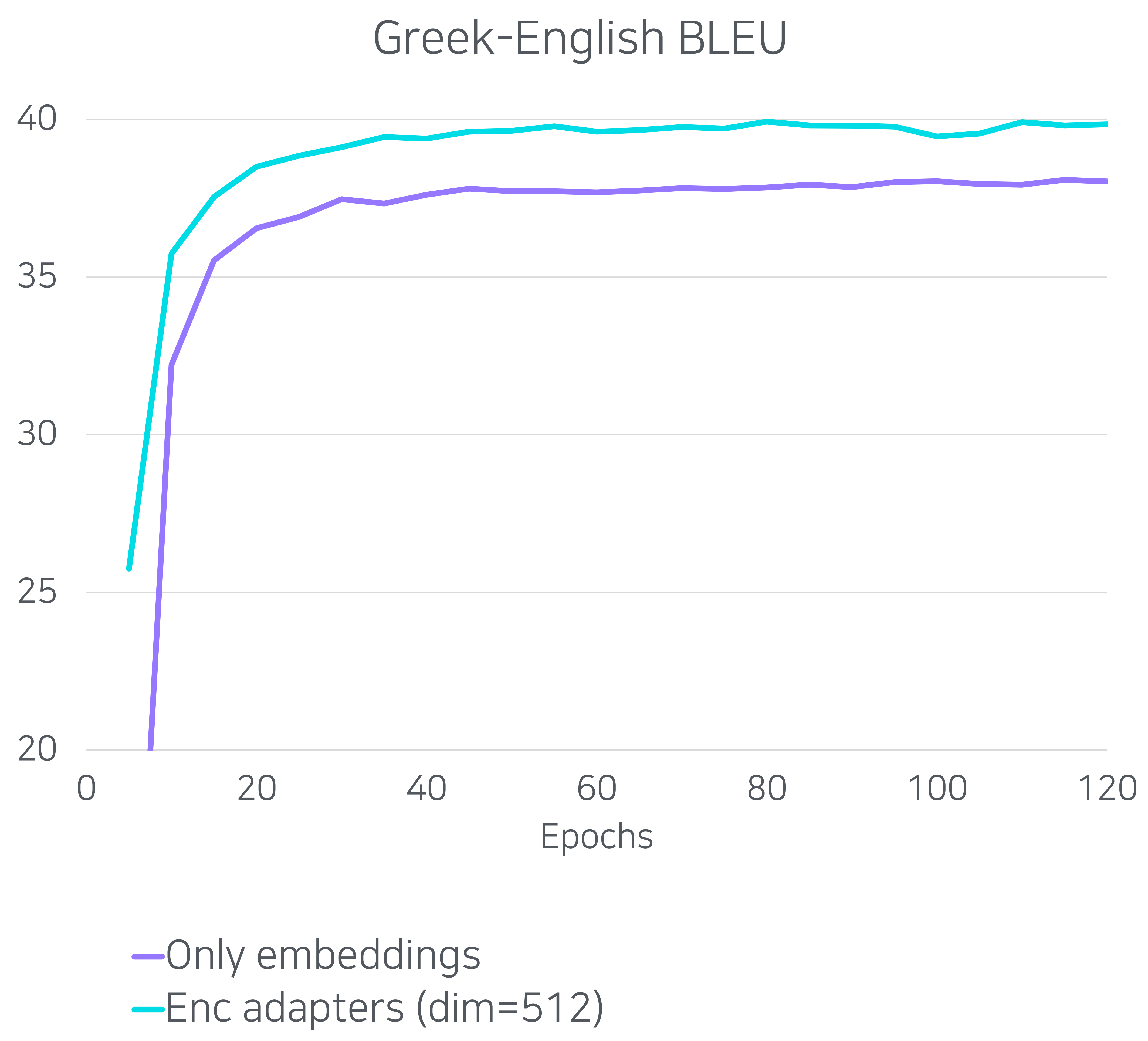
4.4 Results: new target language

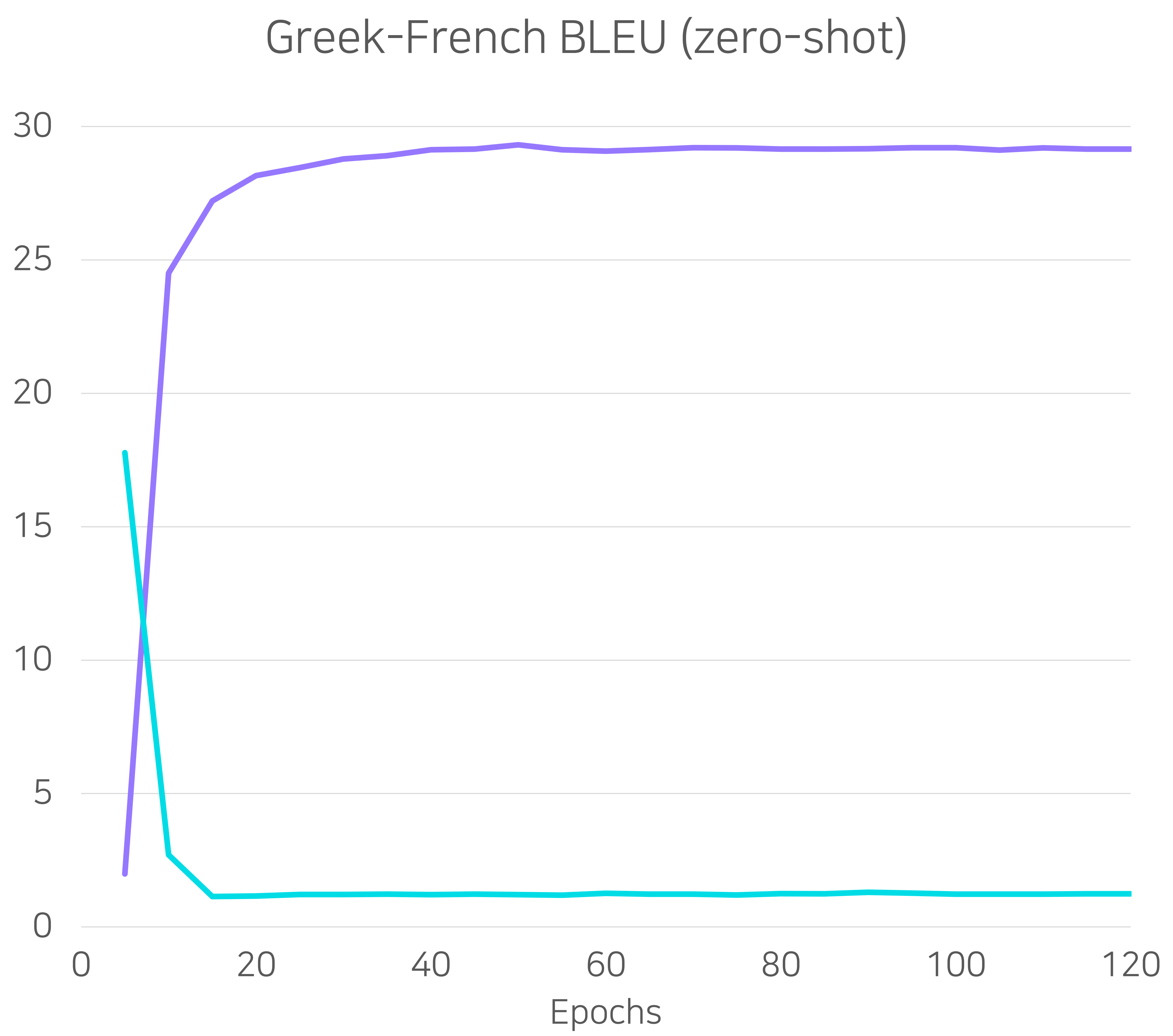


Extra params:

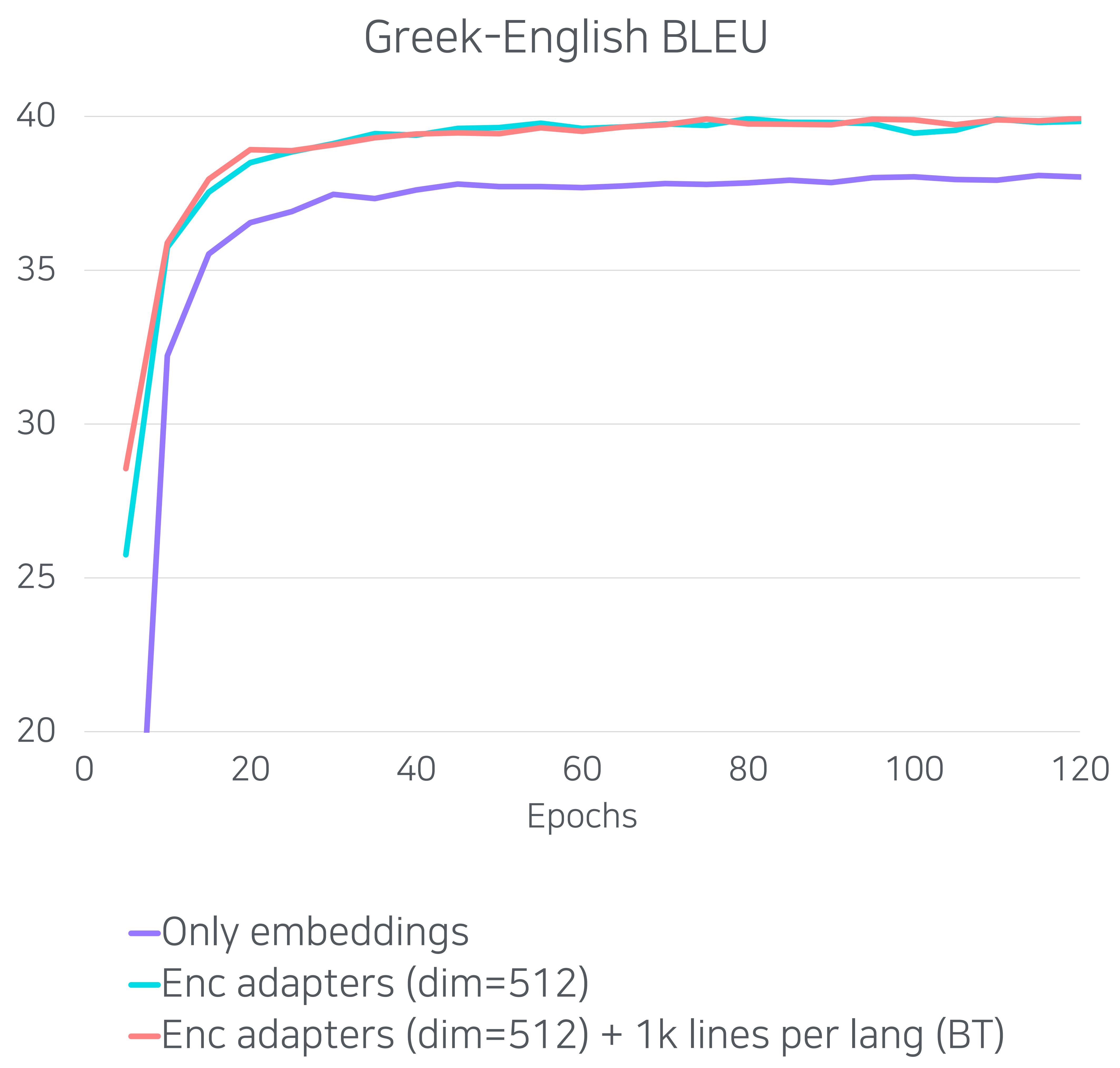
Bilingual baseline	+44%
Re-training (Garcia et al., 2021)	+0%
Only embeddings	+2.6%
Adapters (dim=64)	+3.7%
Dec adapters (690) + enc adapters last (1024)	+9.2%

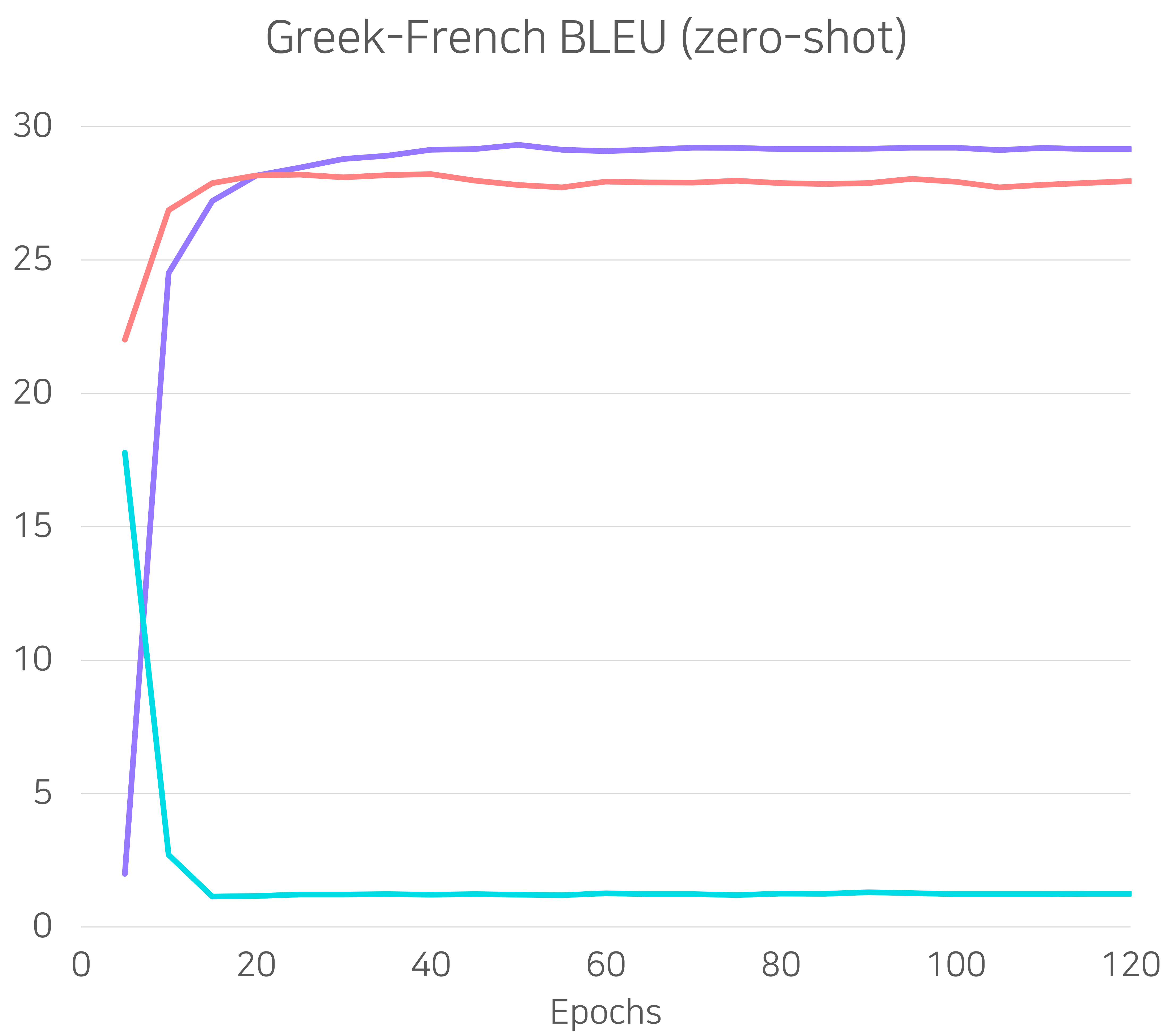
4.4 Results: zero-shot translation



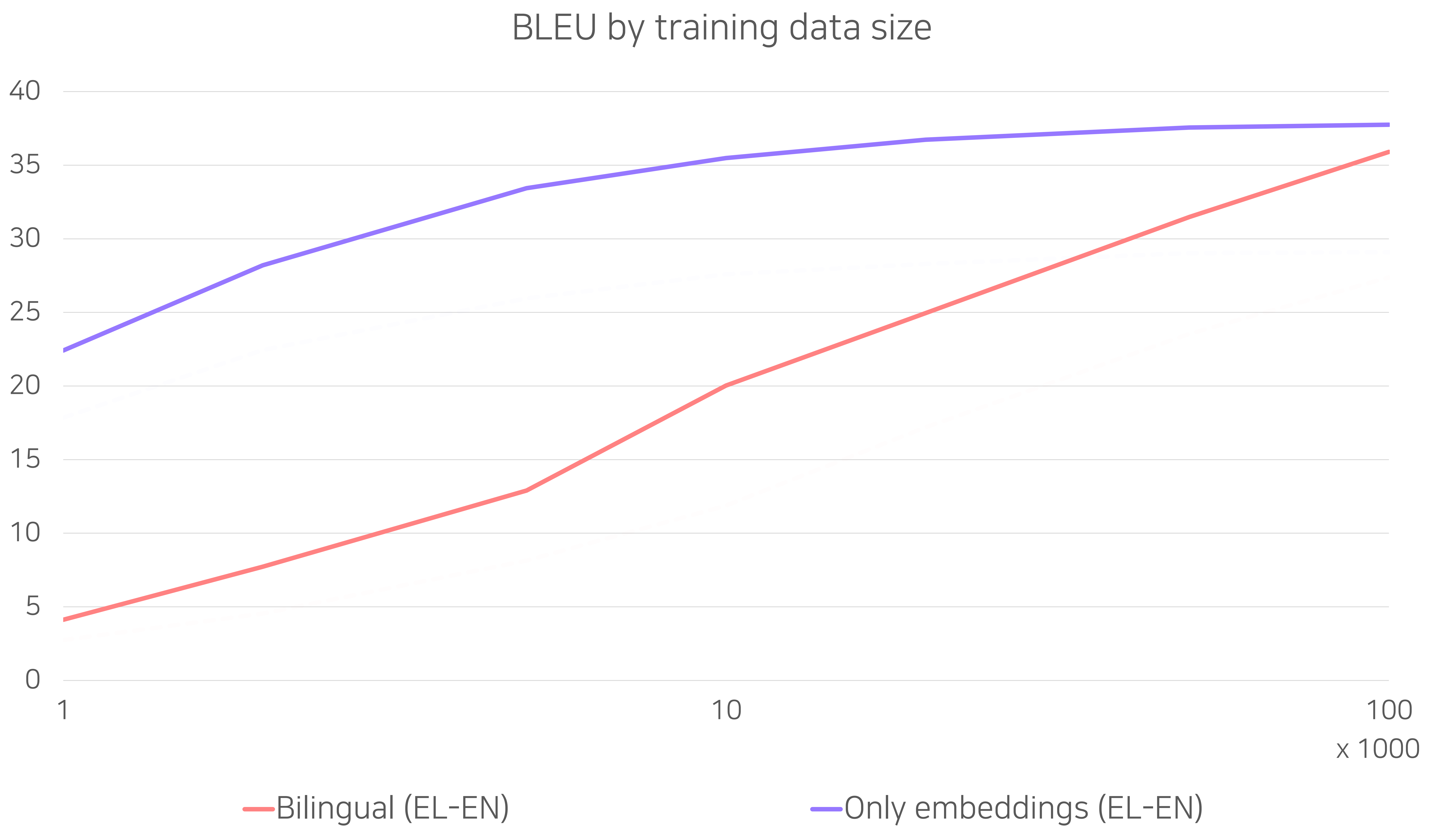


4.4 Results: zero-shot translation

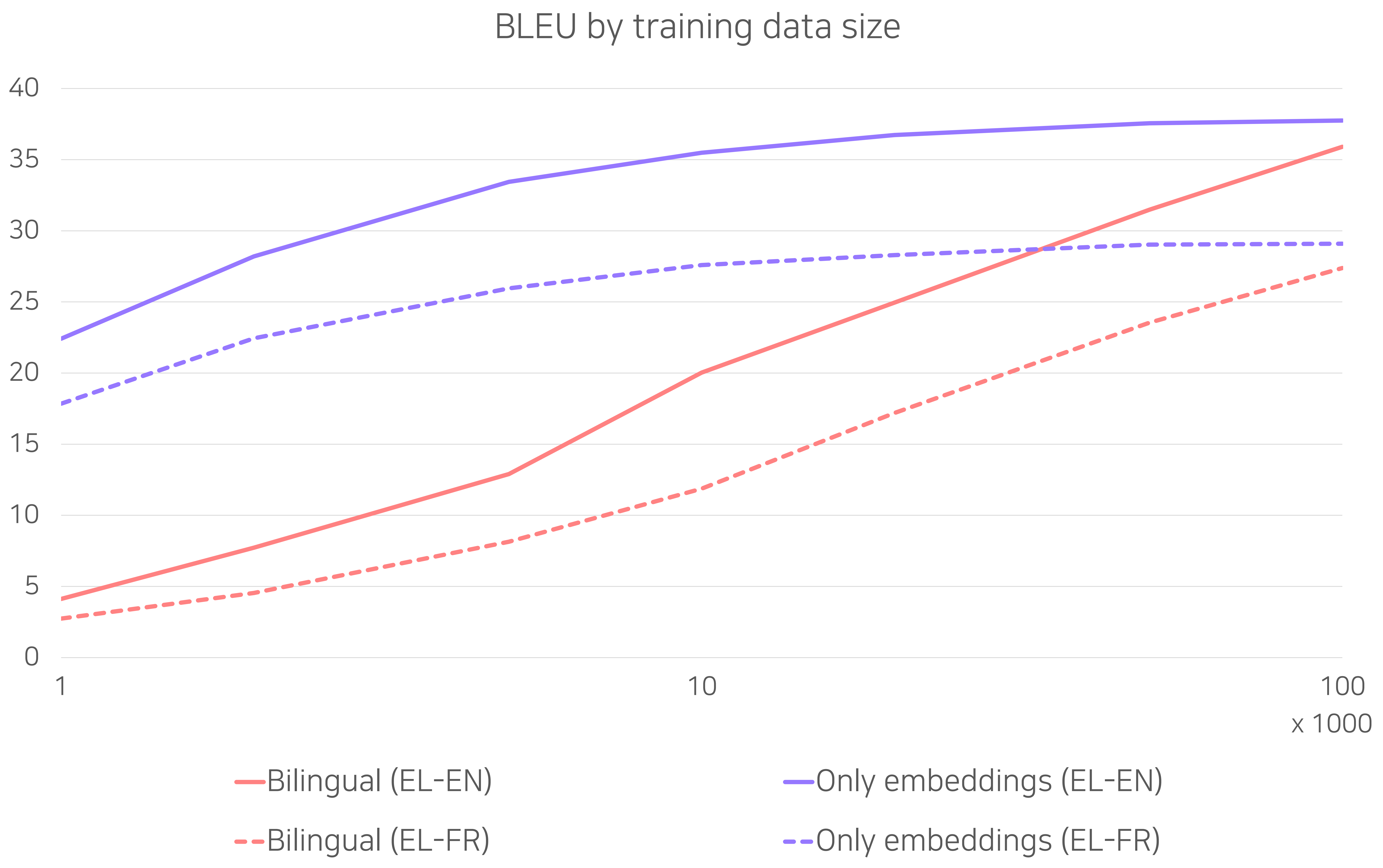




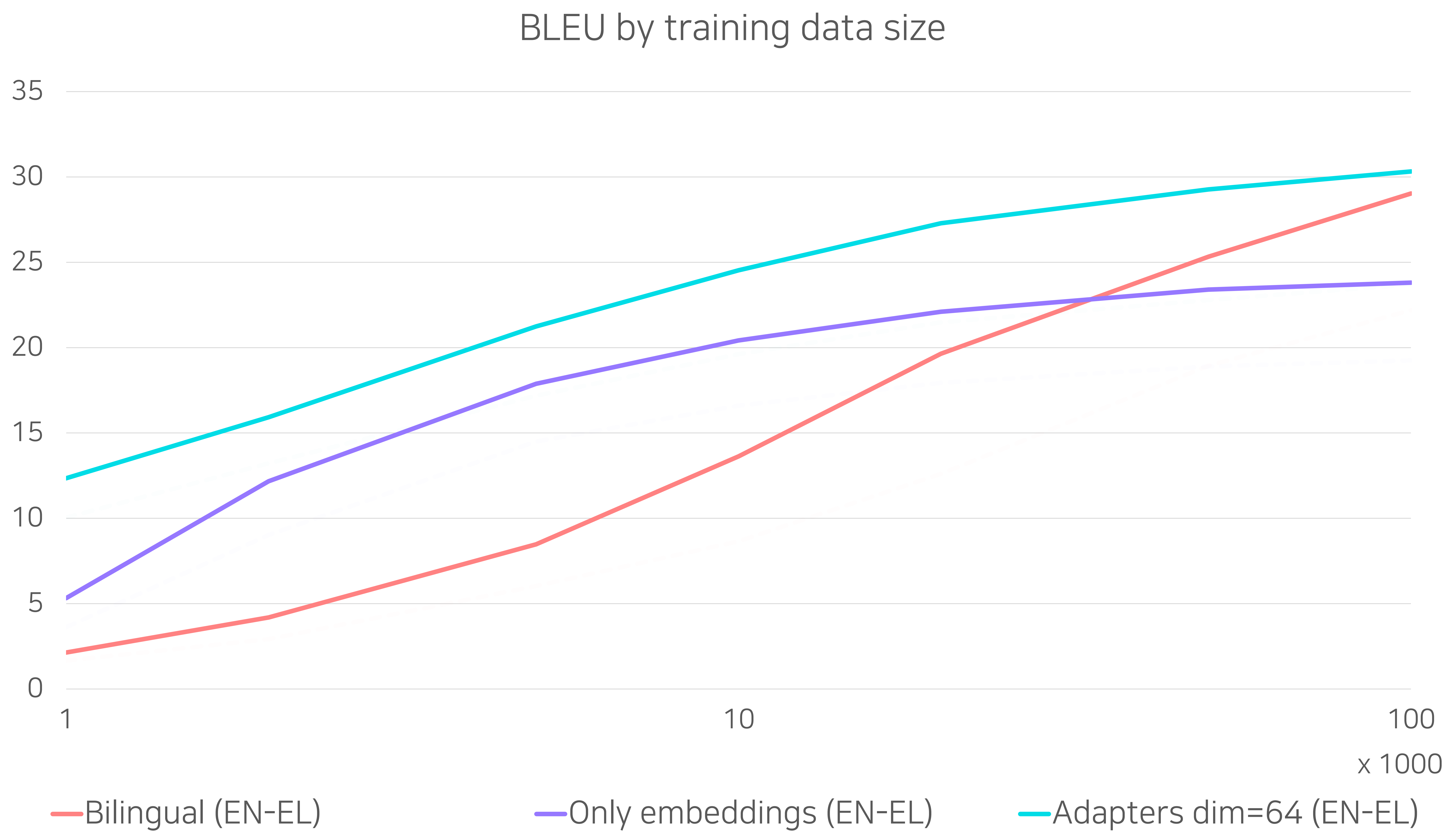
4.4 Results: data efficiency



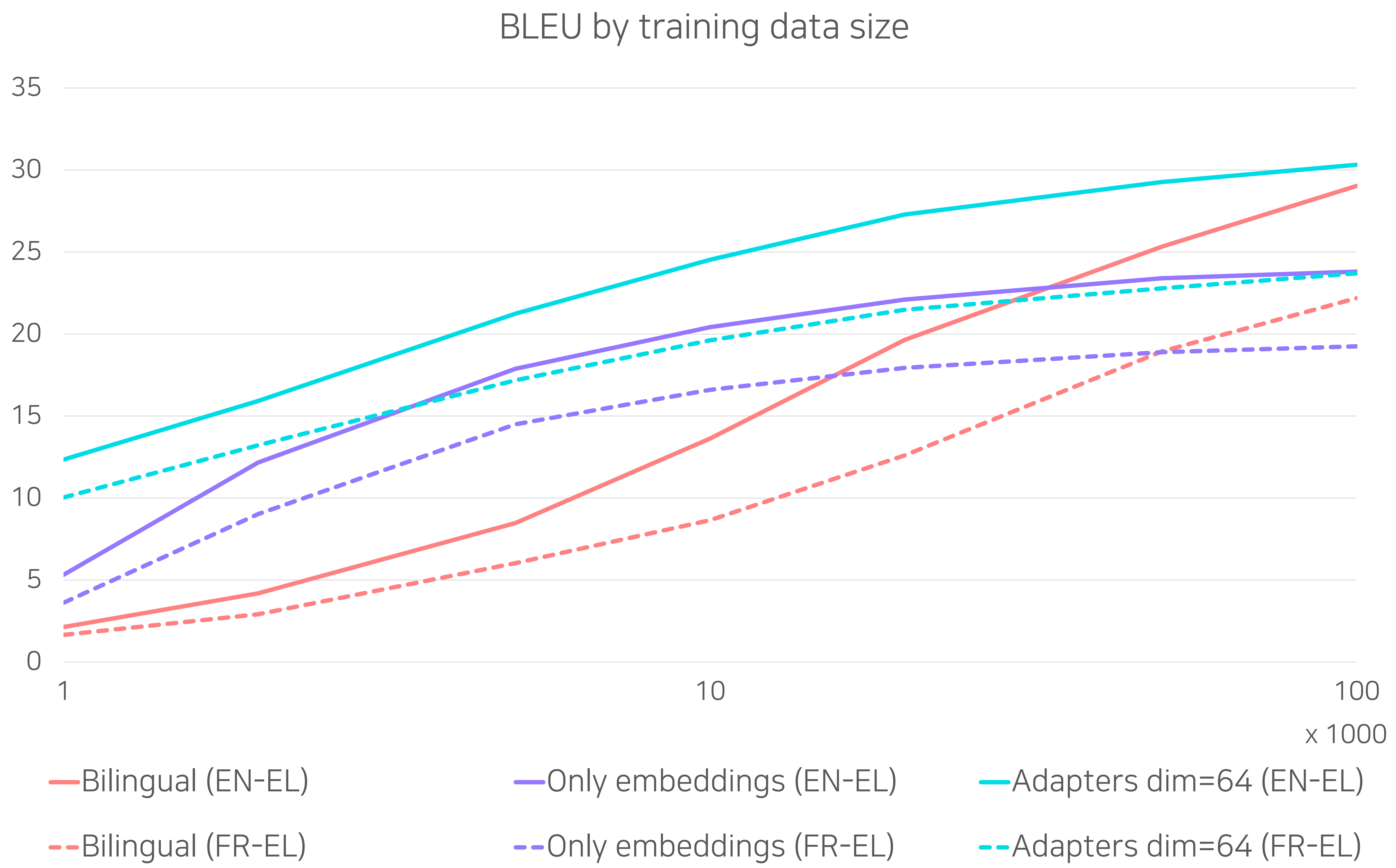
4.4 Results: data efficiency



4.4 Results: data efficiency



4.4 Results: data efficiency



4.4 Results: new source and target language

DEVIEW 2021

Model	BLEU
Bilingual baselines	14.9
Re-training + {EL, UK, SV, ID}	**22.0**

4.4 Results: new source and target language

Model	BLEU
Bilingual baselines	14.9
Re-training + {EL, UK, SV, ID}	**22.0**

Test-time combination of source and target params

Source model	Target model	BLEU
Only embeddings	Only embeddings	19.0
Only embeddings	Dec adapters + enc adapters last	21.2
Enc adapters + BT		20.7

4.5 Conclusion

- How to learn a new source or target language:
  - Create a new vocabulary for that language
  - Replace the source (resp. target) shared embeddings by lang-specific ones
  - Train the new embeddings plus some adapter modules

4.5 Conclusion

- How to learn a new source or target language:
  - Create a new vocabulary for that language
  - Replace the source (resp. target) shared embeddings by lang-specific ones
  - Train the new embeddings plus some adapter modules
- Zero-shot translation issue solved with tiny amounts of back-translation

4.5 Conclusion

DEVIEW 2021

- How to learn a new source or target language:
  - Create a new vocabulary for that language
  - Replace the source (resp. target) shared embeddings by lang-specific ones
  - Train the new embeddings plus some adapter modules
- Zero-shot translation issue solved with tiny amounts of back-translation
- Translation between 2 new languages by combining their lang-specific params

5. Learning new languages without parallel data

Multilingual Unsupervised Neural Machine Translation with Denoising Adapters

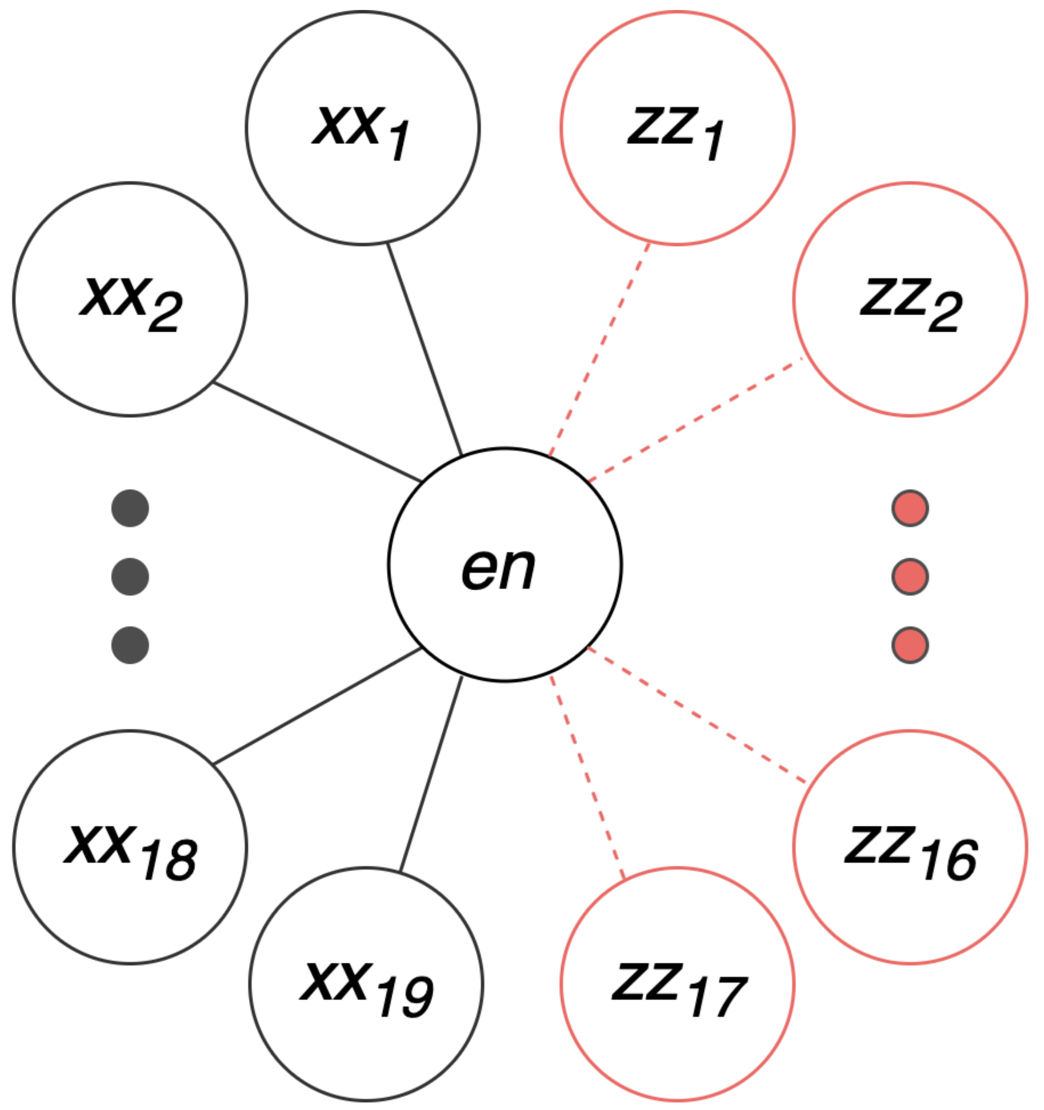
A. Ustun, A. Berard, L. Besacier and M. Galle

EMNLP 2021

5.1 Introduction

Unsupervised MNMT

- A single multilingual NMT model that can translate from/into multiple languages with *incomplete* parallel data
- Learn from both parallel data (EN↔XX_n) and monolingual data (ZZ_n)
- Add a new language (ZZ_n) to an existing MNMT model
  - Without retraining the full model
  - Using only monolingual data



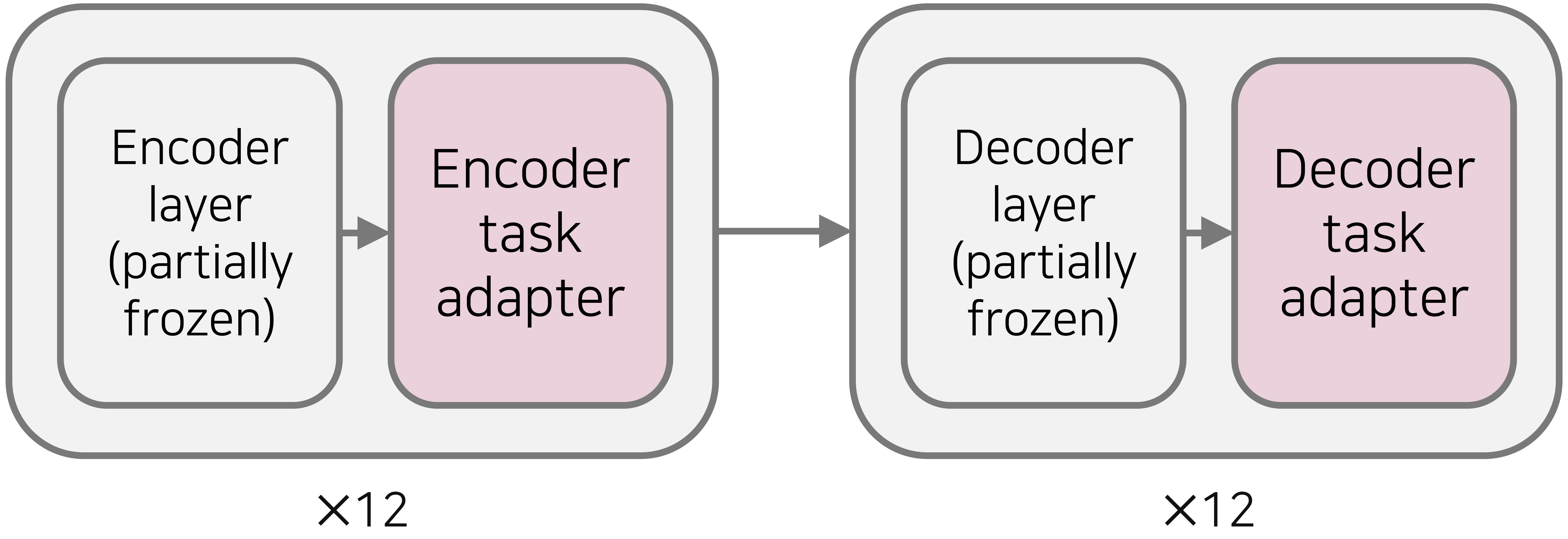
Overview of our multilingual UNMT setup. Figure adapted from Garcia et al. (2020)

5.2 mBART

Starting point: mBART50 (Tang et al., 2020)
A 50-language sequence-to-sequence model trained with a denoising objective.
Cannot do MT but can improve final performance when used as initialization.

Can be adapted to MT with parallel data, by:

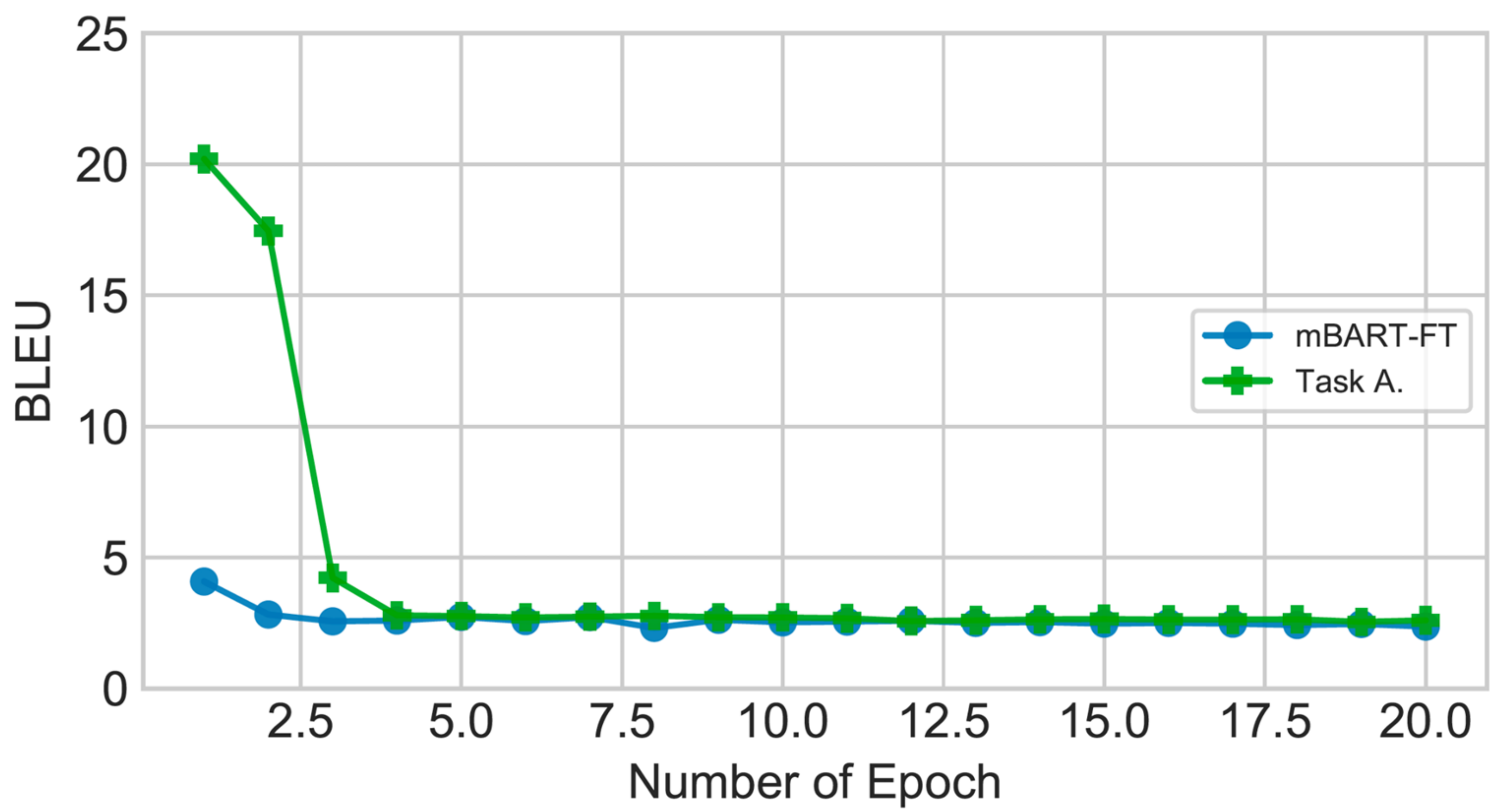
- Full fine-tuning
- Partial fine-tuning (e.g., cross-attention) + task adapters (Stickland et al., 2021)



5.2 mBART: unsupervised MNMT

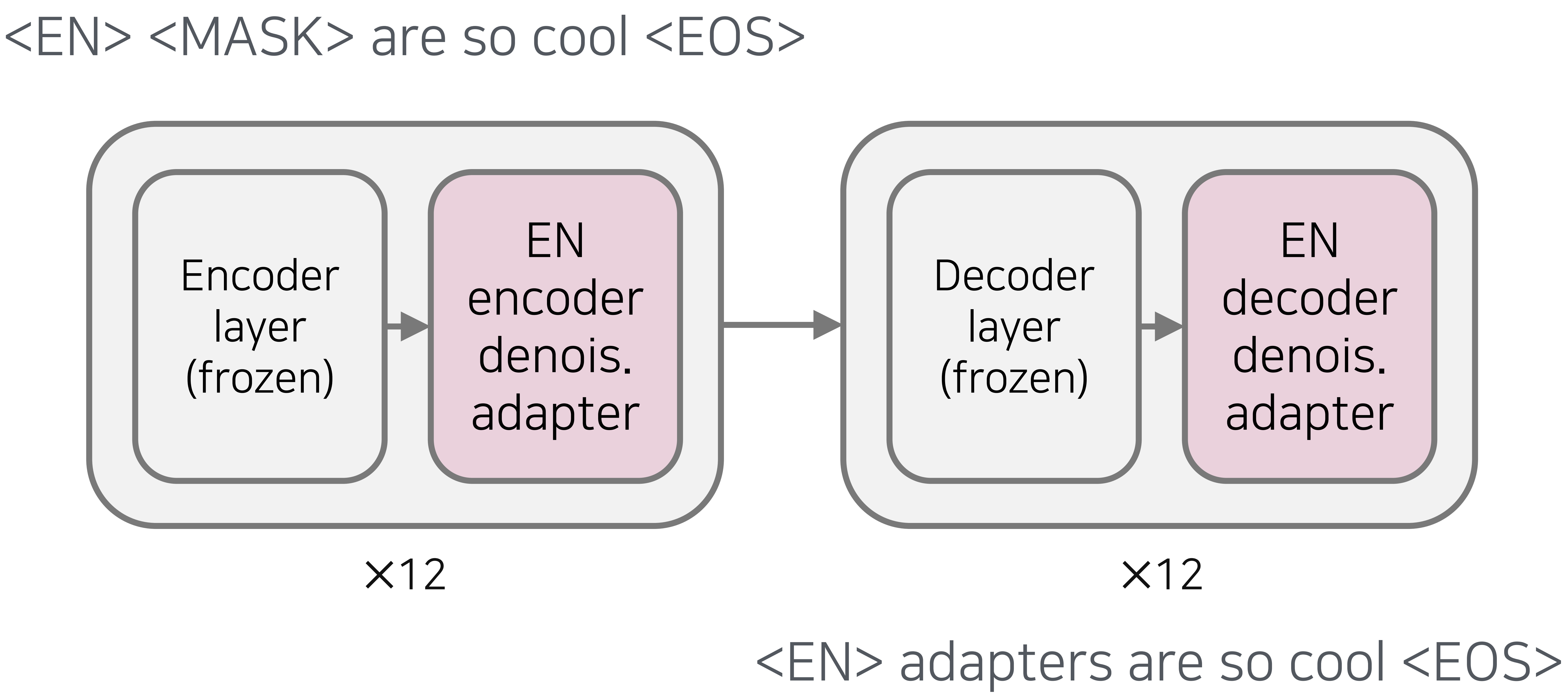
1. Fine-tune mBART with parallel data in a subset of N languages
2. Use it translate into/from the other (50 – N) languages

Issue: when fine-tuned (even partially), mBART quickly forgets about the other languages



Unsupervised EN → NL performance when fine-tuning mBART on 19 language pairs

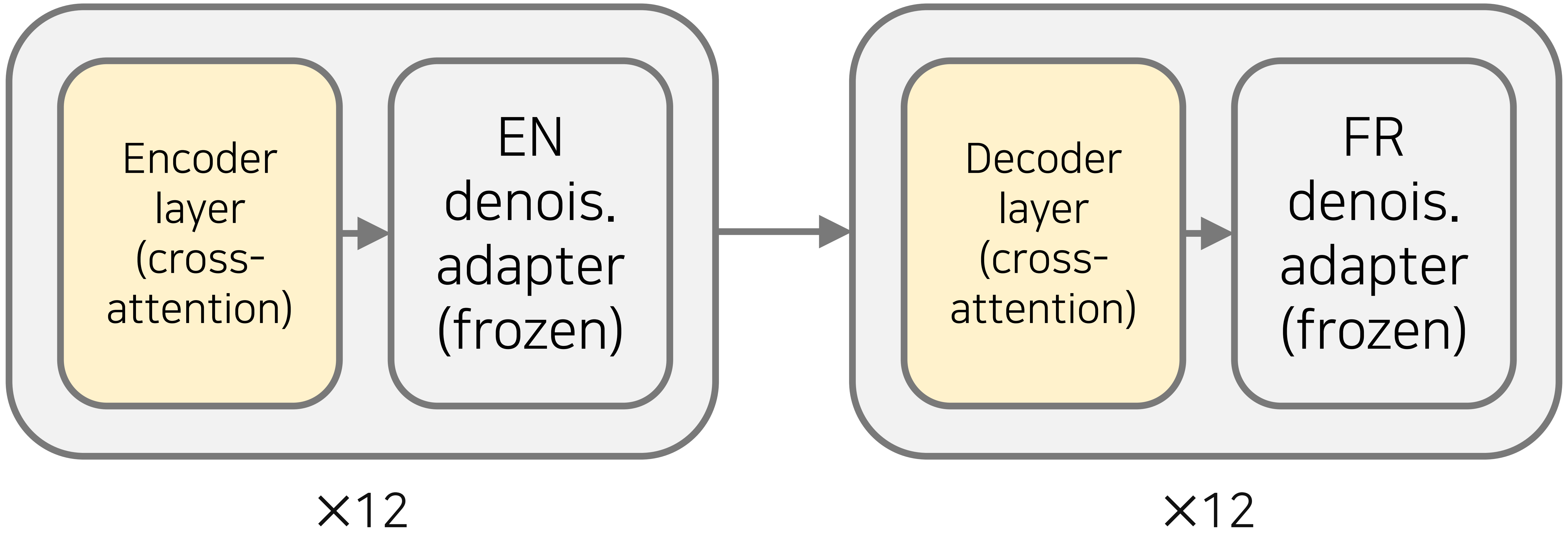
5.3 Technique



1. Train denoising adapters for all languages with monolingual data

5.3 Technique

<EN> adapters are so cool <EOS>

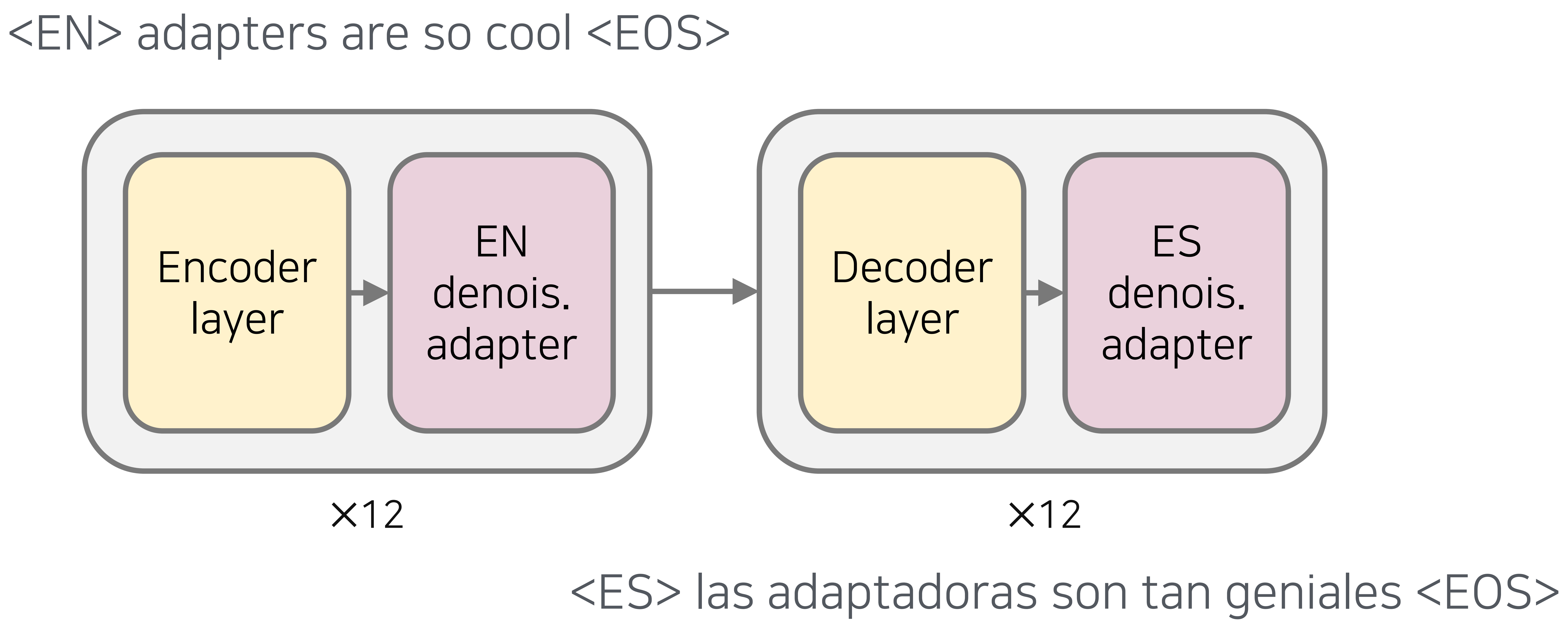


<FR> les adapteurs sont tellement cool <EOS>

2. Plug-in denoising adapters and fine-tune cross-attention on the available parallel data

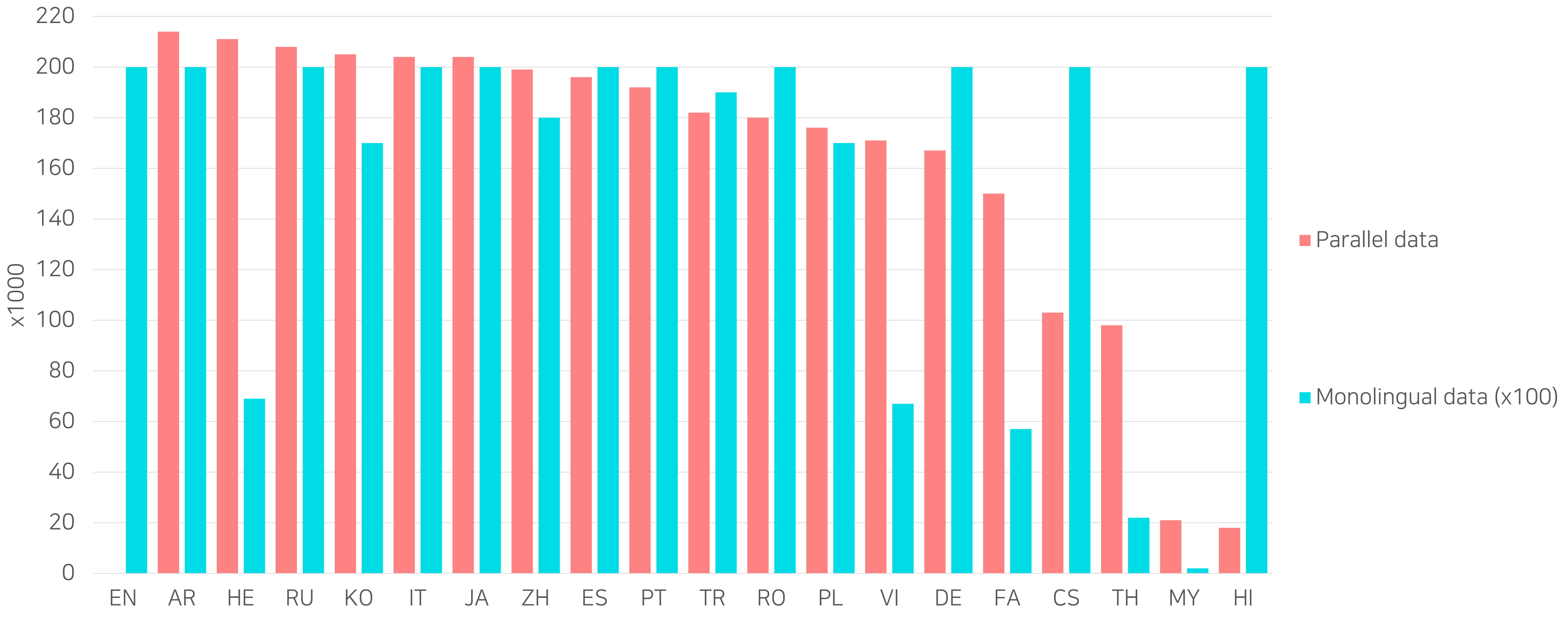
5.3 Technique

DEVIEW 2021



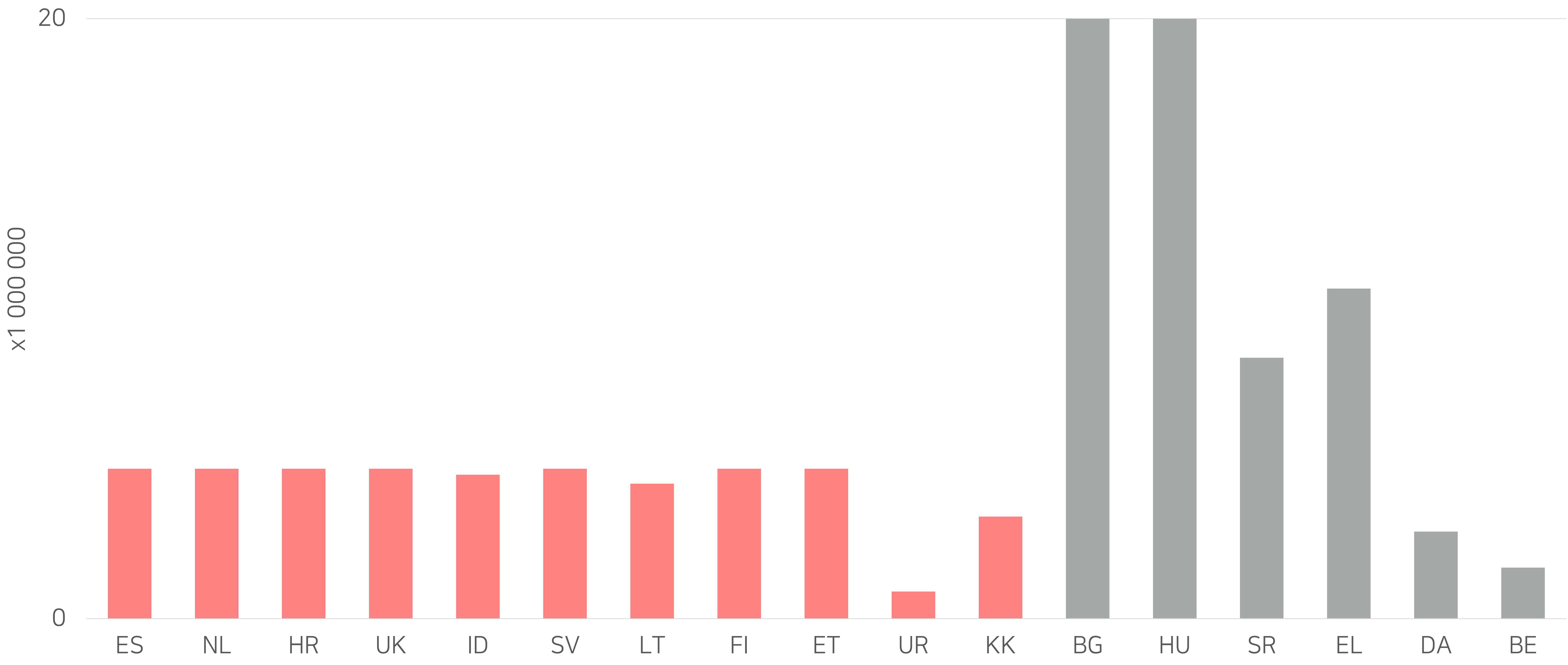
3. Plug-in the denoising adapters of an unsupervised language (ES) and translate

5.4 Experiments



20 languages with both English-centric parallel data (TED Talks)
and monolingual data (Wikipedia and News)

5.4 Experiments



17 languages with only monolingual data

6 languages unknown by mBART

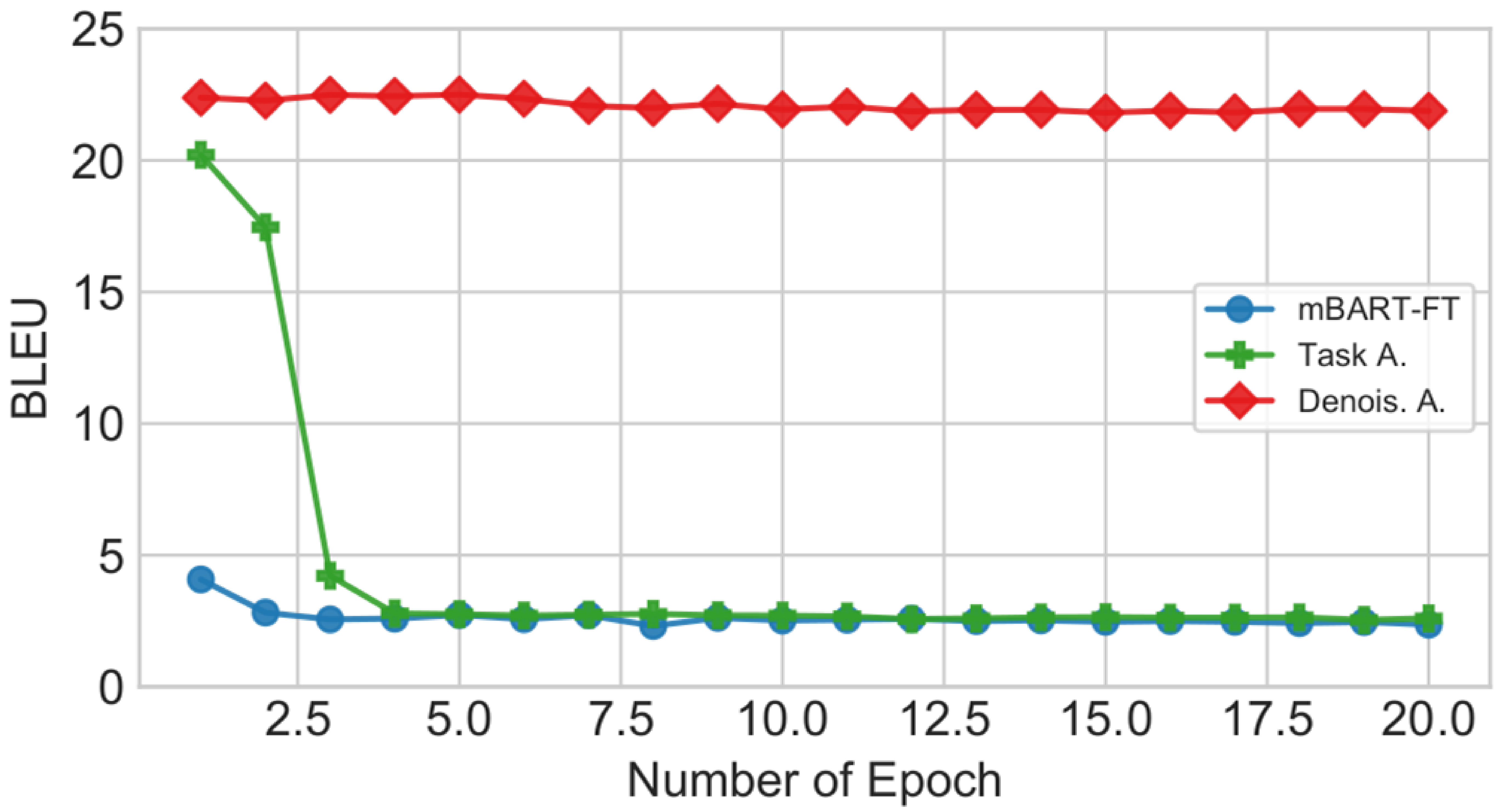
5.4 Experiments

DEVIEW 2021

Baselines

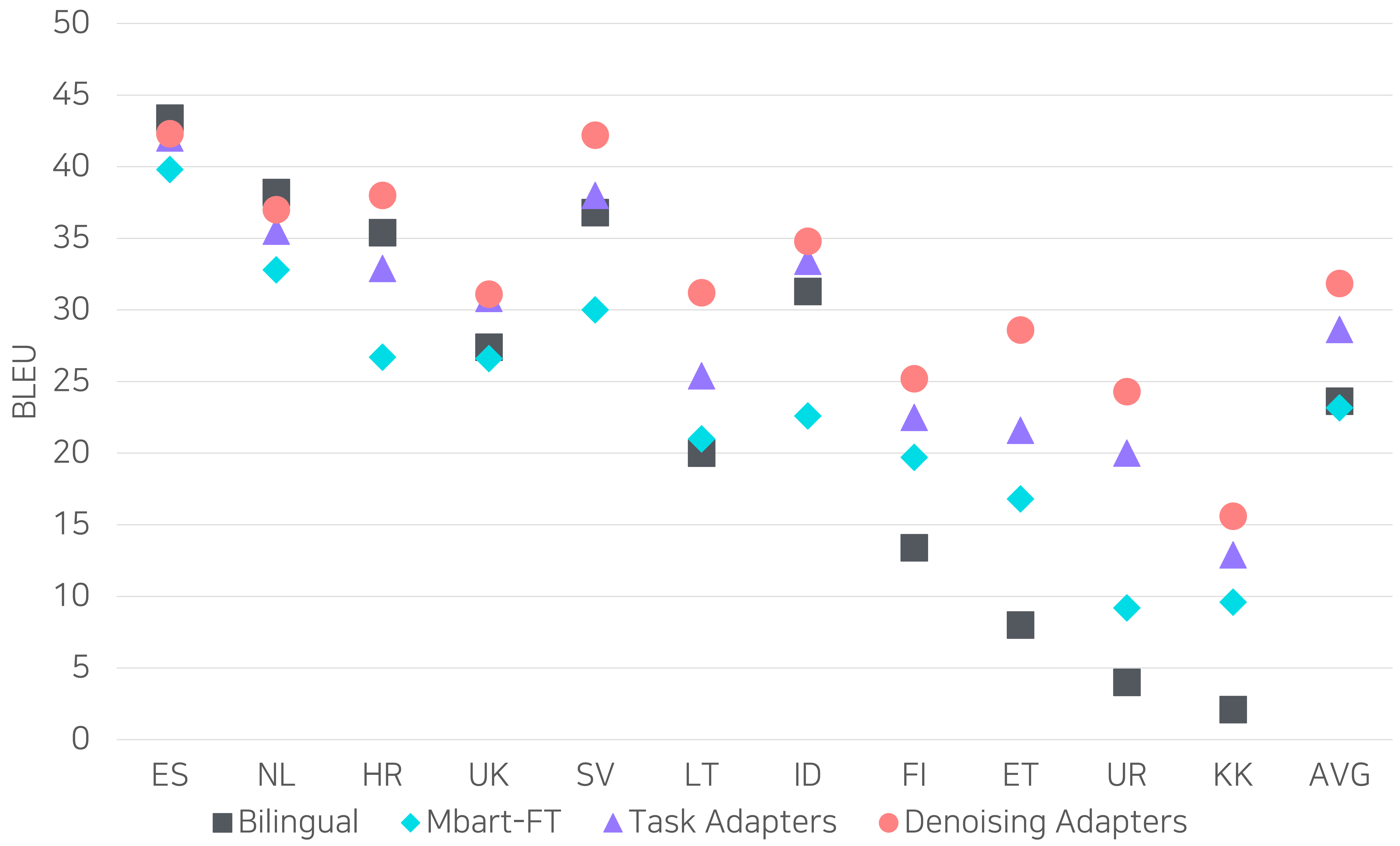
- Bilingual supervised models (Bilingual)
- mBART full fine-tuning (mBART-FT)
- Fine-tune cross-attention + task adapters (Task Adapters)
- Same models with back-translation (+ BT)

5.5 Results



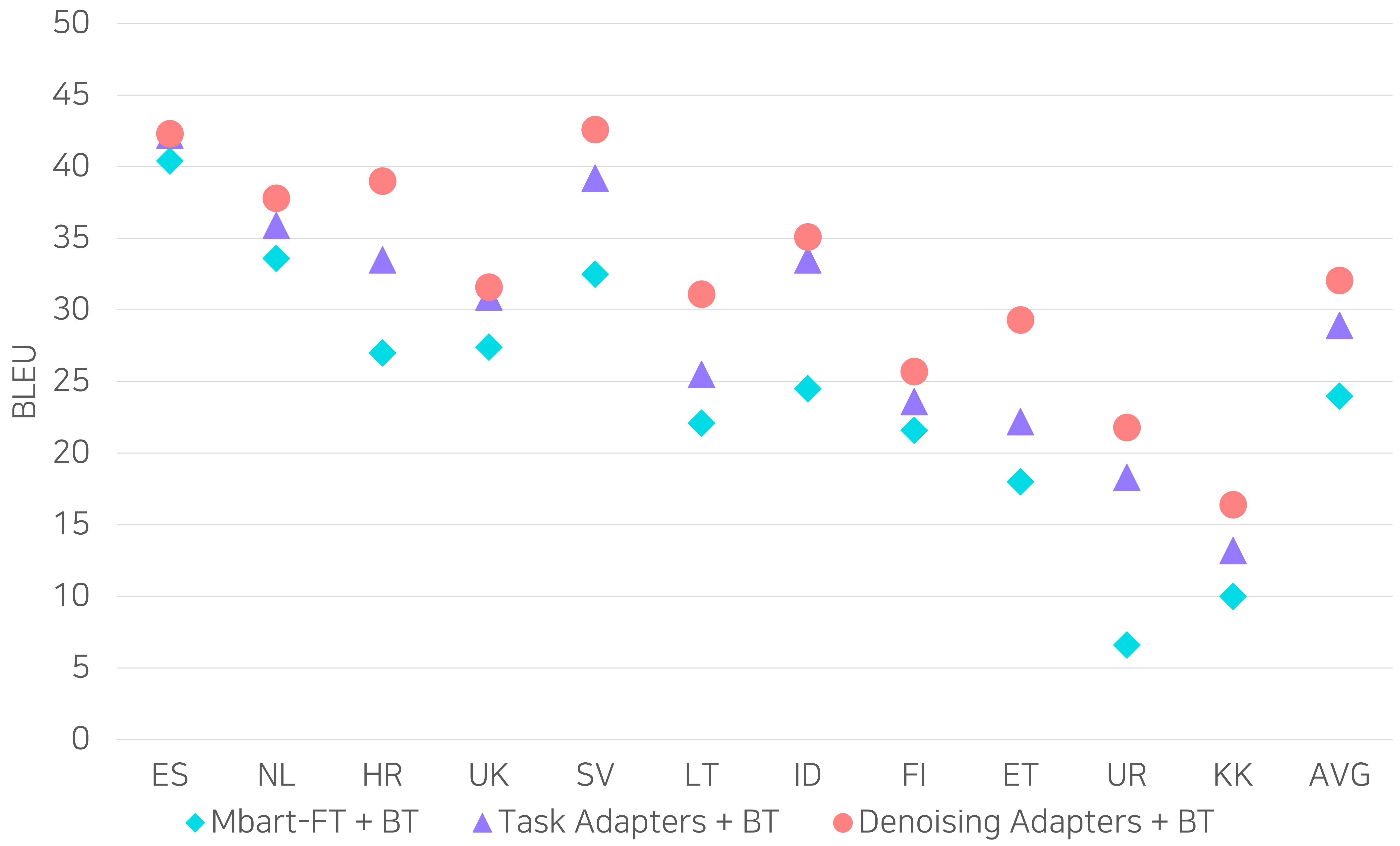
Unsupervised **EN → NL** performance when fine-tuning mBART on 19 language pairs

5.5 Results



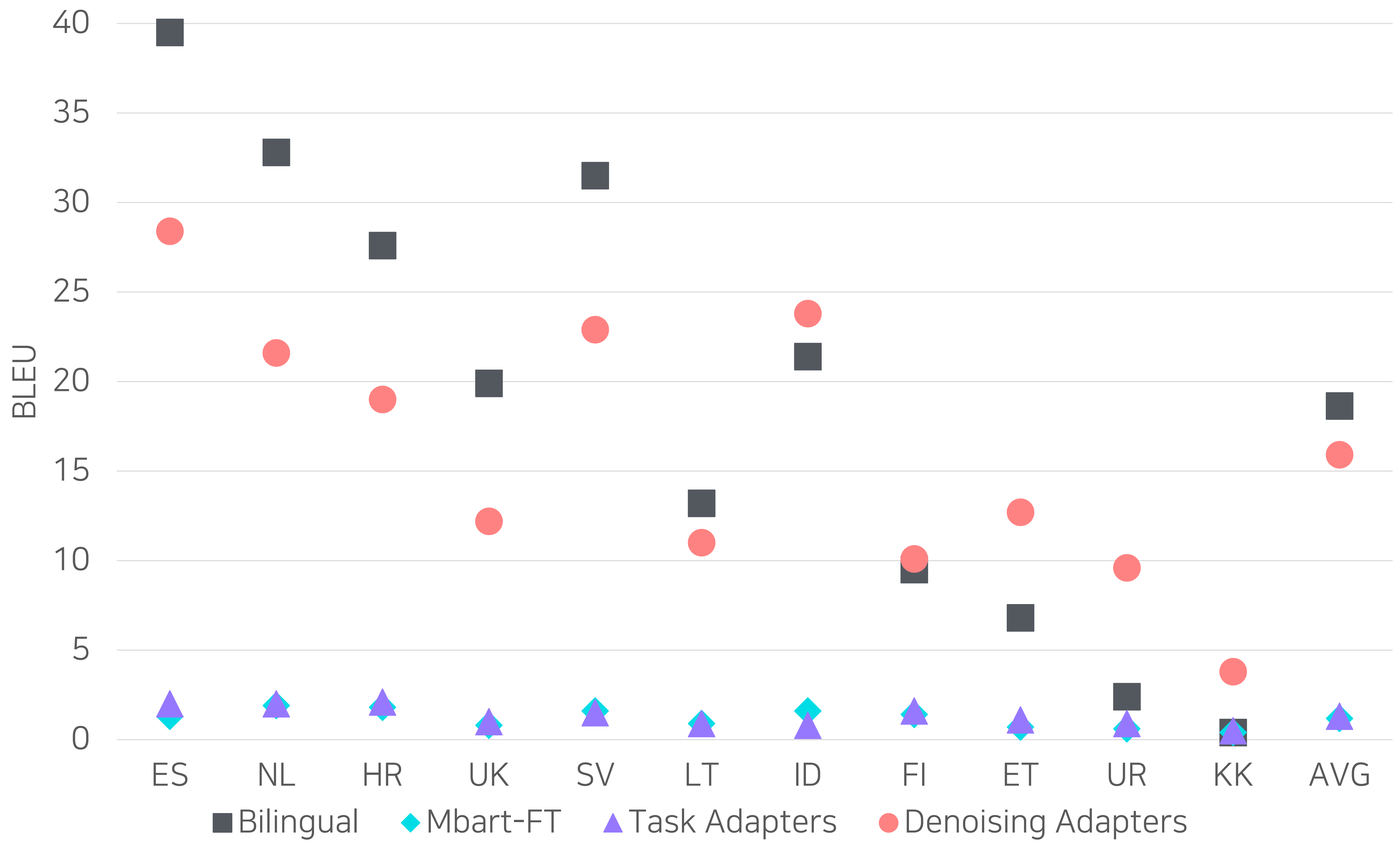
Unsupervised translation **into English**

5.5 Results



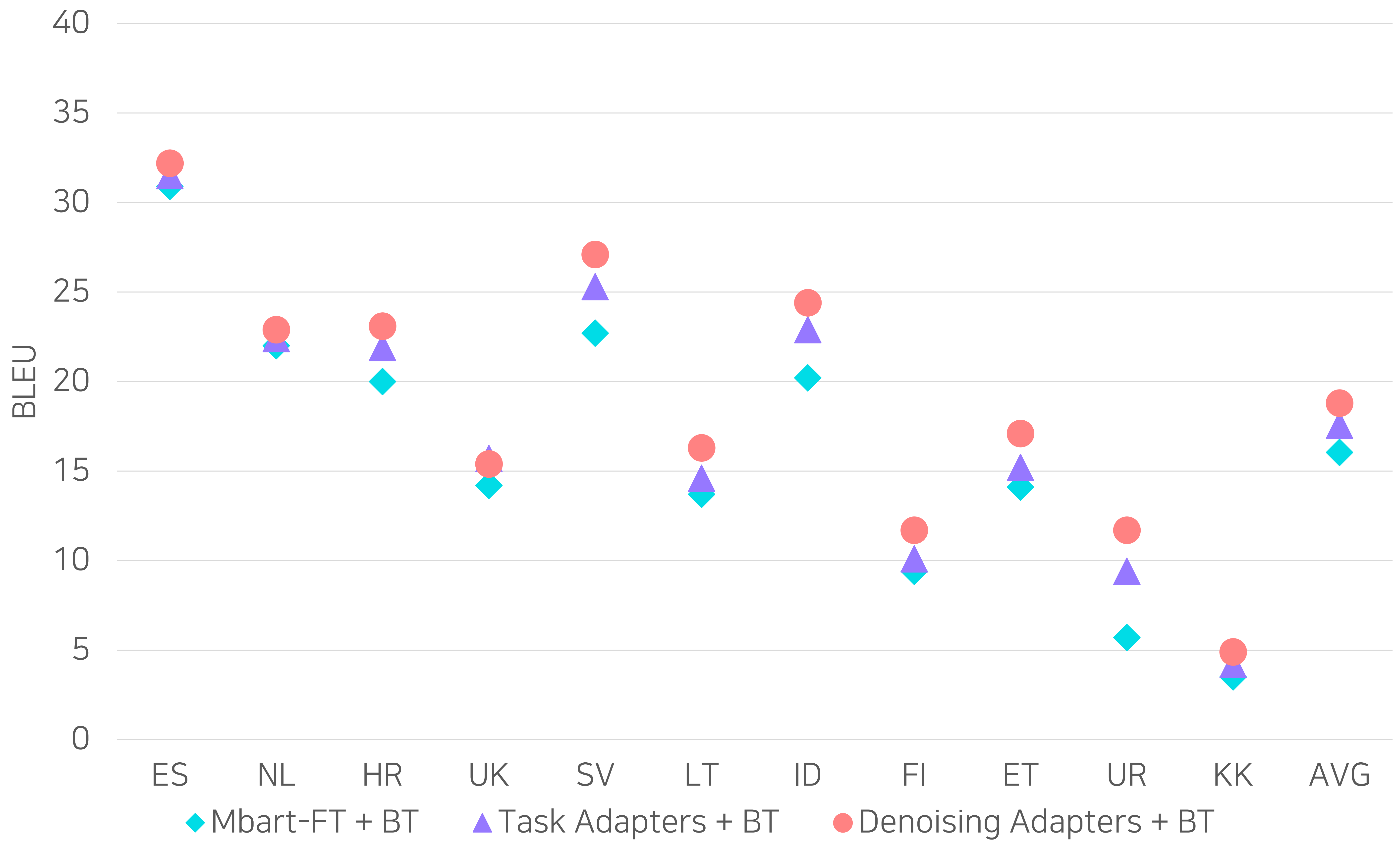
Unsupervised translation **into English** with back-translation

5.5 Results



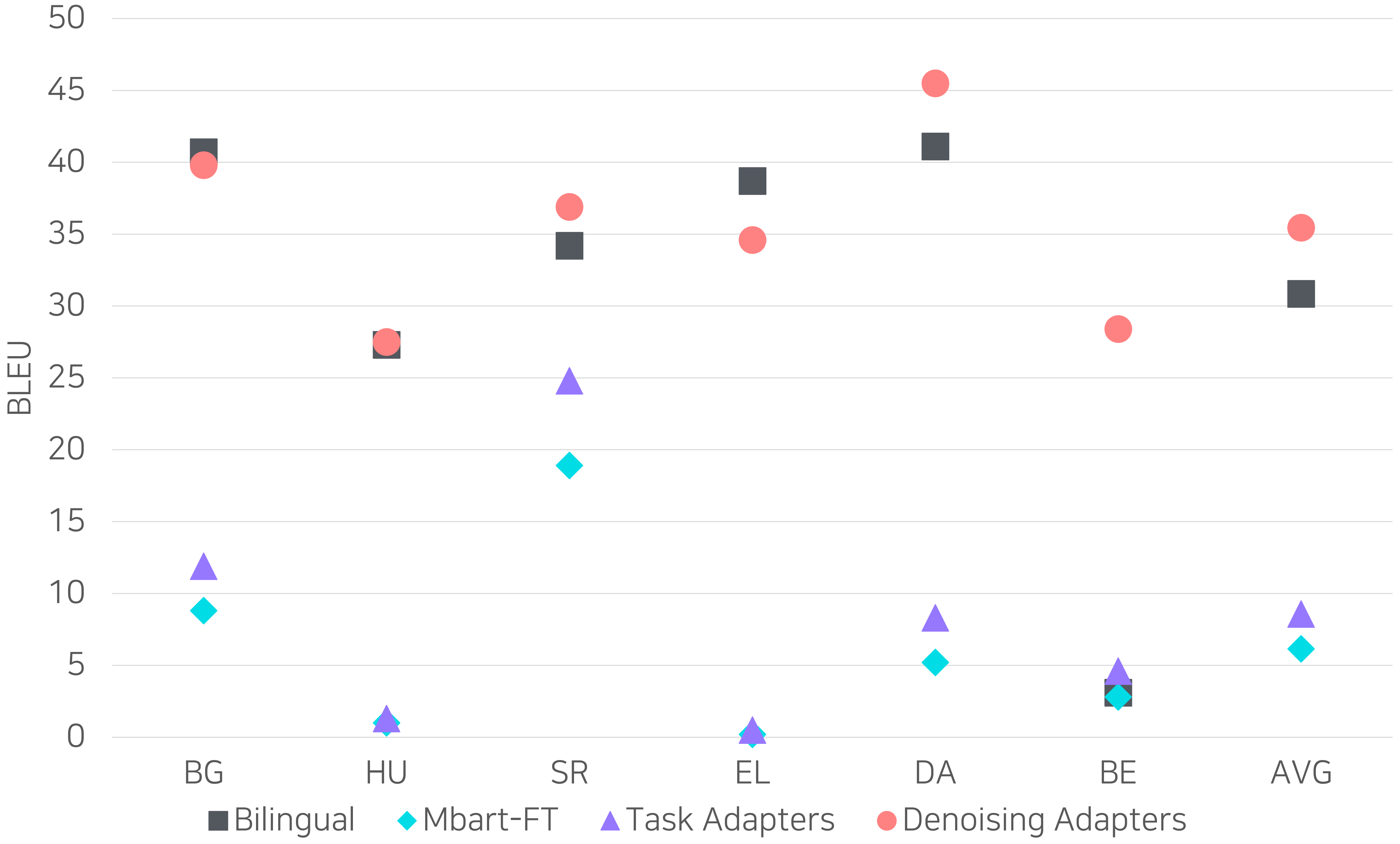
Unsupervised translation **from English**

5.5 Results



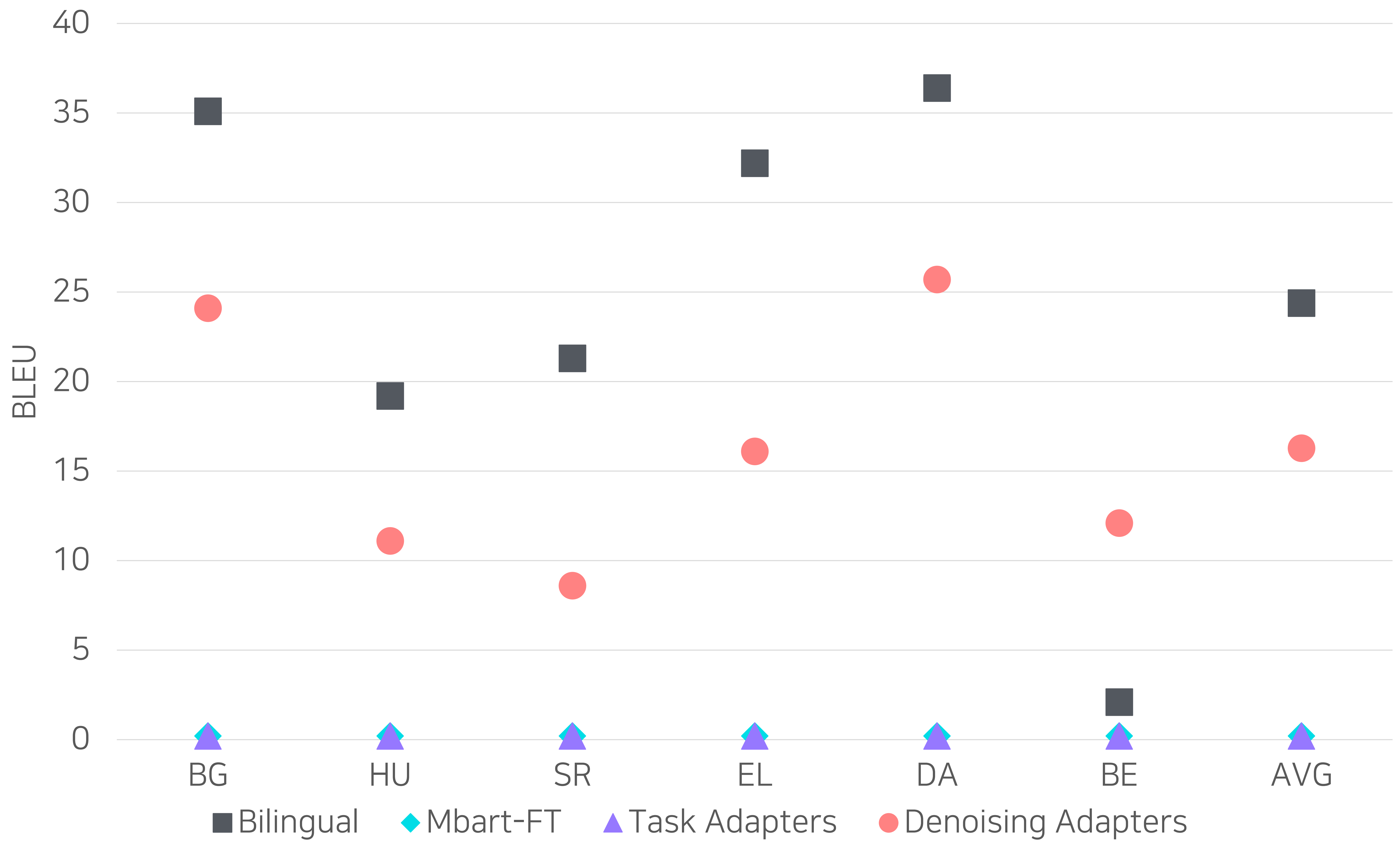
Unsupervised translation **from English** with back-translation

5.5 Results



Unsupervised translation **into English** from languages unseen by mBART

5.5 Results



Unsupervised translation **from English** into languages unseen by mBART

5.6 Conclusion

- Adapting mBART50 to multilingual NMT comes with challenges
  - Multilingual parallel data is needed
  - Poor performance for languages NOT covered by parallel data
- We propose denoising adapters, monolingually-trained adapter layers to leverage monolingual data for unsupervised MT
- Our experiments on a large set of languages show the effectiveness of denoising adapters with and without BT
- We also show that denoising adapters can be used to add languages unknown by mBART

Takeaways

Multilingual NMT is appealing in production but it comes with challenges

- Larger models are slower at inference
- Need for parameter-efficient domain and language adaptation

Towards continual learning in Multilingual NMT

- Learn new languages efficiently
- Learn new languages without parallel data

Thank You