291K
Deep Learning for Machine Translation
Pre-training Language Model for MT

Lei Li
UCSB

10/25/2021

What happens?

Does BERT matter in NMT?

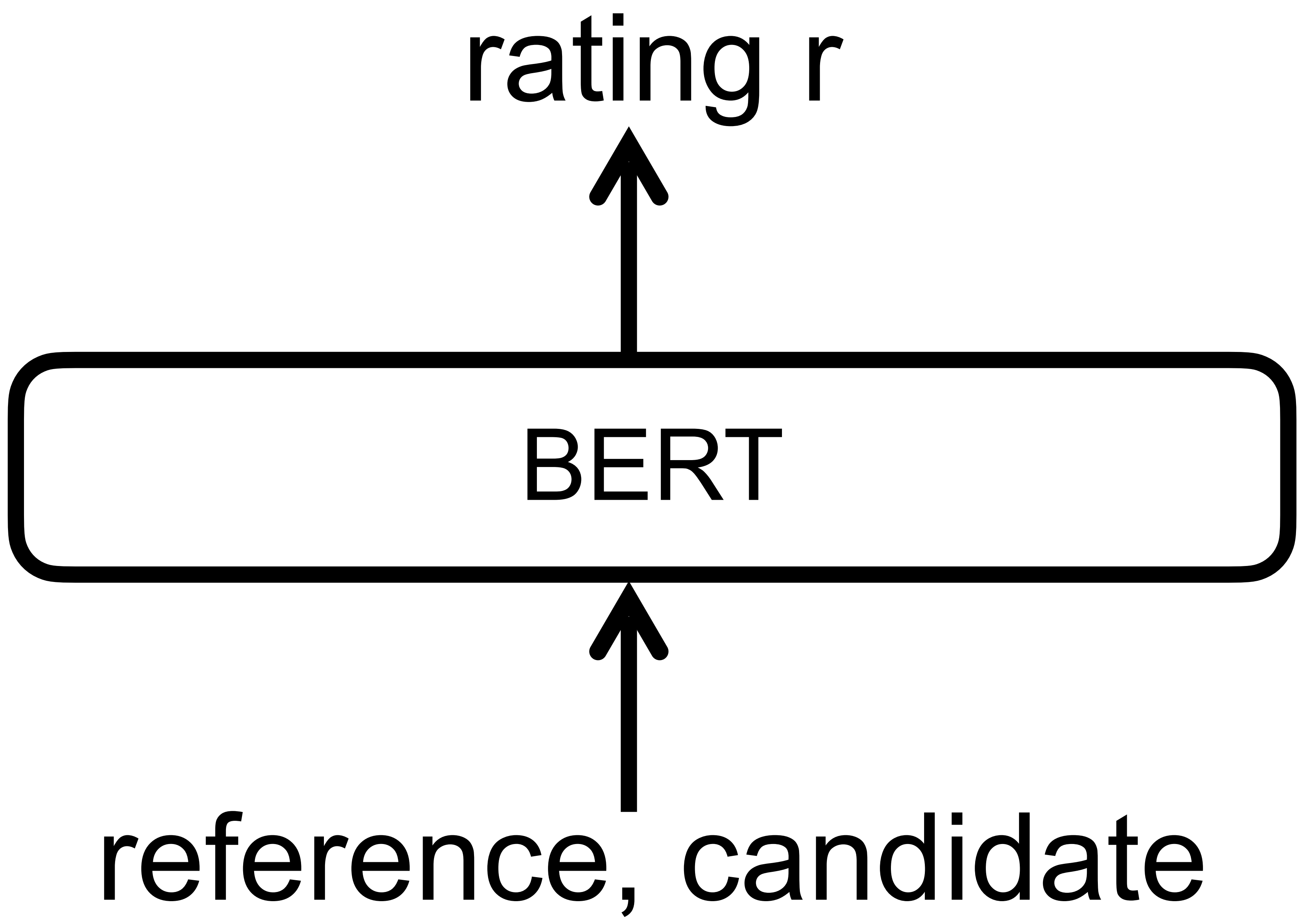
Outline

- Learned Metrics for MT using BERT
- BERT NMT Distillation
- BERT NMT Fusion

Learned Metric for MT using BERT

BLEURT

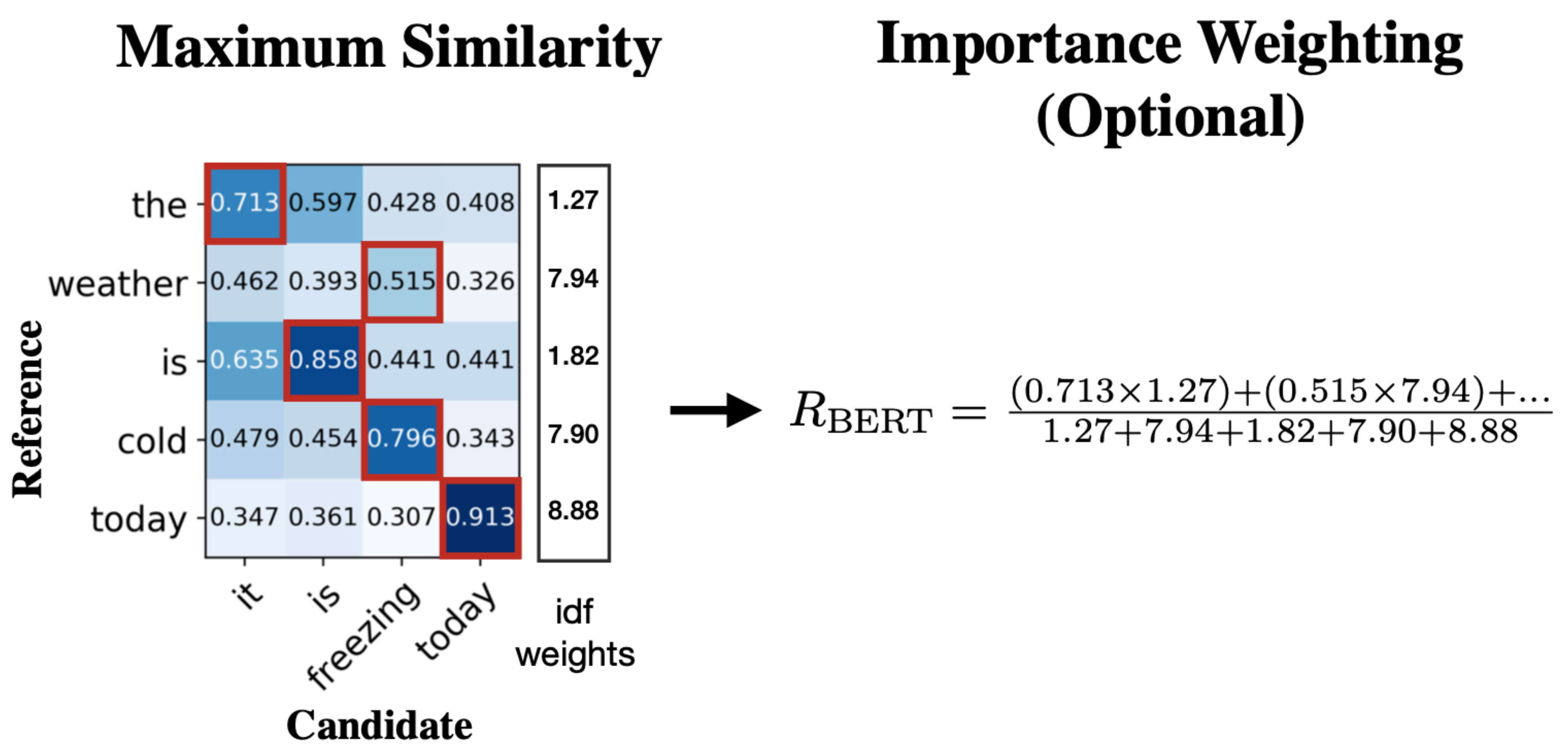
- Input reference y* and candidate y into BERT, and directly predict rating r
- With model pre-training



Sellam et al. BLEURT: Learning Robust Metrics for Text Generation, 2020

BERTScore

- Idea:
  - Use a pre-trained BERT to compute contextual embeddings for each token in reference sentence and candidate sentence.
  - Compute precision, recall for every token based on embedding (instead of matching on the surface level).
- Recall: $R(y^*, y) = \dfrac{\sum_{i=1}^{|y^*|} \max_{j=1}^{|y|} f(y^*)_i^T \cdot f(y)_j^T}{|y^*|}$
- Precision: $P(y^*, y) = \dfrac{\sum_{j=1}^{|y|} \max_{i=1}^{|y^*|} f(y^*)_i^T \cdot f(y)_j}{|y|}$
- $F(y^*, y) = \dfrac{P \cdot R}{P + R}$



- can be weighted by IDF (inverse document frequency), if a word appears in many sentences, it is less important. $\text{idf}(w) = \log \dfrac{\#\text{sentences}}{\#\text{sentences contain } w}$

Zhang et al. BERTScore: Evaluating Text Generation with BERT. 2020

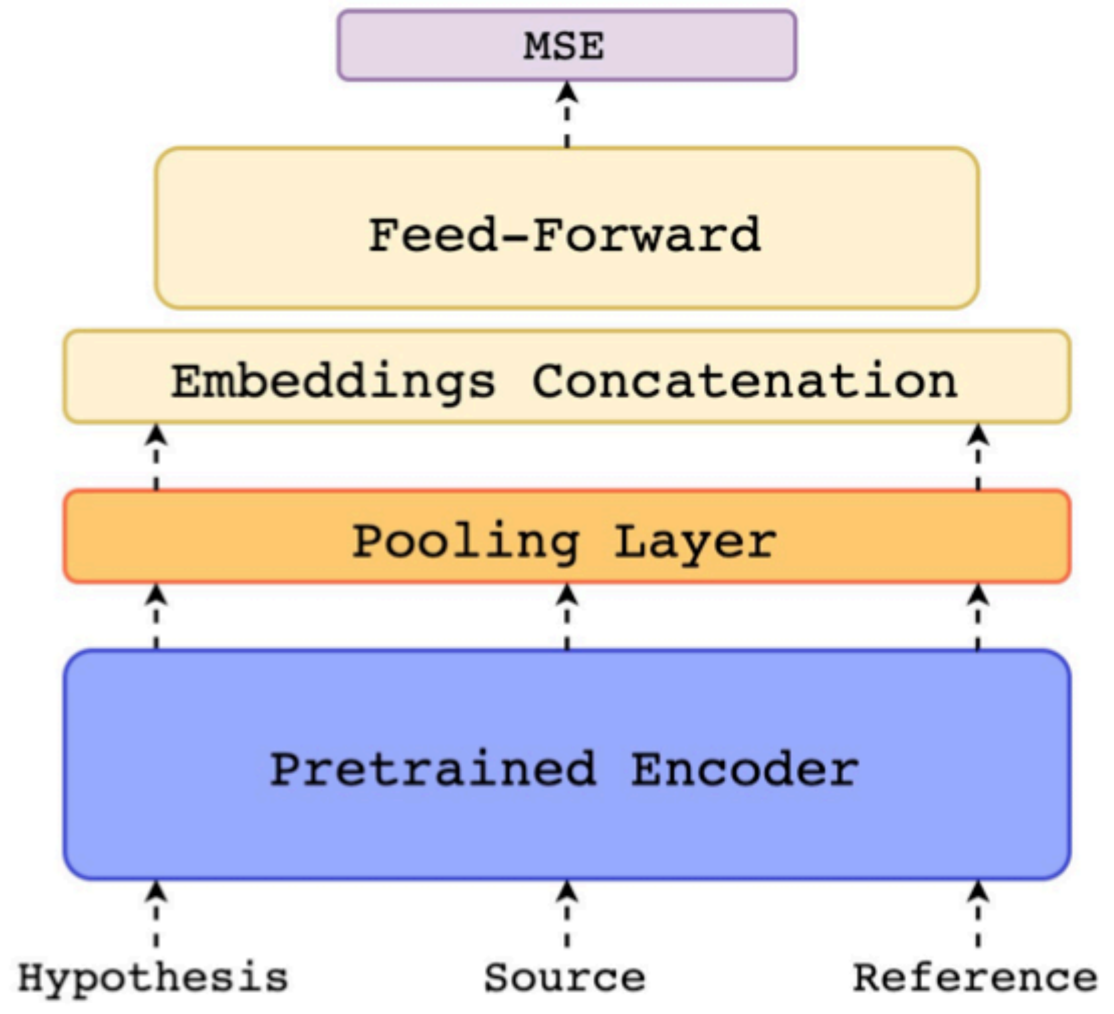
Correlation of BERTScore and Human evaluation for WMT18

Metric	en↔cs (5/5)	en↔de (16/16)	en↔et (14/14)	en↔fi (9/12)	en↔ru (8/9)	en↔tr (5/8)	en↔zh (14/14)
BLEU	.970/**.995**	.971/**.981**	**.986/.975**	.973/**.962**	.979/**.983**	**.657**/.826	.978/.947
ITER	.975/.915	.990/**.984**	.975/**.981**	**.996/.973**	.937/.975	**.861**/.865	.980/ –
RUSE	.981/ –	.997/ –	**.990**/ –	.991/ –	**.988**/ –	**.853**/ –	**.981**/ –
YiSi-1	.950/**.987**	.992/**.985**	.979/**.979**	.973/.940	**.991/.992**	**.958/.976**	.951/**.963**
P_{BERT}	.980/**.994**	**.998/.988**	**.990/.981**	.995/.957	.982/**.990**	**.791/.935**	.981/.954
R_{BERT}	**.998/.997**	.997/**.990**	.986/**.980**	**.997/.980**	**.995/.989**	.054/.879	**.990/.976**
F_{BERT}	**.990/.997**	**.999/.989**	.990/**.982**	**.998/.972**	**.990**/.990	**.499**/.908	**.988**/.967
F_{BERT} (idf)	.985/**.995**	**.999/.990**	**.992/.981**	.992/**.972**	**.991/.991**	**.826/.941**	**.989/.973**

Table 1: Absolute Pearson correlations with system-level human judgments on WMT18. For each language pair, the left number is the to-English correlation, and the right is the from-English. We bold correlations of metrics not significantly outperformed by any other metric under Williams Test for that language pair and direction. The numbers in parenthesis are the number of systems used for each language pair and direction.

COMET

- Use source sentence x, reference y*, candidate y, to learn a rating function
  - $x = [h; r; h \odot s; h \odot r; |h - s|; |h - r|]$, where h is embedding for y
- COMET-rank: instead of rating, learn a ranking for candiate y+ and y- given source sentence x and reference y*



Rei et al. COMET: A Neural Framework for MT Evaluation. 2020

Correlation between COMET and Human Evaluation

Table 1: Kendall's Tau (τ) correlations on language pairs with English as source for the WMT19 Metrics DARR corpus. For BERTSCORE we report results with the default encoder model for a complete comparison, but also with XLM-RoBERTa (base) for fairness with our models. The values reported for YiSi-1 are taken directly from the shared task paper (Ma et al., 2019).

Metric	en-cs	en-de	en-fi	en-gu	en-kk	en-lt	en-ru	en-zh
BLEU	0.364	0.248	0.395	0.463	0.363	0.333	0.469	0.235
CHRF	0.444	0.321	0.518	0.548	0.510	0.438	0.548	0.241
YISI-1	0.475	0.351	0.537	0.551	0.546	0.470	0.585	0.355
BERTSCORE (default)	0.500	0.363	0.527	0.568	0.540	0.464	0.585	0.356
BERTSCORE (xlmr-base)	0.503	0.369	0.553	0.584	0.536	0.514	0.599	0.317
COMET-HTER	0.524	0.383	0.560	0.552	0.508	0.577	0.539	0.380
COMET-MQM	0.537	0.398	0.567	0.564	0.534	0.574	**0.615**	0.378
COMET-RANK	**0.603**	**0.427**	**0.664**	**0.611**	**0.693**	**0.665**	0.580	**0.449**

Rei et al. COMET: A Neural Framework for MT Evaluation. 2020

Correlation between COMET and Human Evaluation

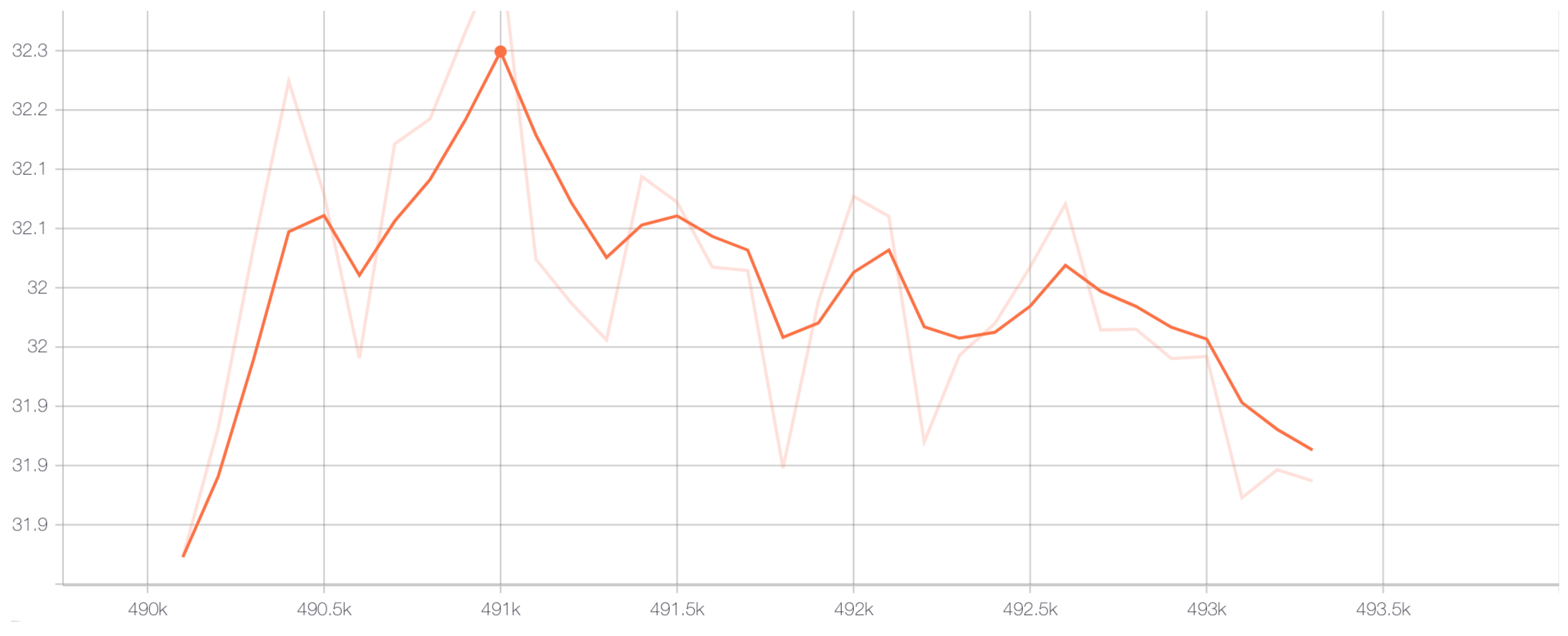
Table 2: Kendall's Tau (τ) correlations on language pairs with English as a target for the WMT19 Metrics DARR corpus. As for BERTSCORE, for BLEURT we report results for two models: the base model, which is comparable in size with the encoder we used and the large model that is twice the size.

Metric	de-en	fi-en	gu-en	kk-en	lt-en	ru-en	zh-en
BLEU	0.053	0.236	0.194	0.276	0.249	0.177	0.321
CHRF	0.123	0.292	0.240	0.323	0.304	0.115	0.371
YISI-1	0.164	0.347	0.312	**0.440**	0.376	0.217	0.426
BERTSCORE (default)	0.190	0.354	0.292	0.351	0.381	**0.221**	0.432
BERTSCORE (xlmr-base)	0.171	0.335	0.295	0.354	0.356	0.202	0.412
BLEURT (base-128)	0.171	0.372	0.302	0.383	0.387	0.218	0.417
BLEURT (large-512)	0.174	0.374	0.313	0.372	0.388	0.220	0.436
COMET-HTER	0.185	0.333	0.274	0.297	0.364	0.163	0.391
COMET-MQM	**0.207**	0.343	0.282	0.339	0.368	0.187	0.422
COMET-RANK	0.202	**0.399**	**0.341**	0.358	**0.407**	0.180	**0.445**

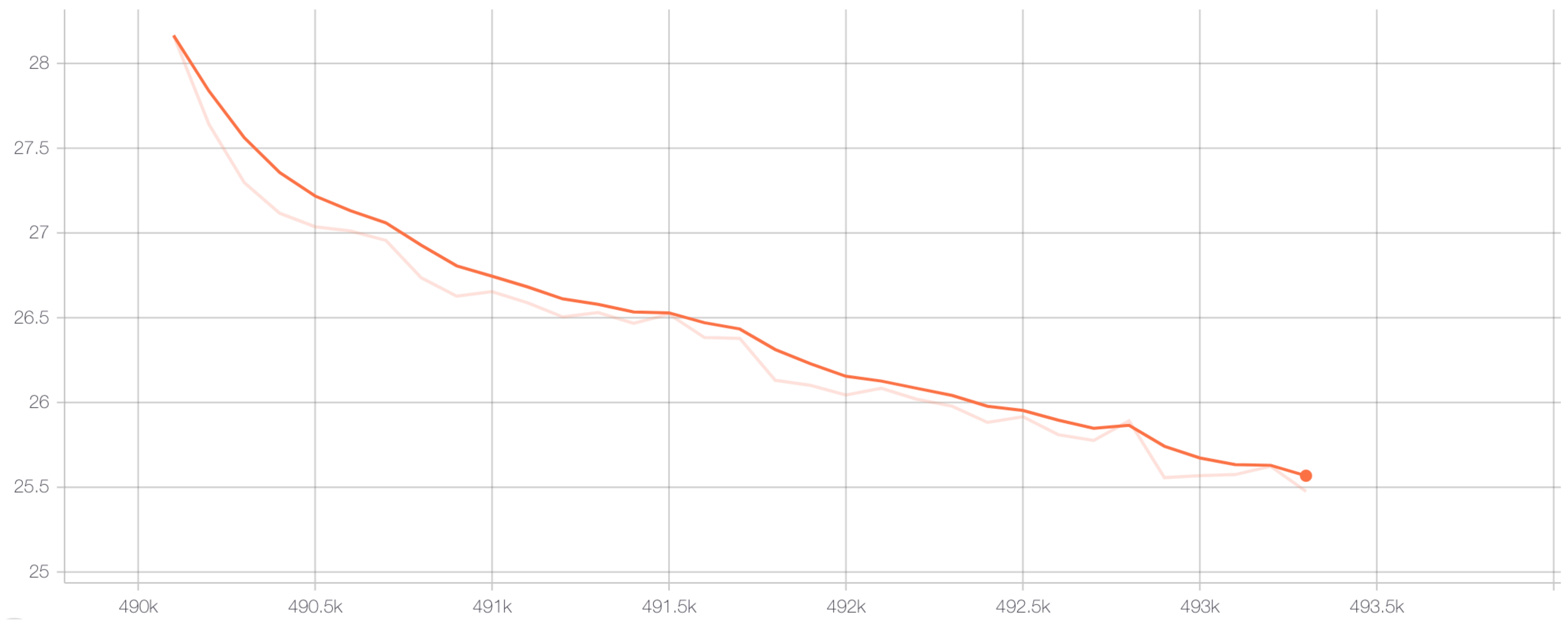
Rei et al. COMET: A Neural Framework for MT Evaluation. 2020

BERT NMT Distillation

BERT Initialization and Fine-tuning



Performance on fine-tuning NMT



Performance on other BERT tasks

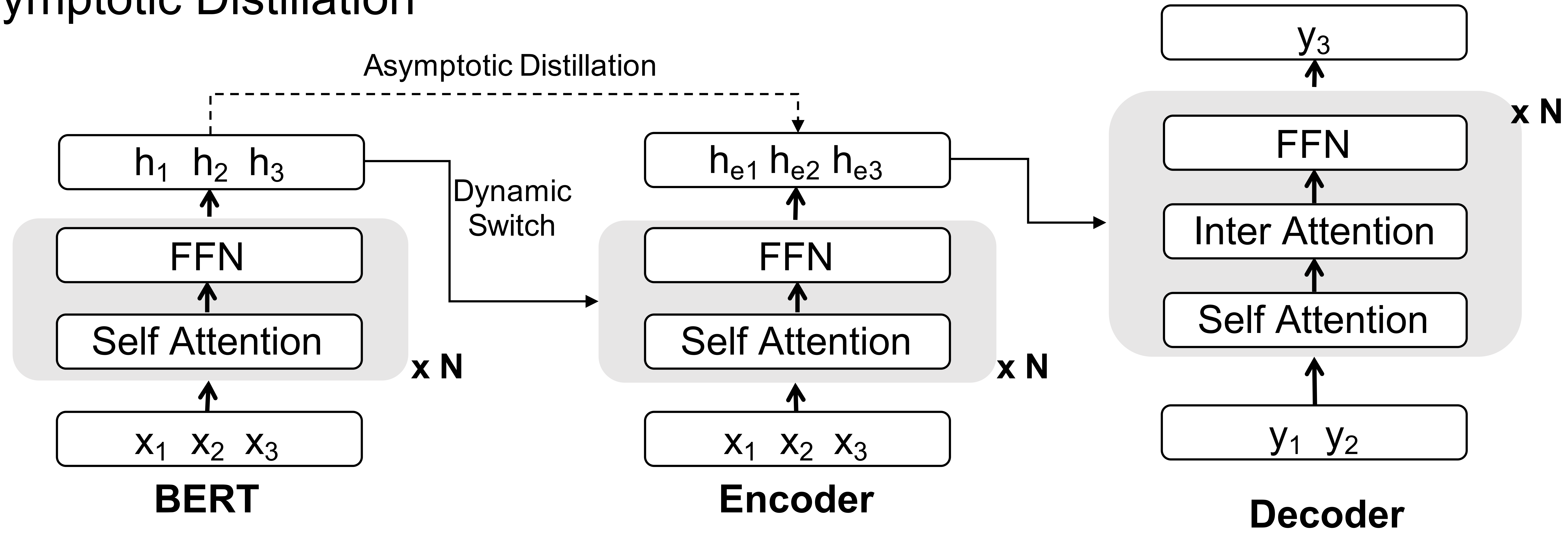
Why simply incorporating BERT does not work as expectation

- Fine-tuning leads to performance degradation on the original task
- The situation is more severe on NMT fine-tuning
  - High capacity of baseline needs much updating
  - Updating to much makes the model forgets its universal knowledge from pre-training

Towards Making Most of BERT for NMT, [Yang et al AAAI 2020]

Not tuning too much

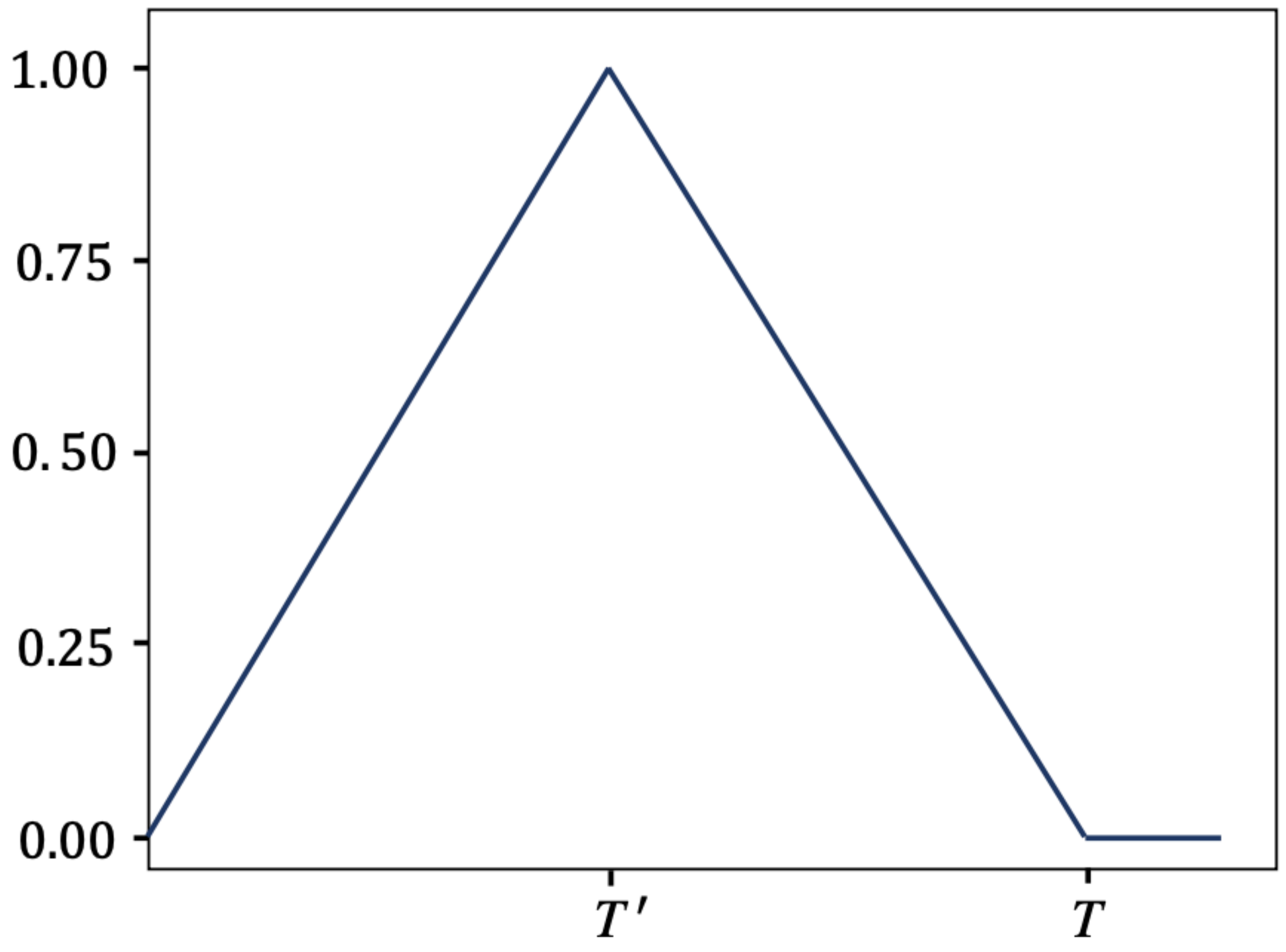
- Concerted training framework
  - Rate-scheduled Learning
  - Dynamic Switch
  - Asymptotic Distillation



Towards Making Most of BERT for NMT, [Yang et al AAAI 2020]

Not tuning too much

- Rate-scheduled Learning rate
  - Gradually increase the learning rate of BERT parameters from 0 to 1
  - Then, decrease the learning rate of BERT parameters from 1 to 0
  - Keep the BERT parameters frozen



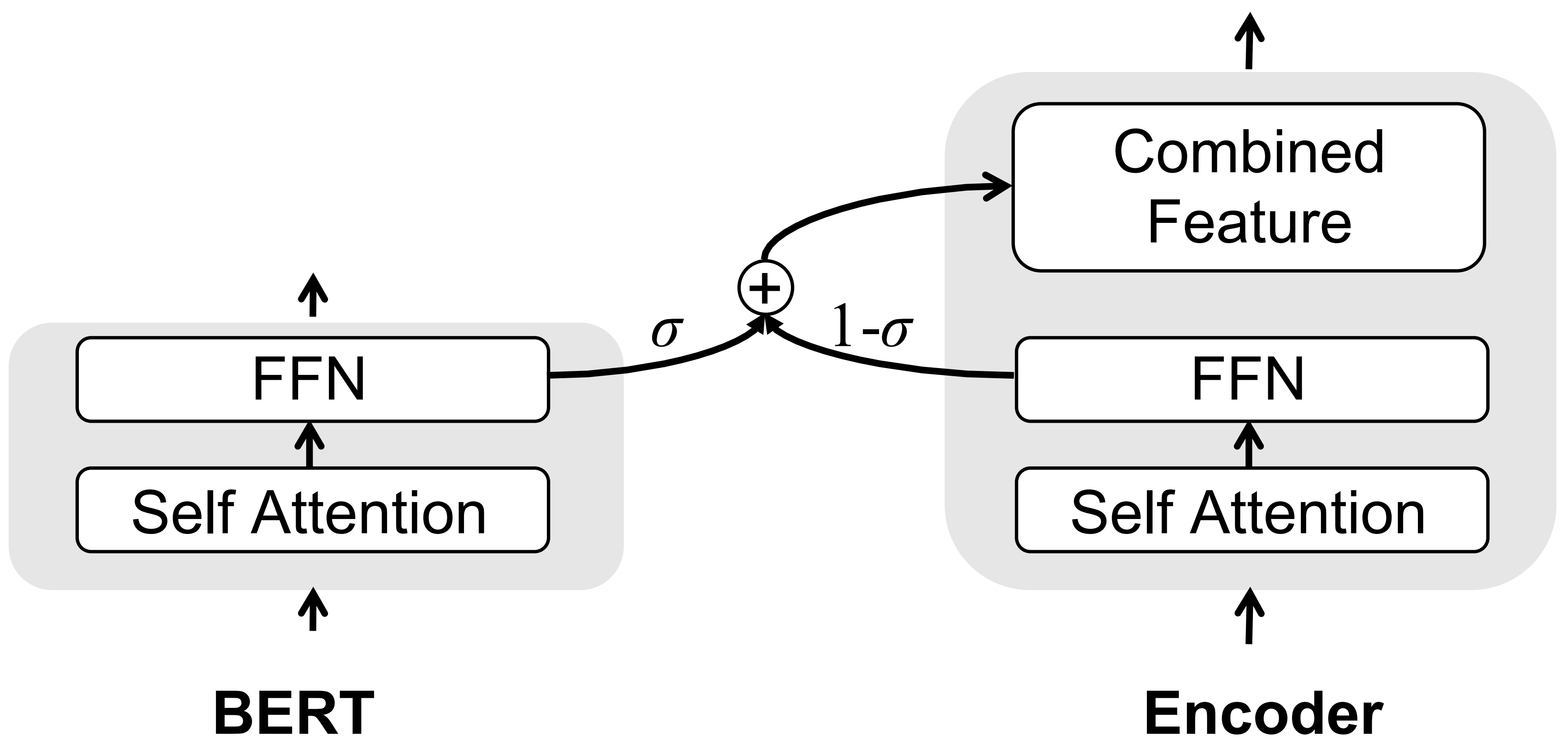
Learning rate scalar for BERT parameter

Rate-scheduled learning rate is actually a trade off between fine-tuning and BERT frozen

Towards Making Most of BERT for NMT, [Yang et al AAAI 2020]

Not tuning too much

- Dynamic Switch
  - Use a gate to dynamically decide which part is more important
  - If σ is learned to 0, it degrade to the NMT model
  - If σ is learned to 1, it simply act as Bert fine-tune approach

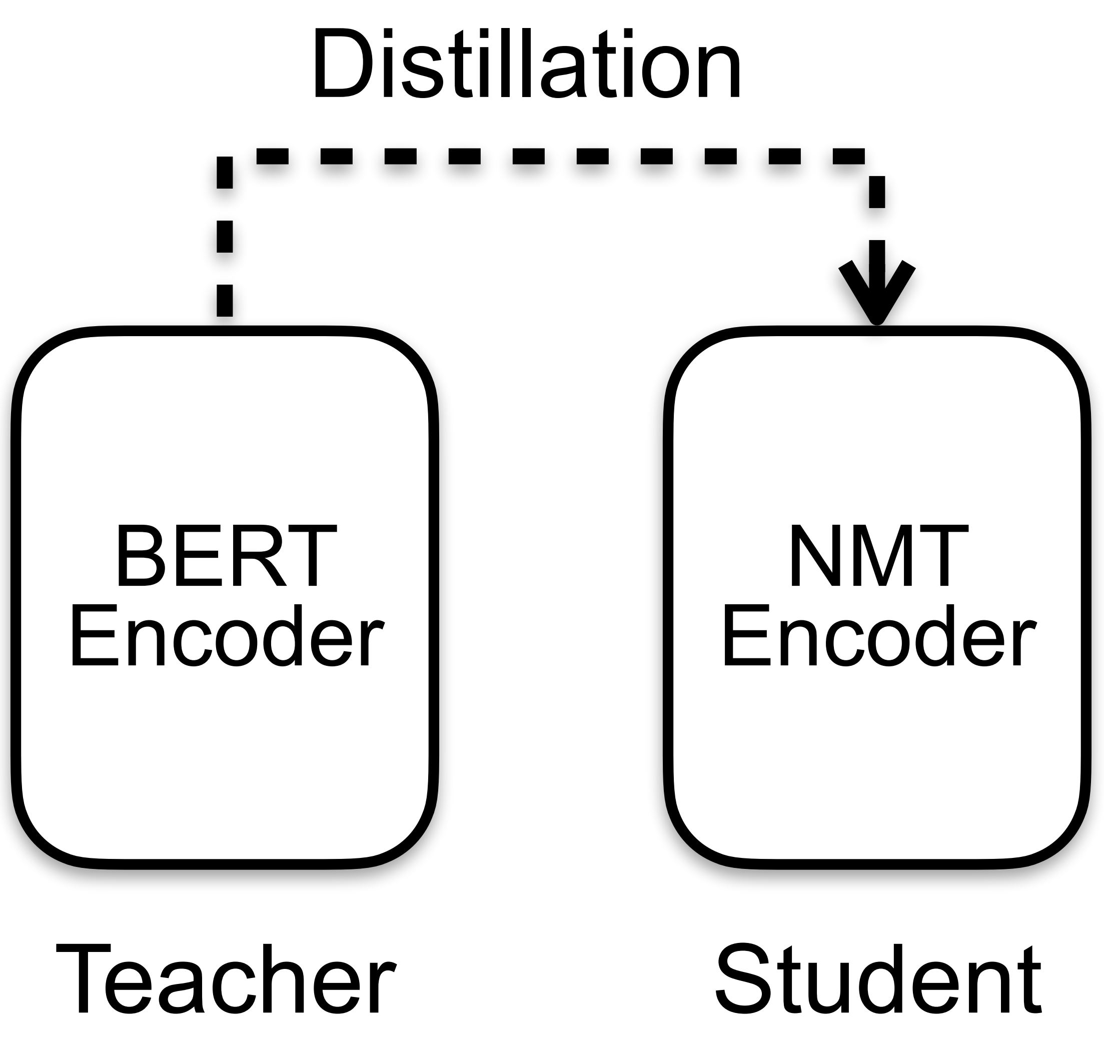


Dynamic Switch is more flexible than rate-scheduled learning rate

Towards Making Most of BERT for NMT, [Yang et al AAAI 2020]

Not tuning too much

- Asymptotic Distillation
  - The pre-trained BERT serves as a teacher network while the encoder of the NMT model serves as a student
  - Minimize MSE loss of hidden states between NMT encoder and BERT to retain the pre-trained information
  - Use a hyper-parameter to balances the preference between pre-training distillation and NMT objective



$$\mathcal{L}_{KD} = \| h_{\text{bert}} - h_{\text{nmt}} \|^2$$

Distillation Without introducing of additional parameters!

Towards Making Most of BERT for NMT, [Yang et al AAAI 2020]

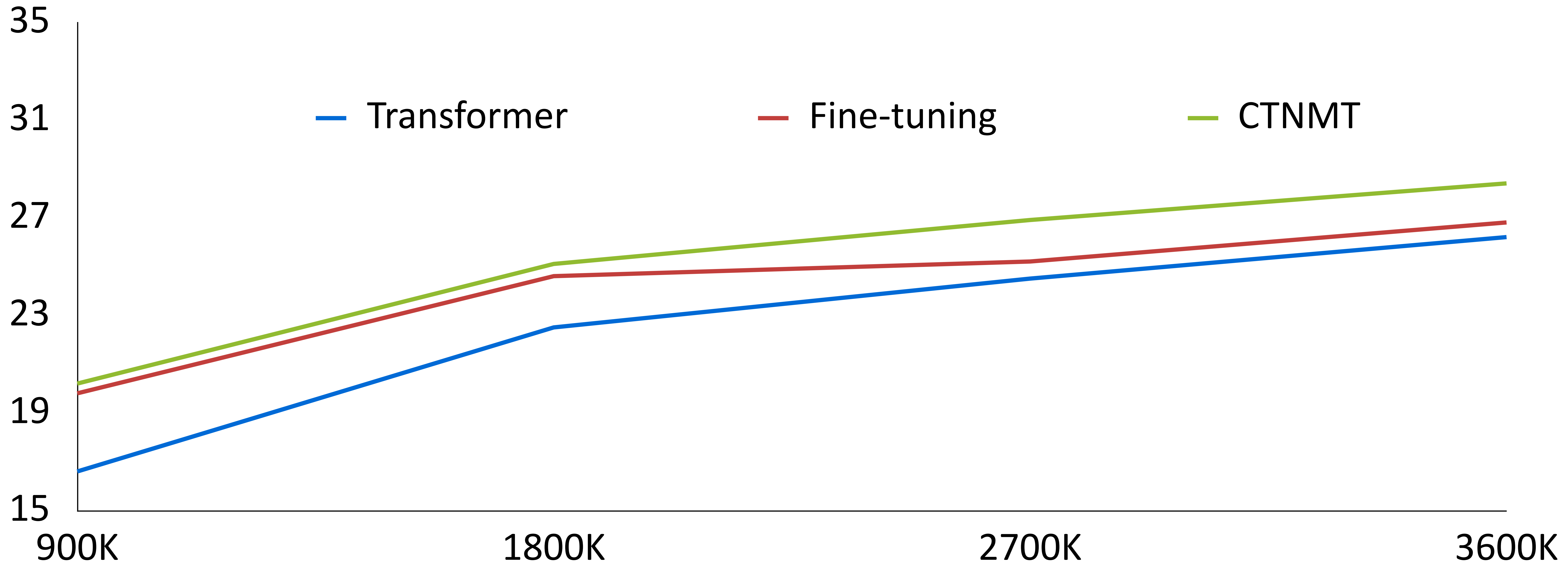
Not tuning too much

System	Architecture	En-De	En-Fr	En-Zh
	Existing systems			
Vaswani et al. (2017)	Transformer base	27.3	38.1	-
Vaswani et al. (2017)	Transformer big	28.4	41.0	-
Lample and Conneau (2019)	Transformer big + Fine-tuning	27.7	-	-
Lample and Conneau (2019)	Transformer big + Frozen Feature	28.7	-	-
Chen et al. (2018)	RNMT+ + MultiCol	28.7	41.7 -	
	Our NMT systems			
CTNMT	Transformer (base)	27.2	41.0	37.3
CTNMT	Rate-scheduling	29.7	41.6	38.4
CTNMT	Dynamic Switch	29.4	41.4	38.6
CTNMT	Asymptotic Distillation	29.2	41.6	38.3
CTNMT	+ ALL	**30.1**	**42.3**	**38.9**

- Three strategies can independently work well on WMT14 En-De, En-Fr and WMT18 En-Zh
- CTNMT base model achieves even better results than Transformer big model

Towards Making Most of BERT for NMT, [Yang et al AAAI 2020]

Not tuning too much



- CTNMT outperforms fine-tuning on all training steps
- The performance gaps is enlarged as the fine-tuning steps increasing

Towards Making Most of BERT for NMT, [Yang et al AAAI 2020]

Summary

- Advantage
  - Simple and effective, obtains +3 BLEU on WMT14 en-de benchmark
  - Three methods can be used separately or jointly
- Limitation
  - Introducing pre-training method for decoder is promising but still difficult
  - Cross attention is import but not pre-trained

Models	En→De BLEU
BERT Enc	29.2
BERT Dec	26.1
GPT-2 Enc	27.7
GPT-2 Dec	27.4

	Encoder	Decoder
GPT	✗	✗
BERT	✓	✗

Towards Making Most of BERT for NMT, [Yang et al AAAI 2020]

BERT Fusion

Incorporate BERT into Neural Machine Translation

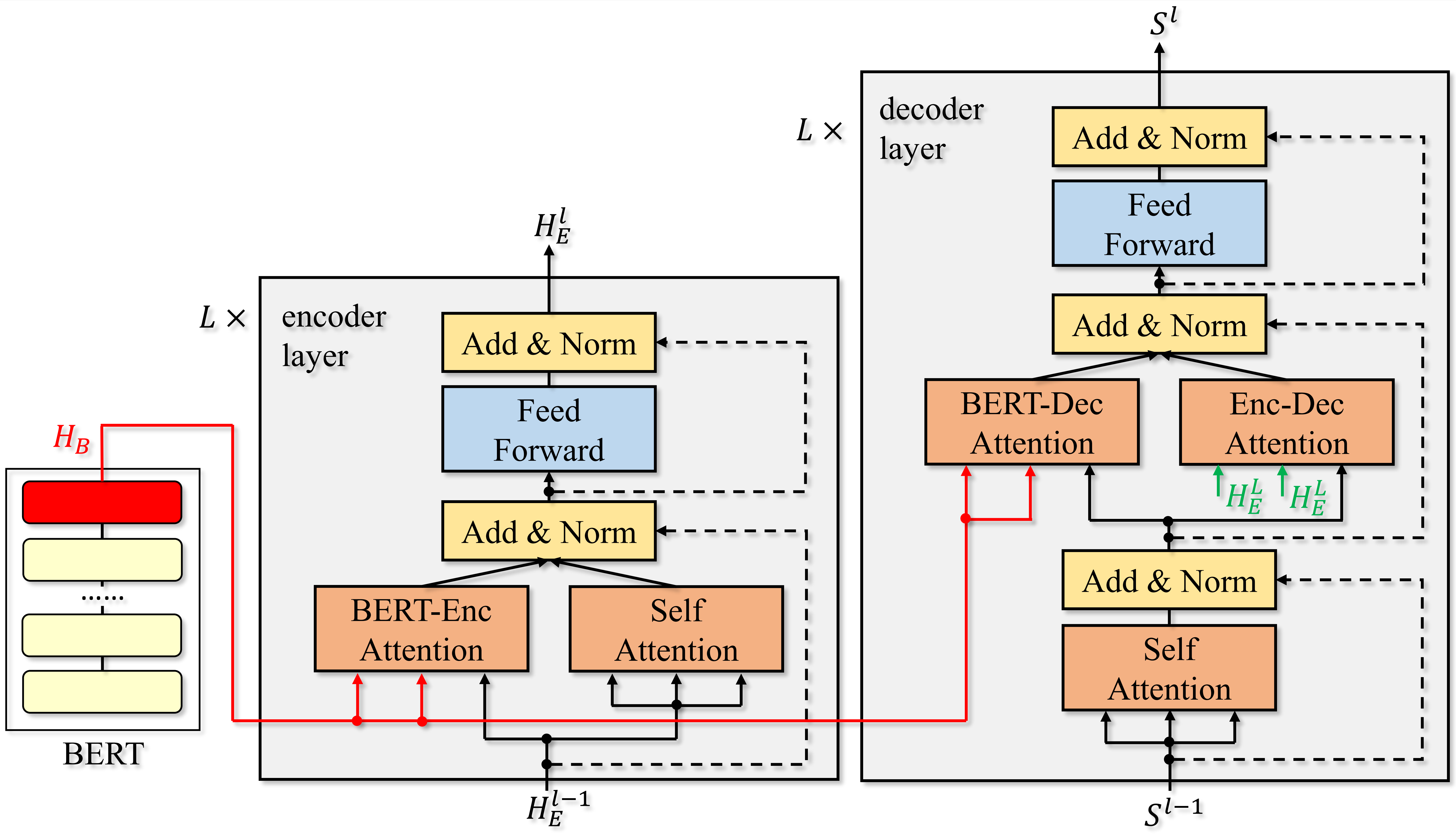
Table 1: Preliminary explorations on IWSLT’14 English→German translatio

Algorithm	BLEU score
Standard Transformer	28.57
Use BERT to initialize the encoder of NMT	27.14
Use XLM to initialize the encoder of NMT	28.22
Use XLM to initialize the decoder of NMT	26.13
Use XLM to initialize both the encoder and decoder of NMT	28.99
Leveraging the output of BERT as embeddings	29.67

- Fine-tuning BERT does NOT work !
  - BERT and XLM pre-training for the encoder decreased the performance
  - XLM pre-training for the decoder enlarged the performance gap
- BERT-Frozen achieved improvements

Incorporate BERT into Neural Machine Translation, [Zhu et al ICLR 2020]

Incorporate BERT into Neural Machine Translation

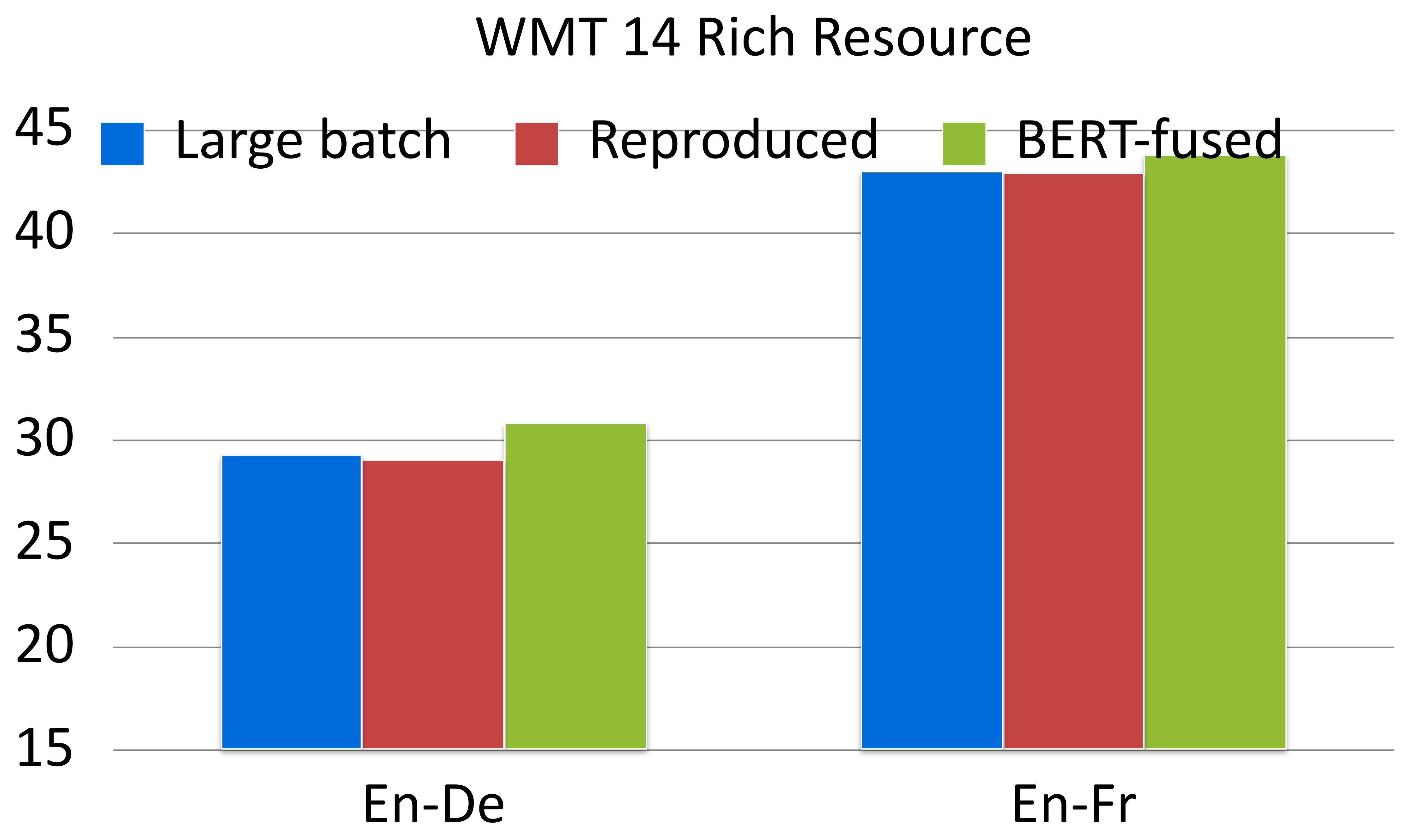


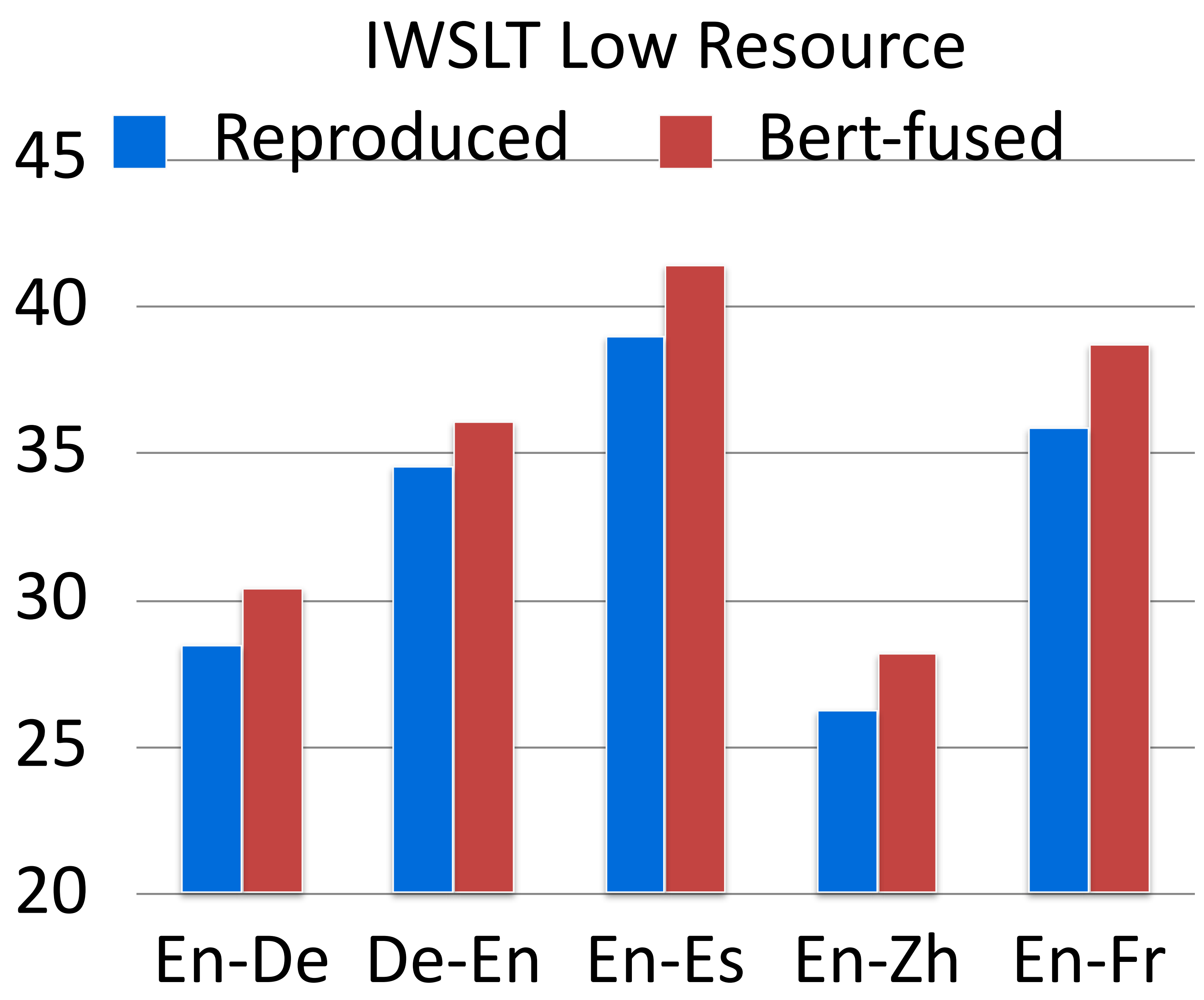
- BERT features are directly fed to both encoder and decoder layers
- Additional attention model to incorporate BERT features

Datasets and settings

- Fine-tuning dataset
  - Low resource: IWSLT En-De, En-FR, En-Zh, En-Es (less than 250 k sentence pairs)
  - Rich resource: WMT14 En-De and En-Fr (4 M and 36 M sentence pairs)
- Settings
  - BERT base for IWSLT
  - BERT large for WMT
  - Both the BERT-encoder and BERTdecoder attention are randomly initialized

Main results on supervised MT

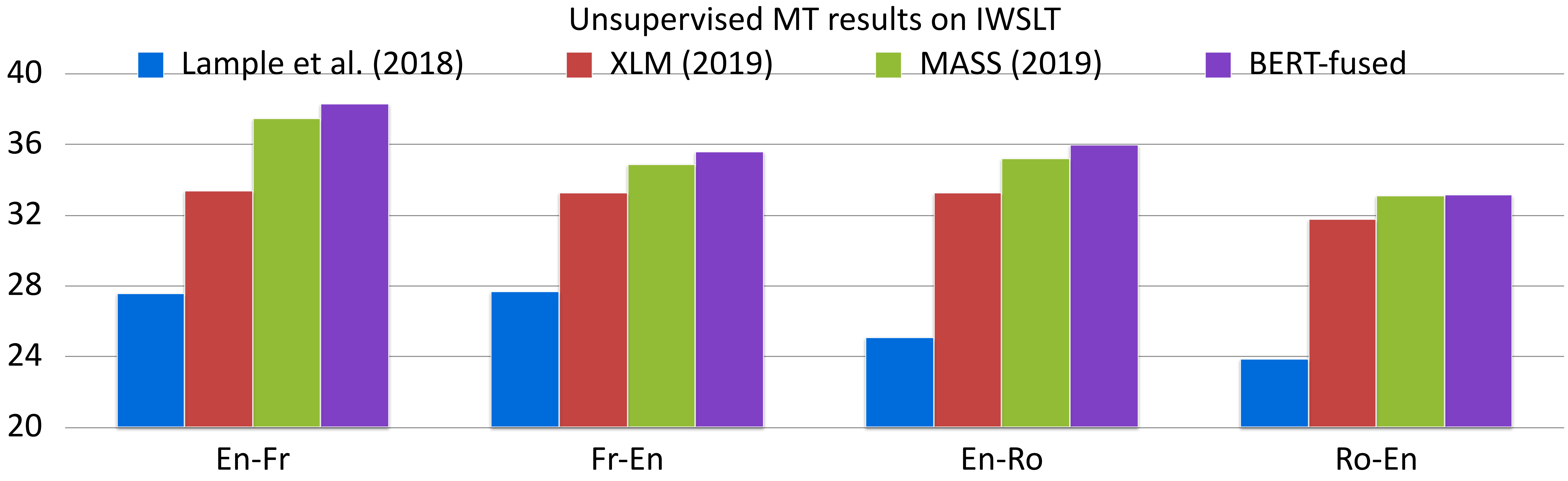




- Experiments on a strong baseline
- BERT-fused model outperforms transformer baseline in all settings

Incorporate BERT into Neural Machine Translation, [Zhu et al ICLR 2020]

Main results on unsupervised MT



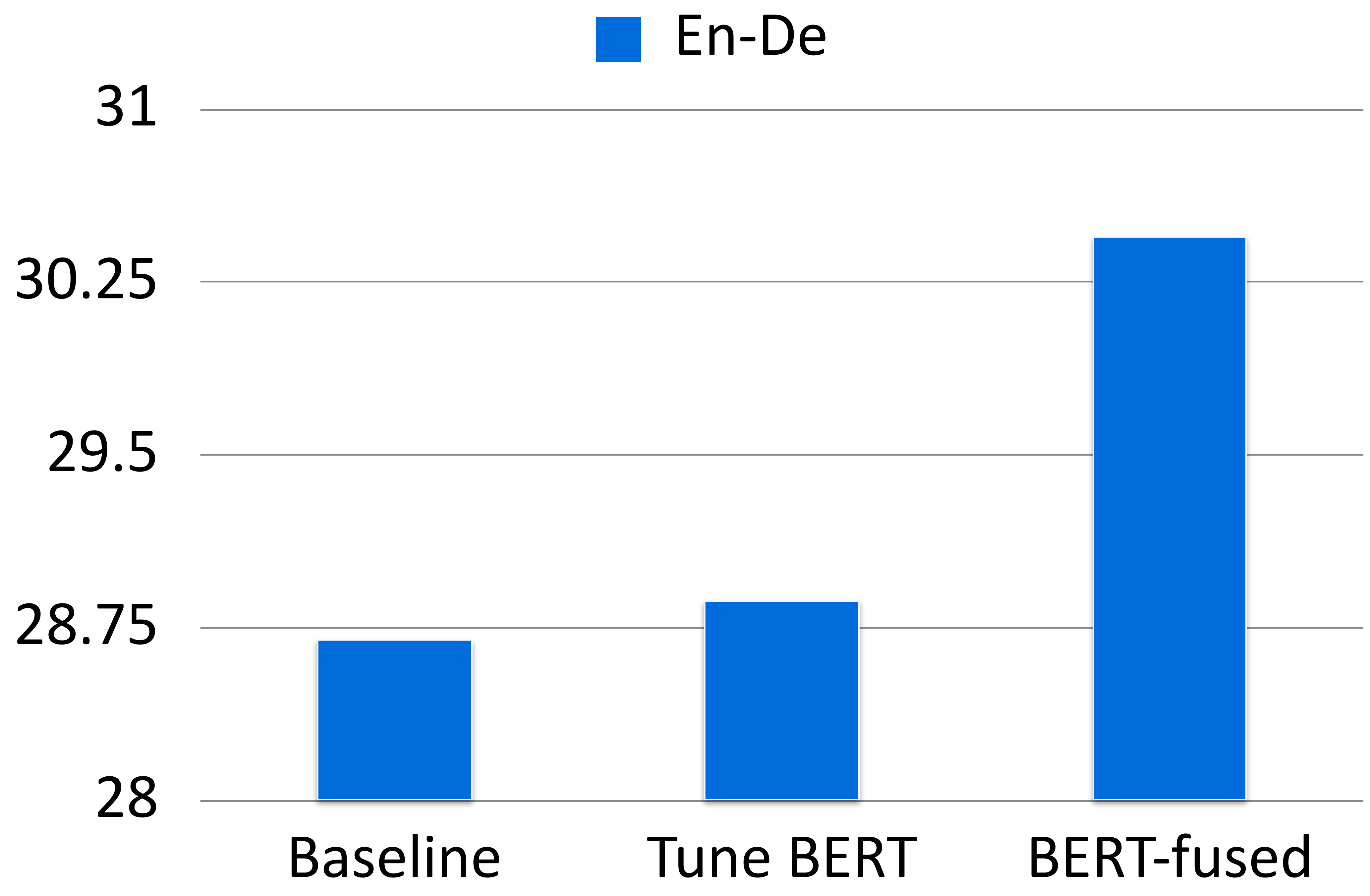
- Pre-training plays an crucial role in unsupervised NMT (Lample v.s. xml, mass and BERT-fused)
- BERT-fused outperforms XLM and MASS
- The comparison is slightly unfair, since BERT-fused introduced additional parameters

Incorporate BERT into Neural Machine Translation, [Zhu et al ICLR 2020]

NOT Tune BERT

Table 6: Ablation study on IWSLT'14 En→De.

Standard Transformer	28.57
BERT-fused model	30.45
Randomly initialize encoder/decoder of BERT-fused model	27.03
Jointly tune BERT and encoder/decoder of BERT-fused model	28.87
Feed BERT feature into all layers without attention	29.61
Replace BERT output with random vectors	28.91
Replace BERT with the encoder of another Transformer model	28.99
Remove BERT-encoder attention	29.87
Remove BERT-decoder attention	29.90



Jointly train BERT model with the NMT can also boost the baseline from 28.57 to 28.87.

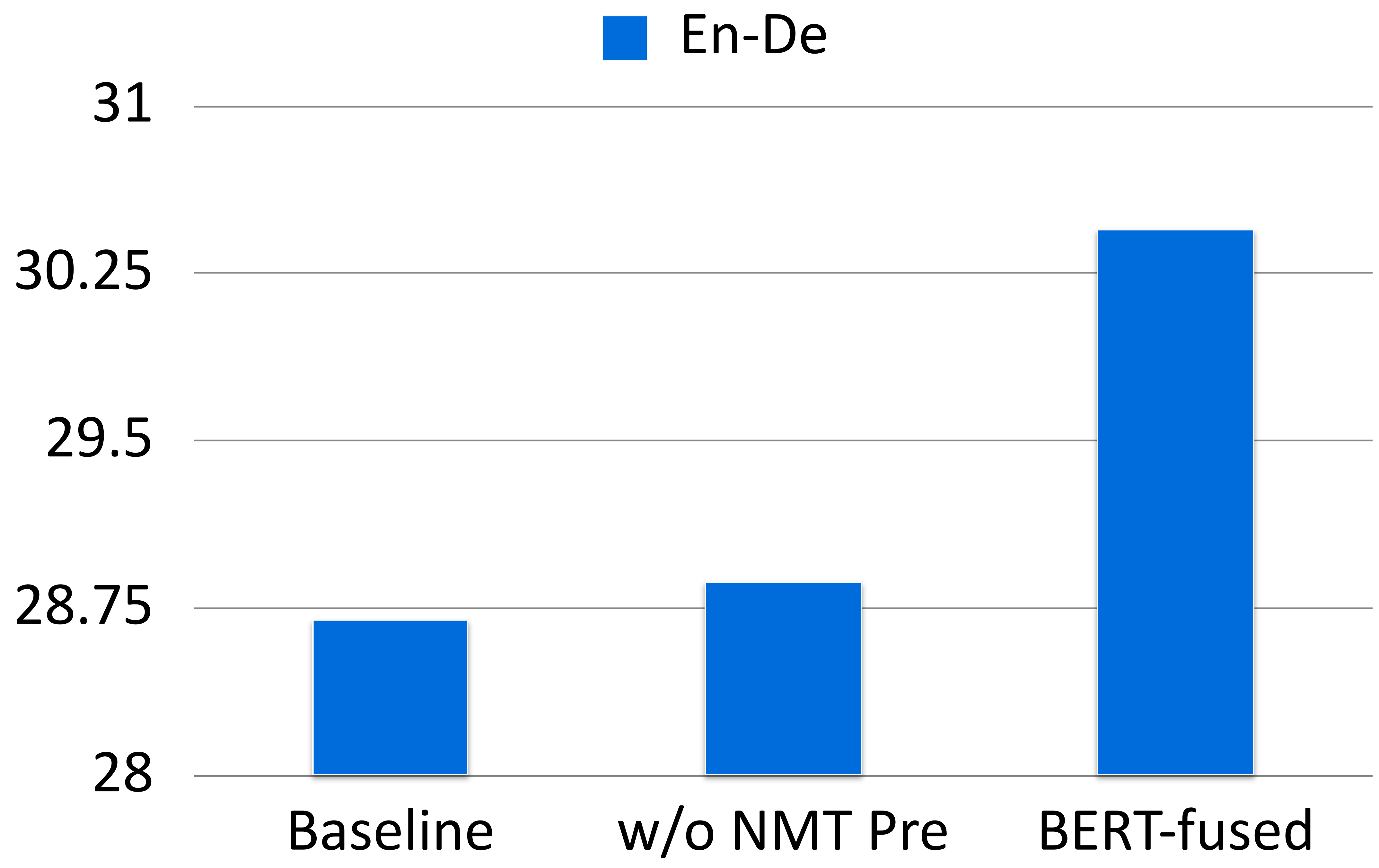
But it is not as good as fixing the BERT part, whose BLEU is 30.45

Incorporate BERT into Neural Machine Translation, [Zhu et al ICLR 2020]

NMT pre-training matters

Table 6: Ablation study on IWSLT'14 En→De.

Standard Transformer	28.57
BERT-fused model	30.45
Randomly initialize encoder/decoder of BERT-fused model	27.03
Jointly tune BERT and encoder/decoder of BERT-fused model	28.87
Feed BERT feature into all layers without attention	29.61
Replace BERT output with random vectors	28.91
Replace BERT with the encoder of another Transformer model	28.99
Remove BERT-encoder attention	29.87
Remove BERT-decoder attention	29.90



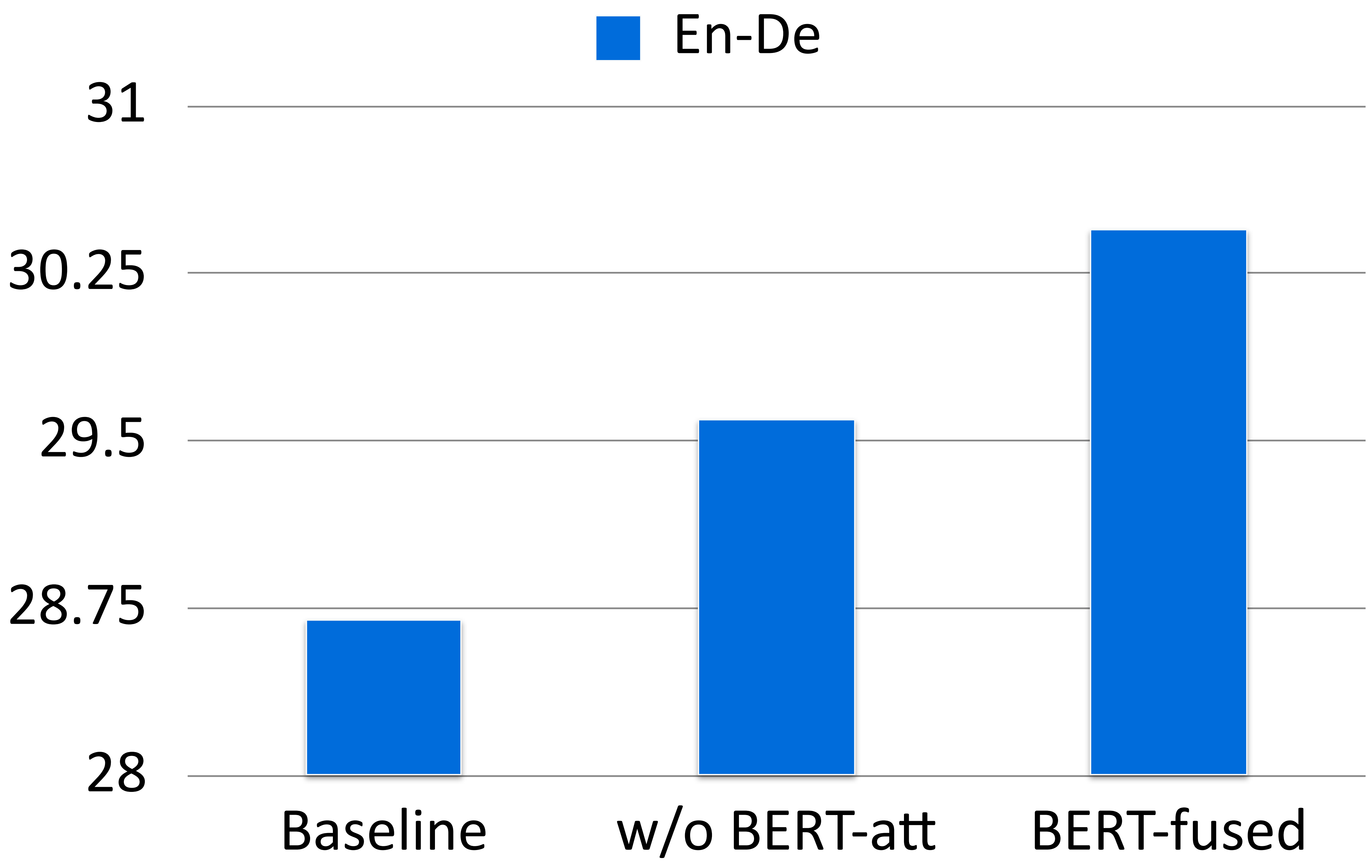
NMT Pre-training is also important to the success of BERT-fused model
Without NMT pre-training, the performance lags behind the baseline model

Incorporate BERT into Neural Machine Translation, [Zhu et al ICLR 2020]

BERT attention module matters

Table 6: Ablation study on IWSLT'14 En→De.

Standard Transformer	28.57
BERT-fused model	30.45
Randomly initialize encoder/decoder of BERT-fused model	27.03
Jointly tune BERT and encoder/decoder of BERT-fused model	28.87
Feed BERT feature into all layers without attention	29.61
Replace BERT output with random vectors	28.91
Replace BERT with the encoder of another Transformer model	28.99
Remove BERT-encoder attention	29.87
Remove BERT-decoder attention	29.90



Remove attention module, the performance still outperforms baseline, but falls behind BERT-fused model

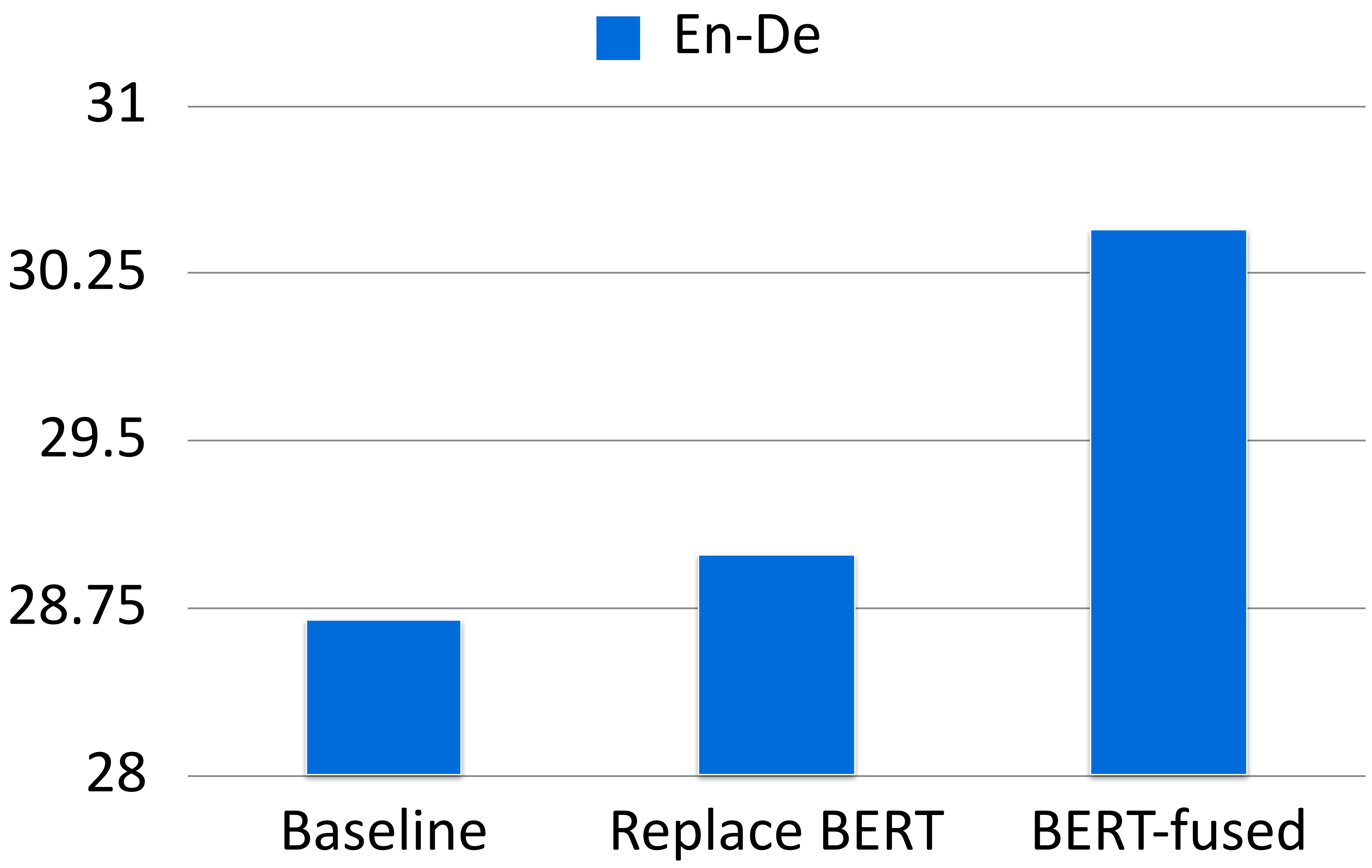
It suggest that separate BERT model provides additional gains

Incorporate BERT into Neural Machine Translation, [Zhu et al ICLR 2020]

Of course, BERT matters

Table 6: Ablation study on IWSLT'14 En→De.

Standard Transformer	28.57
BERT-fused model	30.45
Randomly initialize encoder/decoder of BERT-fused model	27.03
Jointly tune BERT and encoder/decoder of BERT-fused model	28.87
Feed BERT feature into all layers without attention	29.61
Replace BERT output with random vectors	28.91
Replace BERT with the encoder of another Transformer model	28.99
Remove BERT-encoder attention	29.87
Remove BERT-decoder attention	29.90

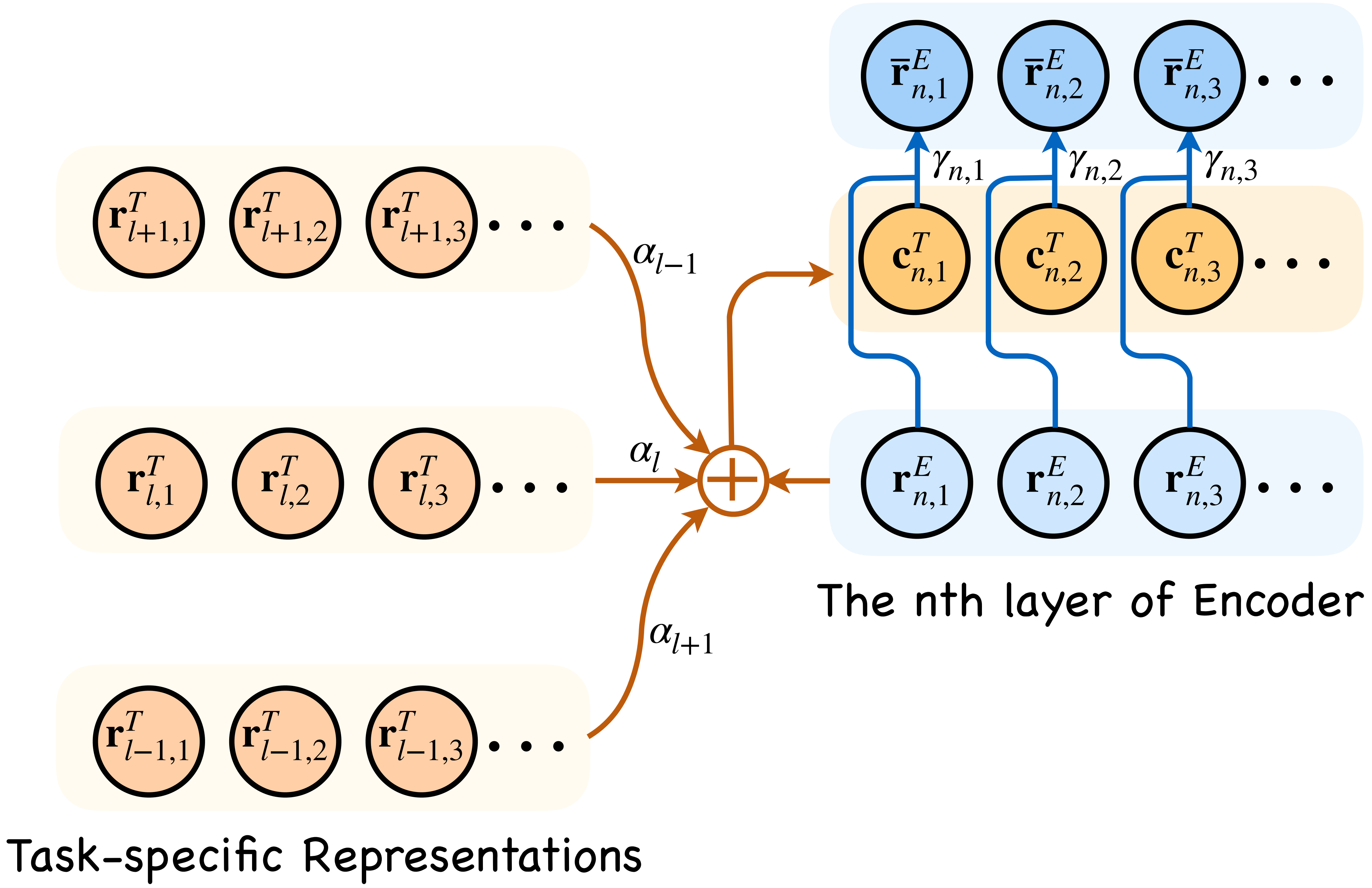


Replace BERT representation with another transformer model, the performance drops significantly

It indicates BERT provides meaningful information and the improvements is not from the additional parameters.

Incorporate BERT into Neural Machine Translation, [Zhu et al ICLR 2020]

Acquiring Knowledge from Pre-trained Model to Neural Machine Translation



- Key idea
  - Dynamic fusion of different BERT layers, while BERT-fused model only uses the last layer of BERT
  - Incorporate BERT into all encoder layers and decoder layers with adaptive weight
  - Experiments including both BERT & GPT

Acquiring Knowledge from Pre-trained Model to Neural Machine Translation, [Weng et al AAAI 2020]

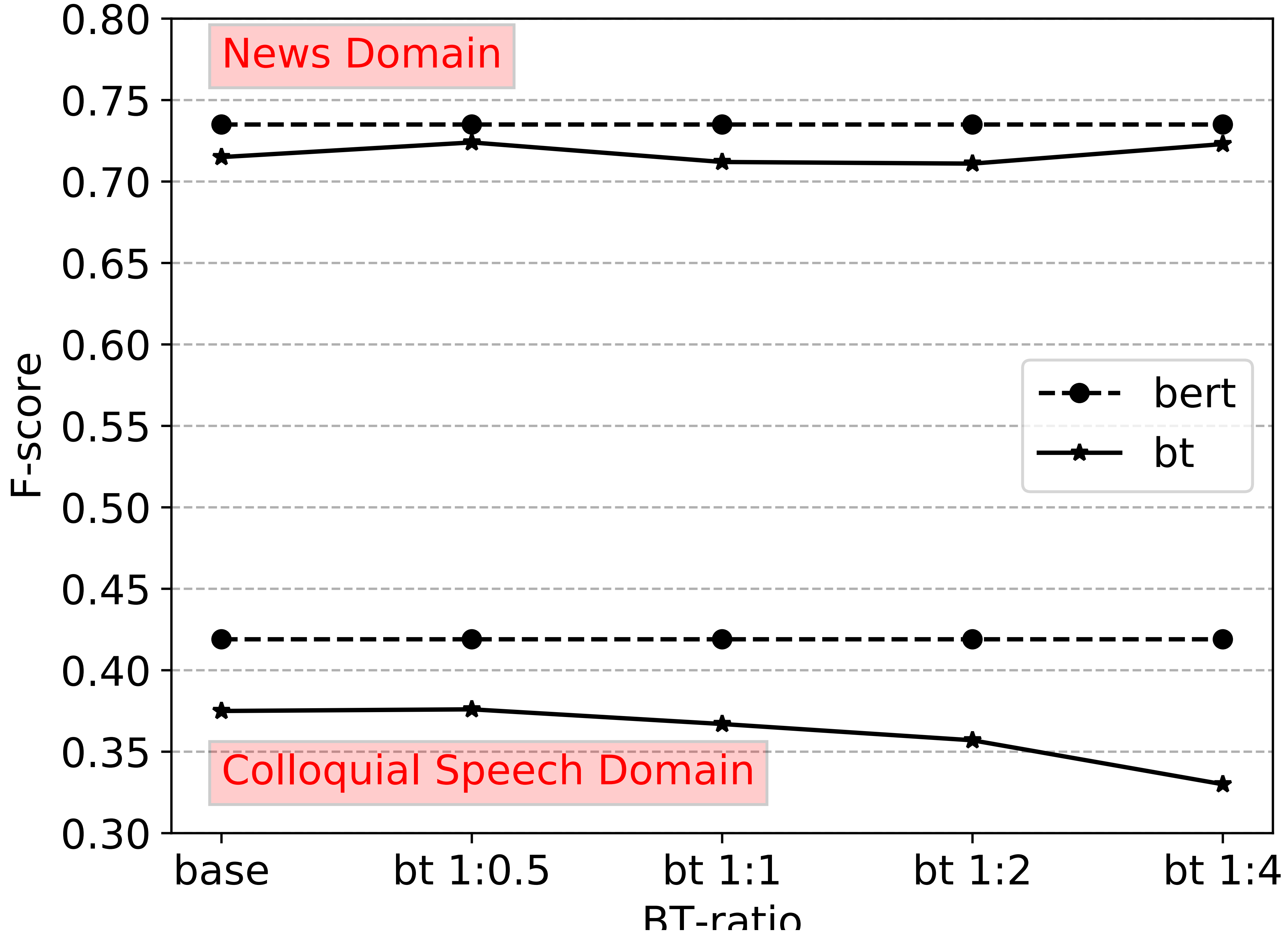
GPT v.s. BERT

Model	Pre-trained Model Encoder	Pre-trained Model Decoder	EN→DE BLEU	EN→DE Δ	DE→EN BLEU	DE→EN Δ	ZH→EN BLEU	ZH→EN Δ
Transformer (Vaswani et al. 2017)	N/A	N/A	27.3	–	N/A	–	N/A	–
Transformer (Zheng et al. 2019)	N/A	N/A	27.14	–	N/A	–	N/A	–
Transformer (Dou et al. 2018)	N/A	N/A	27.31	–	N/A	–	24.13	–
Transformer	N/A	N/A	27.31	–	32.51	–	24.47	–
w/ Fine-tuning	GPT	N/A	27.82	+0.51	33.17	+0.66	25.11	+0.64
	N/A	GPT	27.45	+0.14	32.87	+0.36	24.59	+0.12
	GPT	GPT	27.85	+0.54	32.79	+0.28	25.21	+0.74
	BERT	N/A	28.22	+0.91	33.64	+1.13	25.33	+0.86
	N/A	BERT	27.42	+0.11	33.13	+0.62	24.78	+0.31
	BERT	BERT	28.32	+1.01	33.57	+1.06	25.45	+0.98
	GPT	BERT	28.29	+0.98	33.33	+0.82	25.42	+0.95
	BERT	GPT	28.32	+1.01	33.57	+1.05	25.46	+0.99
	MASS		28.07	+0.76	33.29	+0.78	25.11	+0.64
	DAE		27.63	+0.33	33.03	+0.52	24.67	+0.20
w/ APT Framework	GPT	BERT	28.89	+1.58	34.32	+1.81	25.98	+1.51
	BERT	GPT	**29.23**	**+1.92**	**34.84**	**+2.33**	26.21	+1.74
	GPT	GPT	28.97	+1.66	34.26	+1.75	26.01	+1.54
	BERT	BERT	29.02	+1.71	34.67	+2.16	**26.46**	**+1.99**

Acquiring Knowledge from Pre-trained Model to Neural Machine Translation, [Weng et al AAAI 2020]

Pre-training has better generalization ability

System	En→De	Zh→En
Standard Transformer	29.20	45.15
+ back translation (1:0.5)	**30.41**	46.70
+ back translation (1:1)	30.25	**47.23**
+ back translation (1:2)	30.18	47.04
+ back translation (1:4)	30.25	46.39
BERT-fused model	30.03	46.55



- Pre-training is much more promising
  - better generalization ability
  - Back translation is limited with data scale

Comparison between Pre-training and Large-scale Back-translation, [Huang et al ACL 2021]

Summary

- Advantages
  - BERT features are fused in all layers
  - Additional attention model adaptively determine how to leverage BERT feature
- Limitions
  - Additional cost including training storage and inference time
  - Why not tune BERT?

Language Presentation

Reading

- Zhang et al. BERTScore: Evaluating Text Generation with BERT. 2020
- Rei et al. COMET: A Neural Framework for MT Evaluation. 2020
- Yang et al. Towards Making Most of BERT for NMT. 2020.