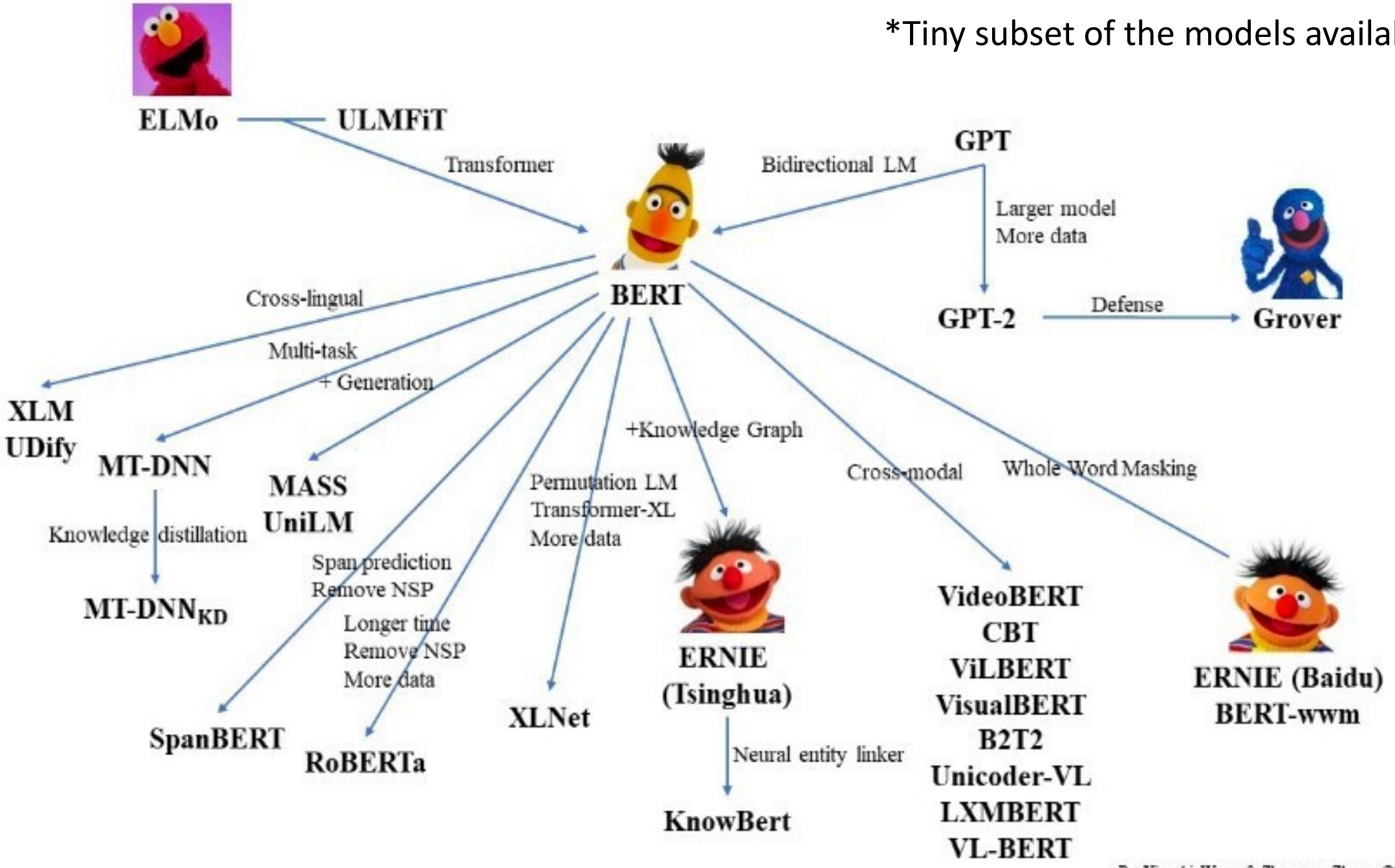
Some Extensions of BERT

*Tiny subset of the models available now



By Xiaozhi Wang & Zhengyan Zhang @THUNLP

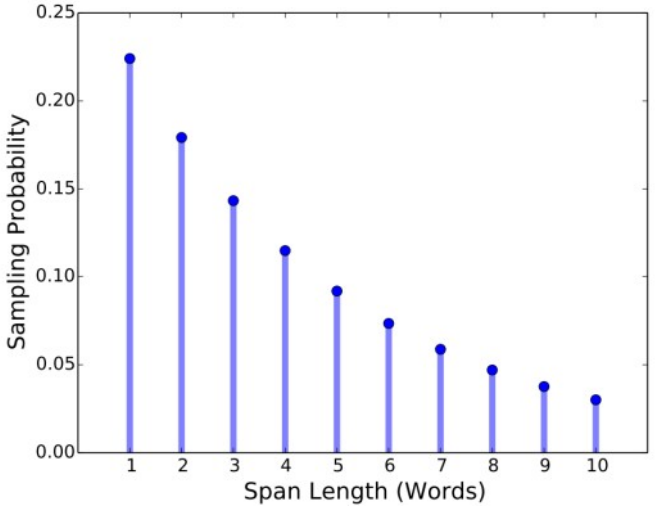
RoBERTa: Robustly Optimized BERT Pretraining Approach

- Improves BERT with better hyperparameter tuning and more data from Common Crawl
  - Training is longer, with bigger batches, on more data
    - Showed that more epochs alone help, even on the same data
  - Removes the next sentence prediction objective
  - Training on longer sequences; and
  - Dynamically changing the masking pattern applied to the training data.

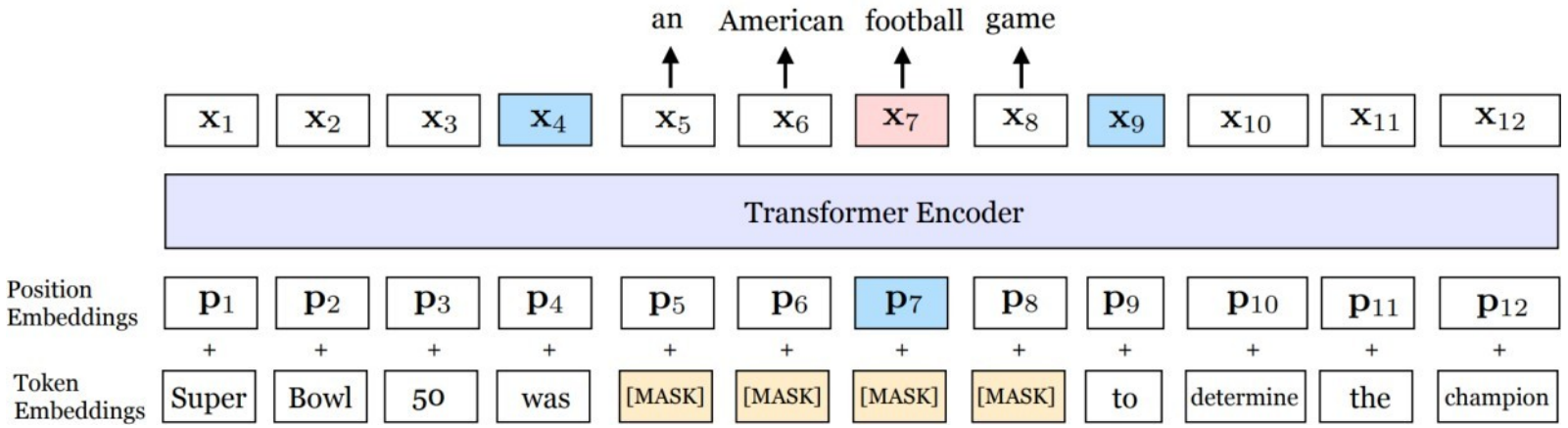
	MNLI	QNLI	QQP	RTE	SST	MRPC	CoLA	STS	WNLI	Avg
Single-task single models on dev										
$\text{BERT}_{\text{LARGE}}$	86.6/-	92.3	91.3	70.4	93.2	88.0	60.6	90.0	-	-
$\text{XLNet}_{\text{LARGE}}$	89.8/-	93.9	91.8	83.8	95.6	89.2	63.6	91.8	-	-
RoBERTa	**90.2/90.2**	**94.7**	**92.2**	**86.6**	**96.4**	**90.9**	**68.0**	**92.4**	**91.3**	-

https://arxiv.org/abs/1907.11692

SpanBERT https://aclanthology.org/2020.tacl-1.5/



- Span masking
  - A random process to mask spans of tokens
- Single sentence training
  - a single contiguous segment of text for each training sample (instead of two)
- Span boundary objective (SBO)
  - pred... $\mathcal{L}(\text{football}) = \mathcal{L}_{\text{MLM}}(\mathbf{x}_7) + \mathcal{L}_{\text{SBO}}(\mathbf{x}_4, \mathbf{x}_9, \mathbf{p}_7)$ ...ndary



SpanBERT: Results

- Masking scheme

	SQuAD 2.0	NewsQA	TriviaQA	Coreference	MNLI-m	QNLI
Subword Tokens	83.8	72.0	76.3	**77.7**	86.7	92.5
Whole Words	84.3	72.8	77.1	76.6	86.3	92.8
Named Entities	84.8	72.7	78.7	75.6	86.0	93.1
Noun Phrases	85.0	**73.0**	77.7	76.7	86.5	93.2
Random Spans	**85.4**	**73.0**	**78.8**	76.4	**87.0**	**93.3**

- Auxiliary ob

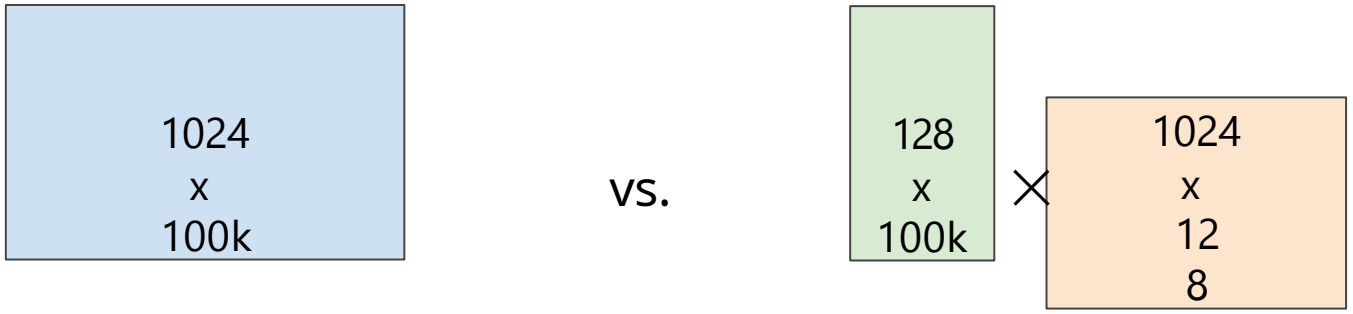
	SQuAD 2.0	NewsQA	TriviaQA	Coreference	MNLI-m	QNLI
Span Masking (2seq) + NSP	85.4	73.0	78.8	76.4	87.0	93.3
Span Masking (1seq)	86.7	73.4	80.0	76.3	87.3	93.8
Span Masking (1seq) + SBO	**86.8**	**74.1**	**80.3**	**79.0**	**87.6**	**93.9**

https://aclanthology.org/2020.tacl-1.5/

ALBERT

- *ALBERT: A Lite BERT for Self-supervised Learning of Language Representations*

- Innovation #1: Factorized embedding parameterization
  - Use small embedding size (e.g., 128) and then project it to the Transformer hidden size (e.g., 1024) with a parameter matrix



https://arxiv.org/abs/1909.11942

ALBERT

- Innovation #2: Cross-layer parameter sharing
  - Share all parameters between the Transformer layers
- Results

Models	MNLI	QNLI	QQP	RTE	SST	MRPC	CoLA	STS
Single-task single models on dev								
BERT-large	86.6	92.3	91.3	70.4	93.2	88.0	60.6	90.0
XLNet-large	89.8	93.9	91.8	83.8	95.6	89.2	63.6	91.8
RoBERTa-large	90.2	94.7	**92.2**	86.6	96.4	**90.9**	68.0	92.4
ALBERT (1M)	90.4	95.2	92.0	88.1	96.8	90.2	68.7	92.7
ALBERT (1.5M)	**90.8**	**95.3**	**92.2**	**89.2**	**96.9**	**90.9**	**71.4**	**93.0**

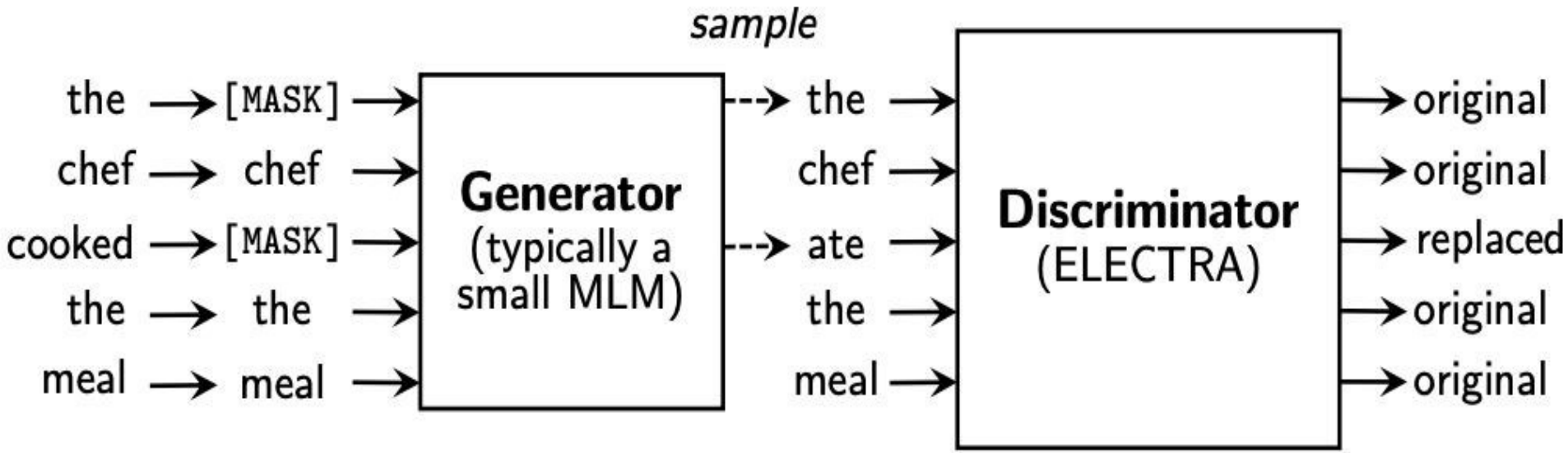
- ALBERT is light in terms of *parameters*, not *speed*

Model		Parameters	SQuAD1.1	SQuAD2.0	MNLI	SST-2	RACE	Avg	Speedup
BERT	base	108M	90.4/83.2	80.4/77.6	84.5	92.8	68.2	82.3	4.7x
	large	334M	92.2/85.5	85.0/82.2	86.6	93.0	73.9	85.2	1.0
ALBERT	base	12M	89.3/82.3	80.0/77.1	81.6	90.3	64.0	80.1	5.6x
	large	18M	90.6/83.9	82.3/79.4	83.5	91.7	68.5	82.4	1.7x
	xlarge	60M	92.5/86.1	86.1/83.1	86.4	92.4	74.8	85.5	0.6x
	xxlarge	235M	**94.1/88.3**	**88.1/85.1**	**88.0**	**95.2**	**82.3**	**88.7**	0.3x

https://arxiv.org/abs/1909.11942

ELECTRA

- *ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators*
- Train model to discriminate locally plausible text from real text



https://arxiv.org/abs/2003.10555

ELECTRA

- Difficult to match SOTA results with less compute

Model	Train FLOPs	Params	SQuAD 1.1		SQuAD 2.0	
			EM	F1	EM	F1
BERT-Base	6.4e19 (0.09x)	110M	80.8	88.5	–	–
BERT	1.9e20 (0.27x)	335M	84.1	90.9	79.0	81.8
SpanBERT	7.1e20 (1x)	335M	88.8	94.6	85.7	88.7
XLNet-Base	6.6e19 (0.09x)	117M	81.3	–	78.5	–
XLNet	3.9e21 (5.4x)	360M	**89.7**	**95.1**	87.9	**90.6**
RoBERTa-100K	6.4e20 (0.90x)	356M	–	94.0	–	87.7
RoBERTa-500K	3.2e21 (4.5x)	356M	88.9	94.6	86.5	89.4
ALBERT	3.1e22 (44x)	235M	89.3	94.8	87.4	90.2
BERT (ours)	7.1e20 (1x)	335M	88.0	93.7	84.7	87.5
ELECTRA-Base	6.4e19 (0.09x)	110M	84.5	90.8	80.5	83.3
ELECTRA-400K	7.1e20 (1x)	335M	88.7	94.2	86.9	89.6
ELECTRA-1.75M	3.1e21 (4.4x)	335M	**89.7**	94.9	**88.1**	**90.6**

https://arxiv.org/abs/2003.10555

Sentence BERT: Sentence Embeddings Using Siamese BERT-Networks

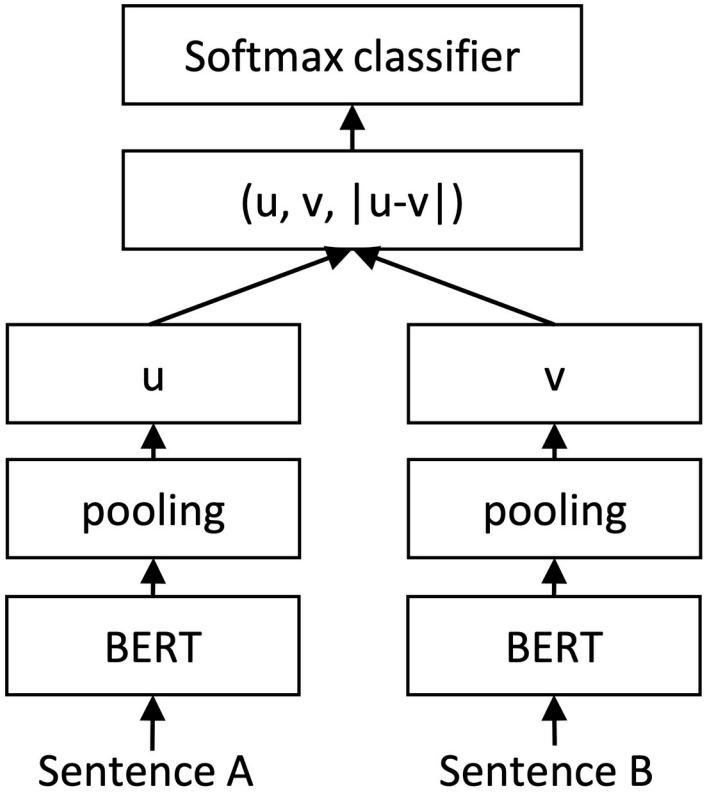


Figure 1: SBERT architecture with classification objective function, e.g., for fine-tuning on SNLI dataset. The two BERT networks have tied weights (siamese network structure).

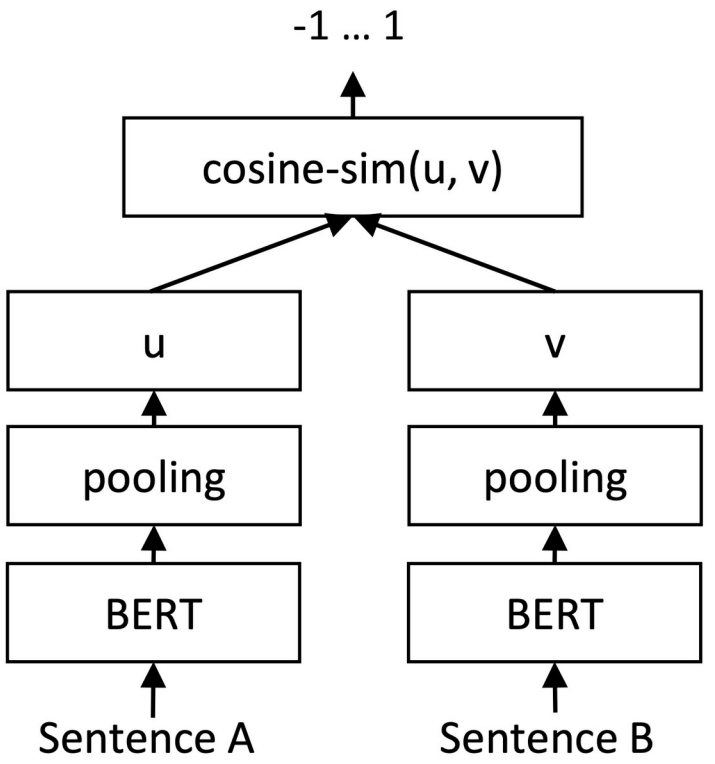


Figure 2: SBERT architecture at inference, for example, to compute similarity scores. This architecture is also used with the regression objective function.

https://aclanthology.org/D19-1410

Sentence BERT: Unsupervised Evaluation

Model	STS12	STS13	STS14	STS15	STS16	STSb	SICK-R	Avg.
Avg. GloVe embeddings	55.14	70.66	59.73	68.25	63.66	58.02	53.76	61.32
Avg. BERT embeddings	38.78	57.98	57.98	63.15	61.06	46.35	58.40	54.81
BERT CLS-vector	20.16	30.01	20.09	36.88	38.08	16.50	42.63	29.19
InferSent - Glove	52.86	66.75	62.15	72.77	66.87	68.03	65.65	65.01
Universal Sentence Encoder	64.49	67.80	64.61	76.83	73.18	74.92	**76.69**	71.22
SBERT-NLI-base	70.97	76.53	73.19	79.09	74.30	77.03	72.91	74.89
SBERT-NLI-large	72.27	**78.46**	**74.90**	80.99	76.25	**79.23**	73.75	76.55
SRoBERTa-NLI-base	71.54	72.49	70.80	78.74	73.69	77.77	74.46	74.21
SRoBERTa-NLI-large	**74.53**	77.00	73.18	**81.85**	**76.82**	79.10	74.29	**76.68**

Table 1: Spearman rank correlation ρ between the cosine similarity of sentence representations and the gold labels for various Textual Similarity (STS) tasks. Performance is reported by convention as $\rho \times 100$. STS12-STS16: SemEval 2012-2016, STSb: STSbenchmark, SICK-R: SICK relatedness dataset.

https://aclanthology.org/D19-1410

Sentence BERT: Supervised Evaluation on STSb

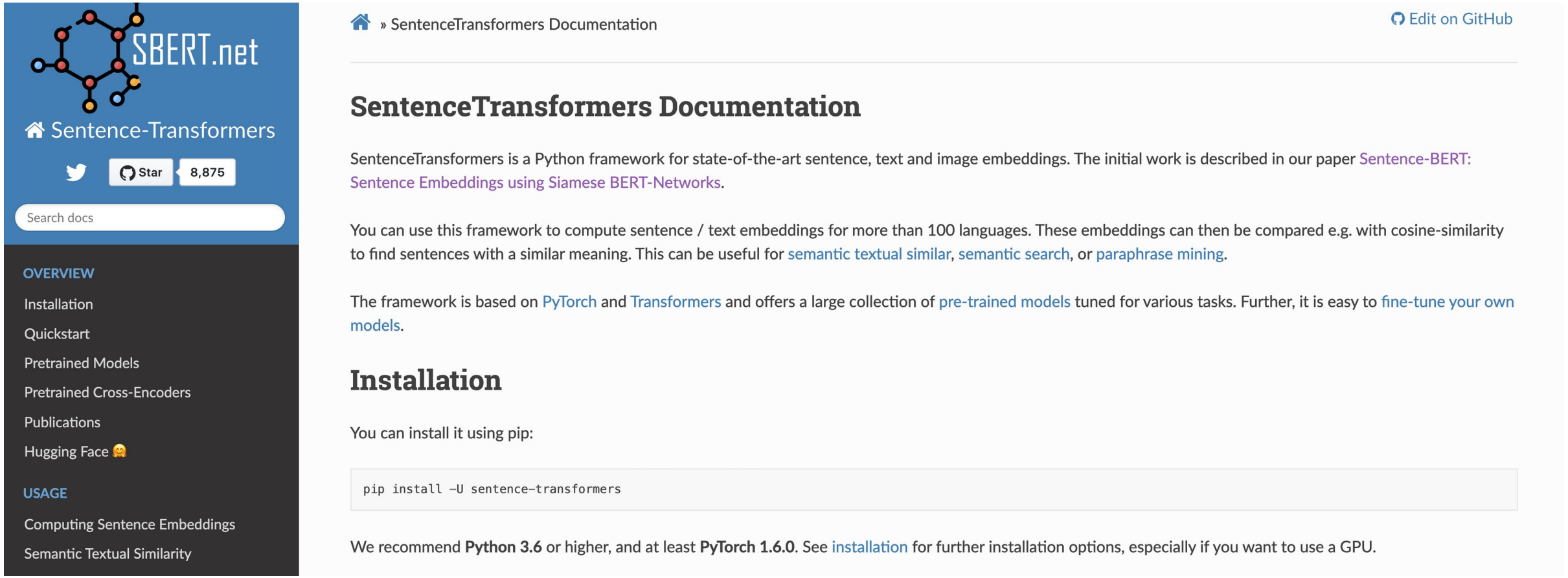
Model	Spearman
Not trained for STS	
Avg. GloVe embeddings	58.02
Avg. BERT embeddings	46.35
InferSent - GloVe	68.03
Universal Sentence Encoder	74.92
SBERT-NLI-base	77.03
SBERT-NLI-large	79.23

Trained on STS benchmark dataset	
BERT-STSb-base	84.30 ± 0.76
SBERT-STSb-base	84.67 ± 0.19
SRoBERTa-STSb-base	**84.92** ± 0.34
BERT-STSb-large	**85.64** ± 0.81
SBERT-STSb-large	84.45 ± 0.43
SRoBERTa-STSb-large	85.02 ± 0.76
Trained on NLI data + STS benchmark data	
BERT-NLI-STSb-base	**88.33** ± 0.19
SBERT-NLI-STSb-base	85.35 ± 0.17
SRoBERTa-NLI-STSb-base	84.79 ± 0.38
BERT-NLI-STSb-large	**88.77** ± 0.46
SBERT-NLI-STSb-large	86.10 ± 0.13
SRoBERTa-NLI-STSb-large	86.15 ± 0.35

https://aclanthology.org/D19-1410

Sentence Transformers: Multilingual, Search, Re-Ranking, Clustering, etc.

https://www.sbert.net/



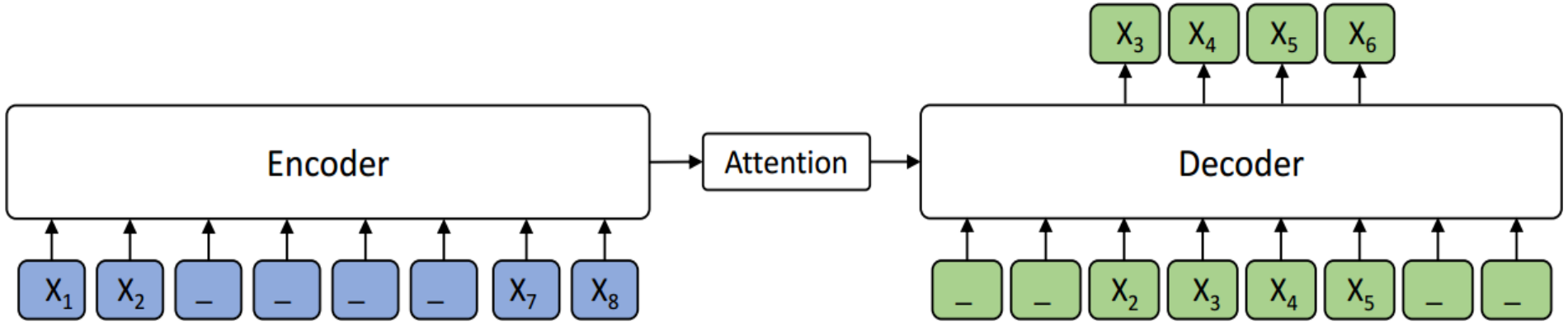
Pre-Training Seq2Seq Transformers: MASS

Pretraining Transformers

- BERT is good for “analysis” tasks (encoder-only)
- GPT-* is a good language model (decoder-only)
- How do we pretrain a seq2seq model?

MASS: Pre-train for Sequence to Sequence Generation

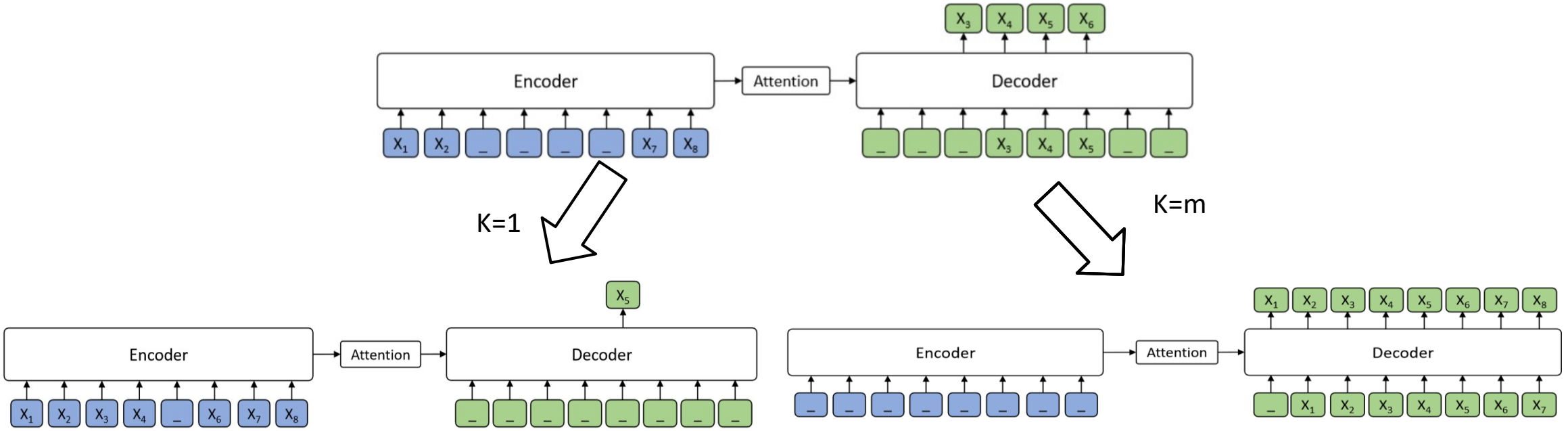
- Jointly pre-train the encoder and the decoder
- Mask *k* consecutive tokens (segment)
  - Force the decoder to attend on the source representations, i.e., encoder-decoder attention
  - Develop the decoder with the ability of language modeling



MASS: Pre-train for Sequence to Sequence Generation

[Song et al ICML 2019]

MASS vs. BERT/GPT



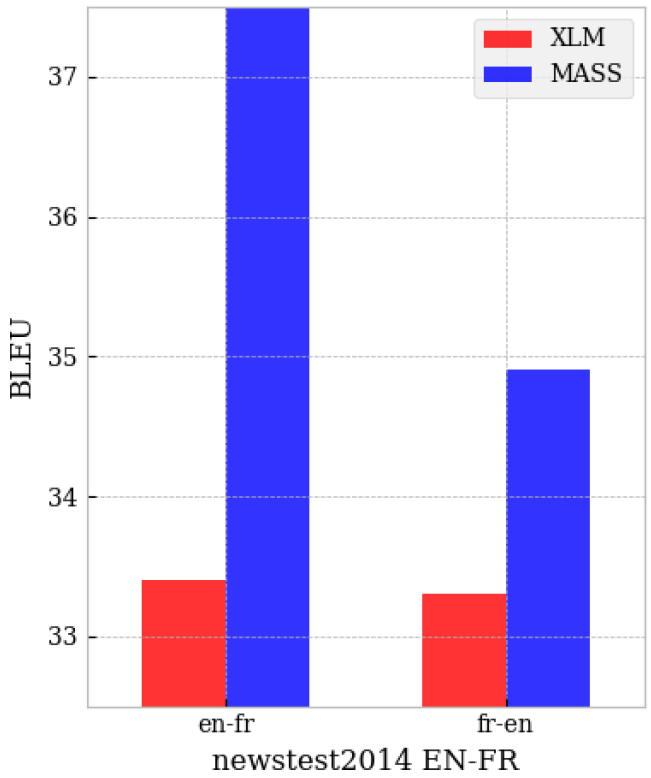
Length	Probability	Model
$k = 1$	$P(x^u \mid x^{\setminus u}; \theta)$	masked LM in BERT
$k \in [1, m]$	$P(x^{u:v} \mid x^{\setminus u:v}; \theta)$	MASS

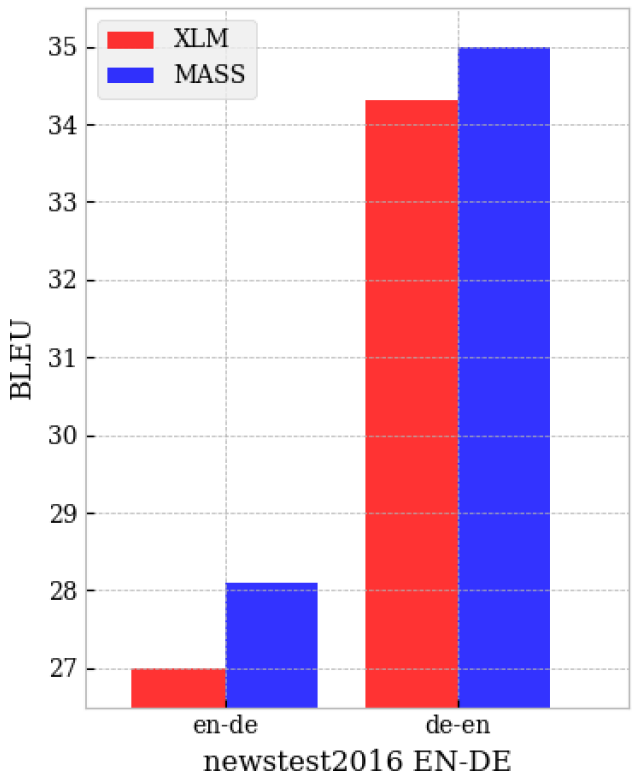
Length	Probability	Model
$k = m$	$P(x^{1:m} \mid x^{\setminus 1:m}; \theta)$	standard LM in GPT
$k \in [1, m]$	$P(x^{u:v} \mid x^{\setminus u:v}; \theta)$	MASS

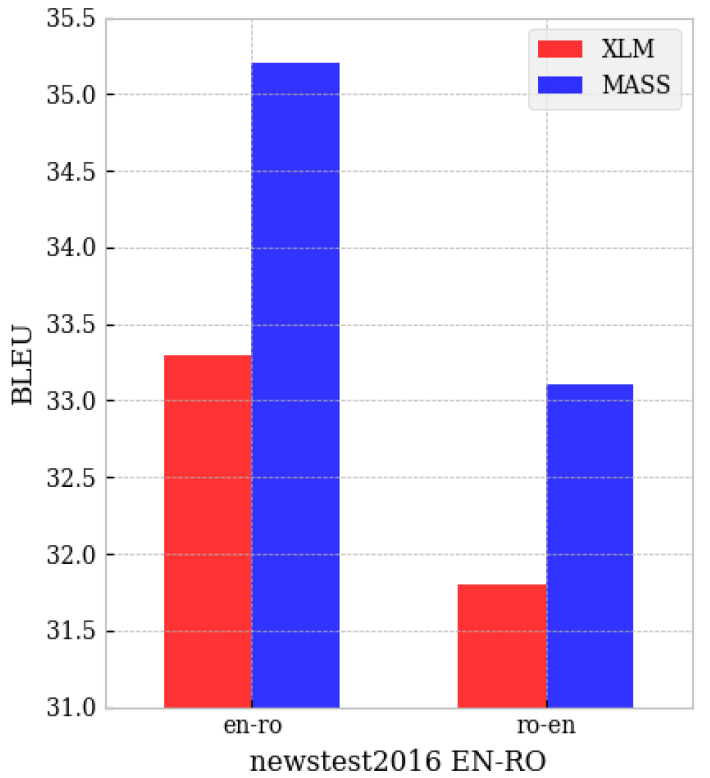
MASS: Pre-train for Sequence to Sequence Generation

[Song et al ICML 2019]

MASS: Results for Unsupervised NMT



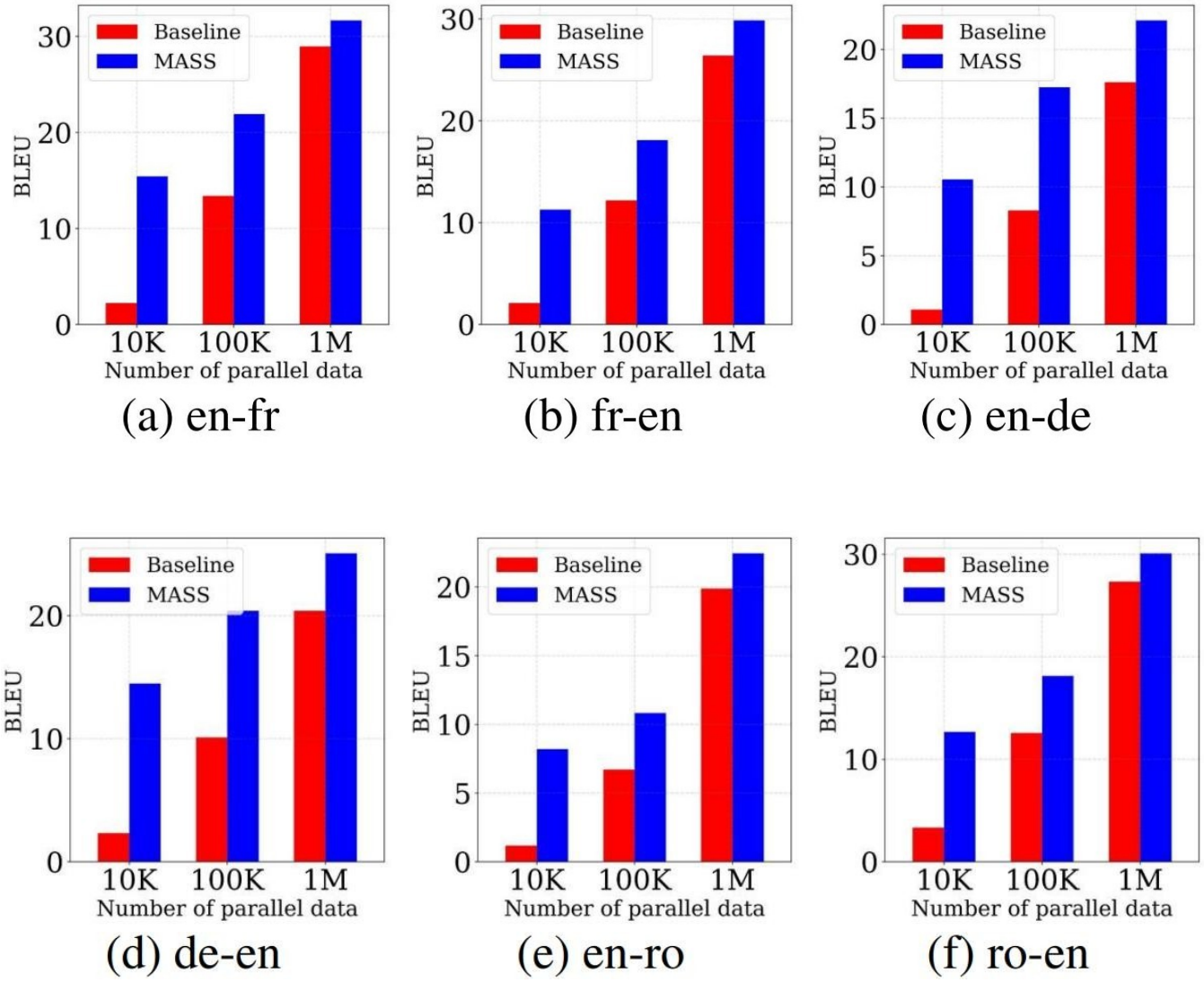




MASS: Pre-train for Sequence to Sequence Generation

[Song et al ICML 2019]

MASS: Results for Low-Resource NMT



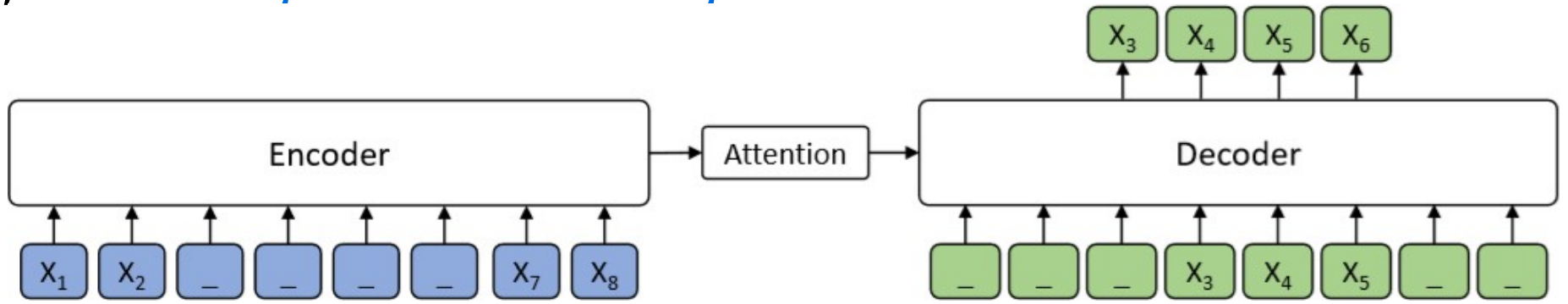
MASS: Pre-train for Sequence to Sequence Generation

[Song et al ICML 2019]

MASS: Summary

- Advantages
  - Unified sequence-to-sequence pretraining that jointly pretrains the encoder, the decoder, and the cross-attention
  - Achieves improvements on zero-shot / unsupervised NMT
- Limitions
  - No experiments on rich resource NMT
  - The pretraining objective is inconsistent with NMT

    e.g., monolinqual vs. multilinqual



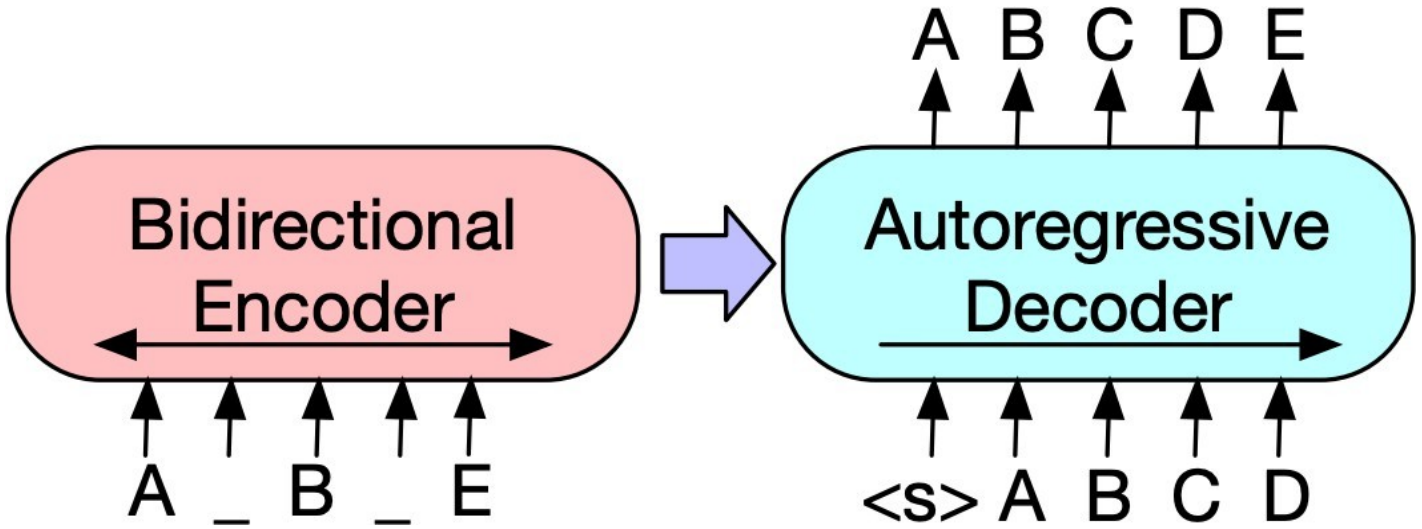
MASS: Pre-train for Sequence to Sequence Generation

[Song et al ICML 2019]

BART

BART

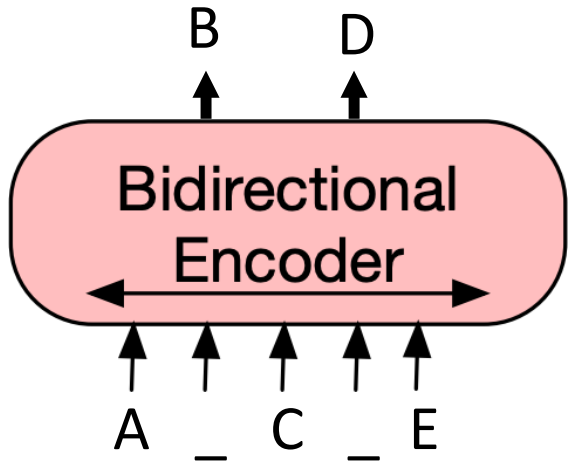
- Standard sequence-to-sequence Transformer architecture
- Trained by corrupting documents
- Optimizes a reconstruction loss: the cross-entropy between the decoder's output and the original document.
- Allows to apply *any* type of document corruption.

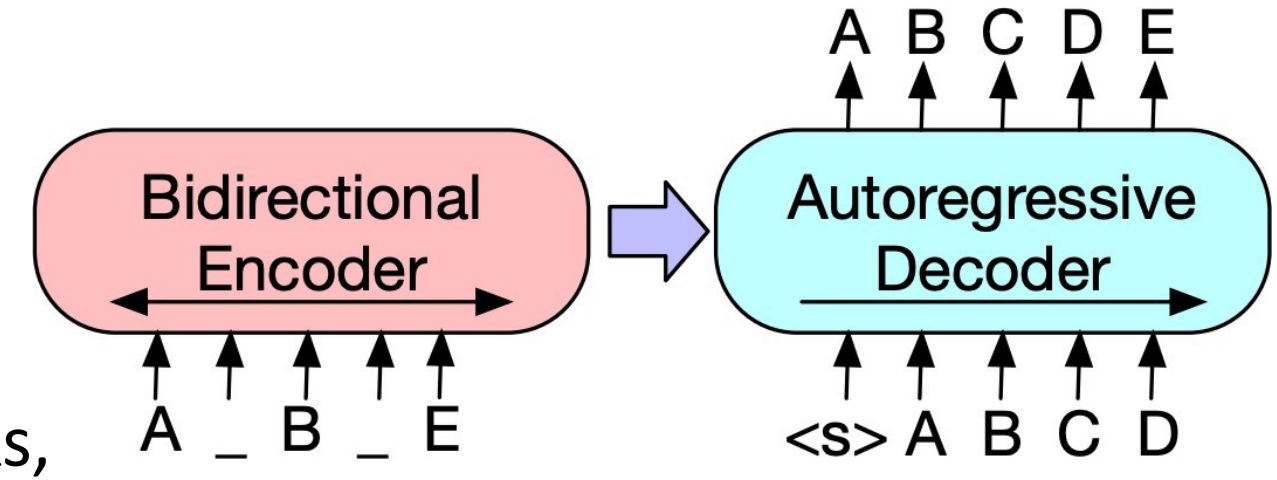


BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension, [Lewis et al ACL 2020]

BERT vs. BART

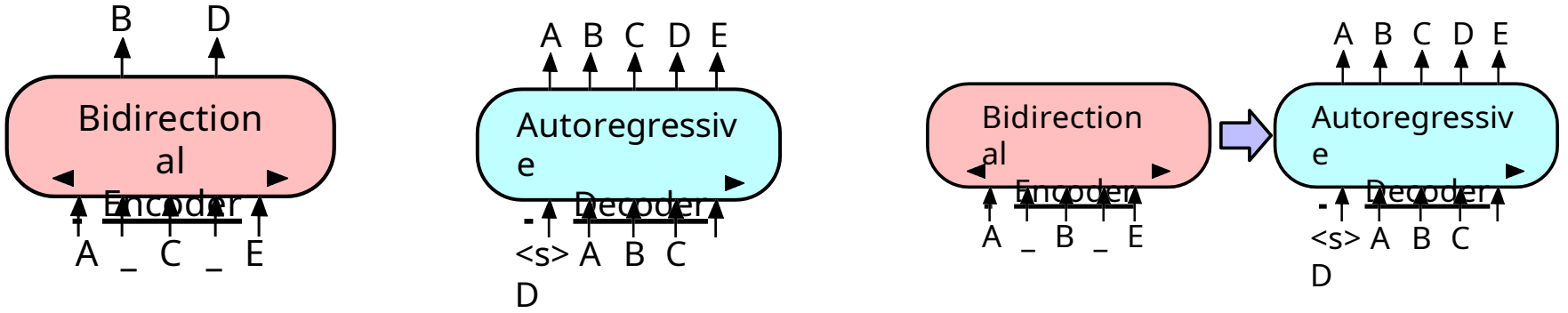
- BERT: encoder only, trained with a masked language modeling objective
  - No way to do translation or left-to-right language modeling
- BART: both an encoder and a decoder
  - Typically used for seq2seq tasks, but can use either for analysis





BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension, ACL 2020] [Lewis et al

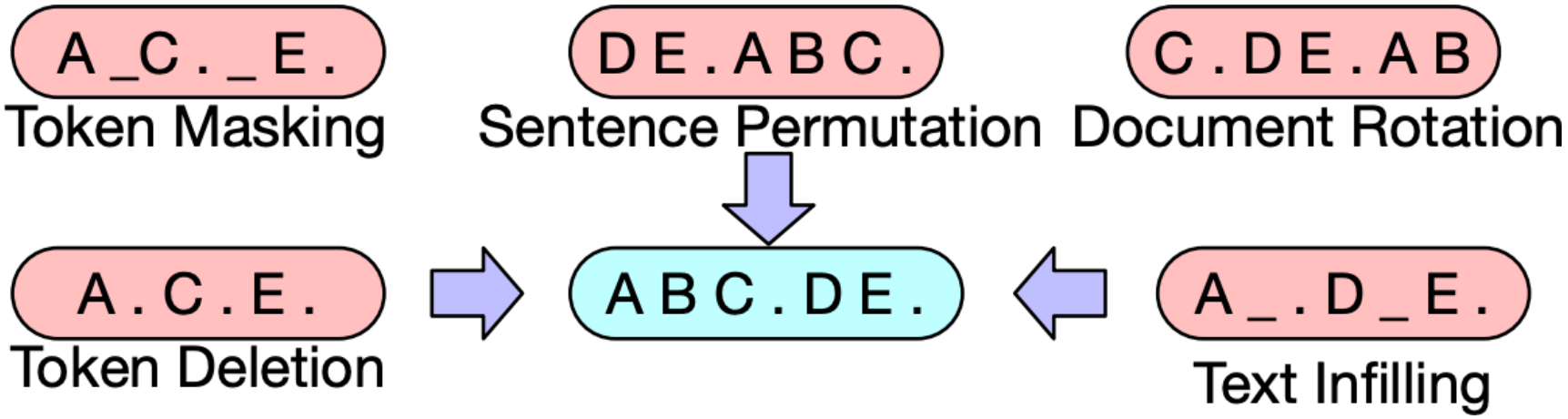
BERT vs. GPT vs. BART



BERT GPT BART

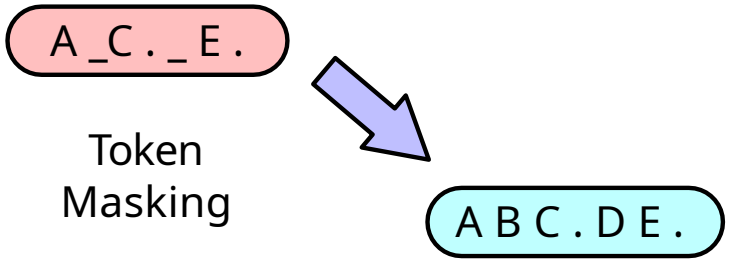
BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension, ACL 2020] [Lewis et al

BART



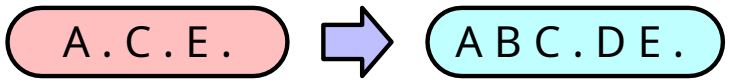
BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension, [Lewis et al ACL 2020]

BART: Noising the Input



Token masking: Random tokens are sampled and replaced with [MASK]

BART: Noising the Input



Token Deletion

Token deletion: Random tokens are deleted from the input.

BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension, ACL 2020] [Lewis et al

BART: Noising the Input

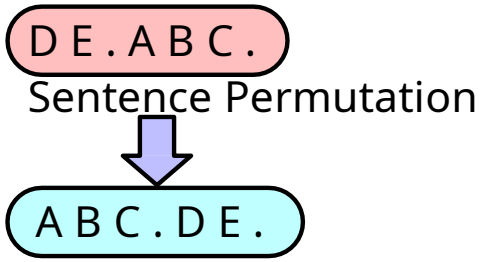
Text Infilling

- **Text infilling:**
  - A number of span are sampled.
  - Each span is replaced with [MASK].

Note: 0-length span corresponds to inserting [MASK].

BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension, [Lewis et al ACL 2020]

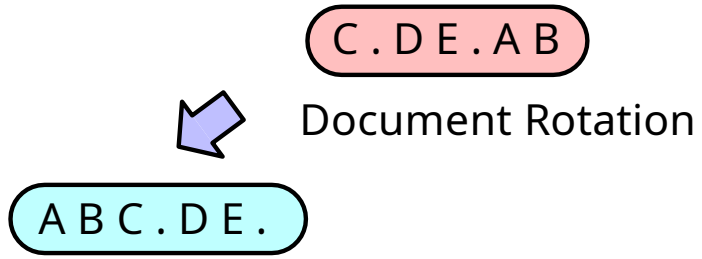
BART: Noising the Input



Sentence permutation: Sentences are shuffled with random order.

BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension, ACL 2020] [Lewis et al

BART: Noising the Input



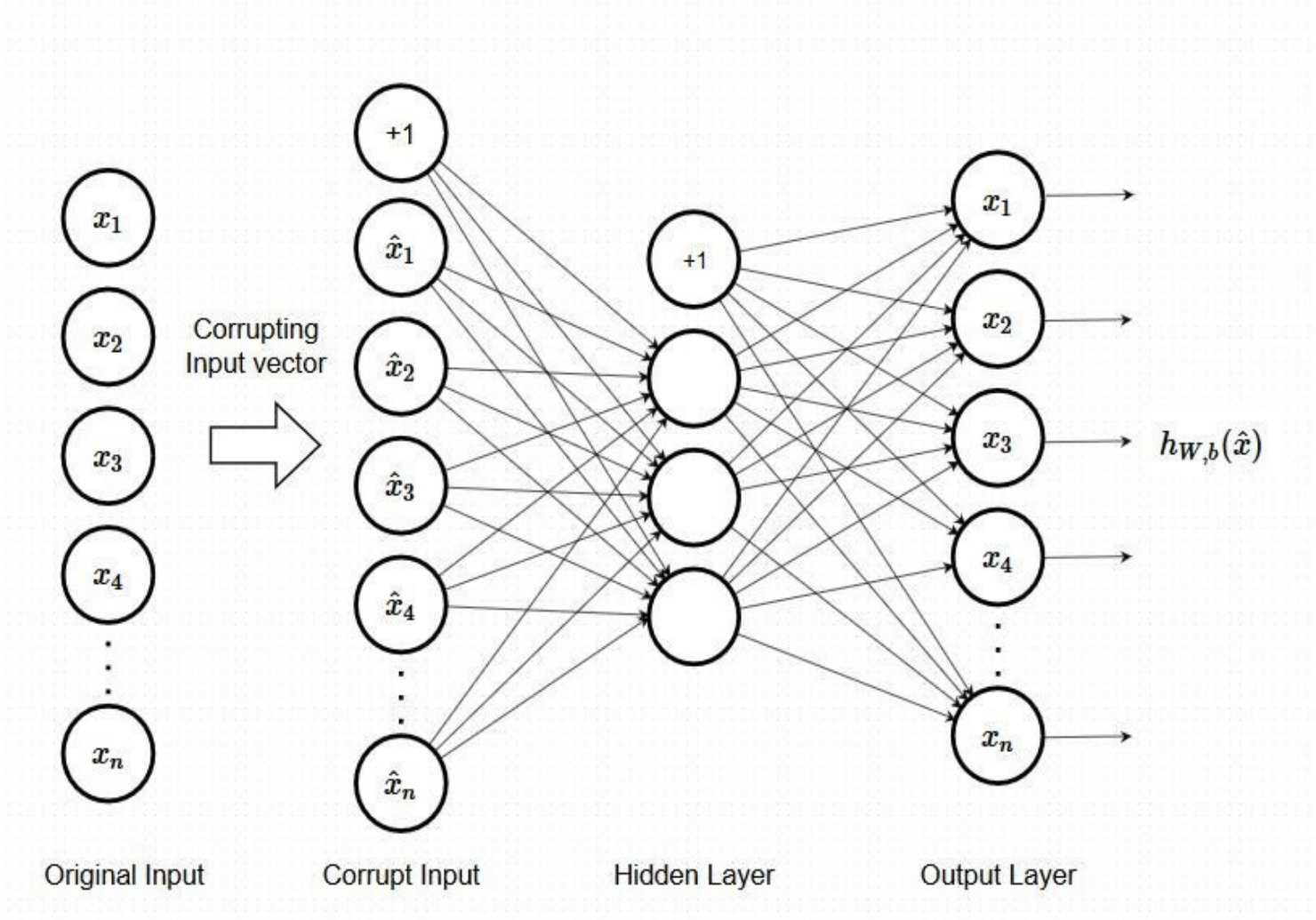
Document rotation:

- a token is chosen uniformly at random
- the document is rotated so that it begins with that token

BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension, ACL 2020] [Lewis et al

BART is a Denoising Autoencoder

A classic denoising auto-encoder



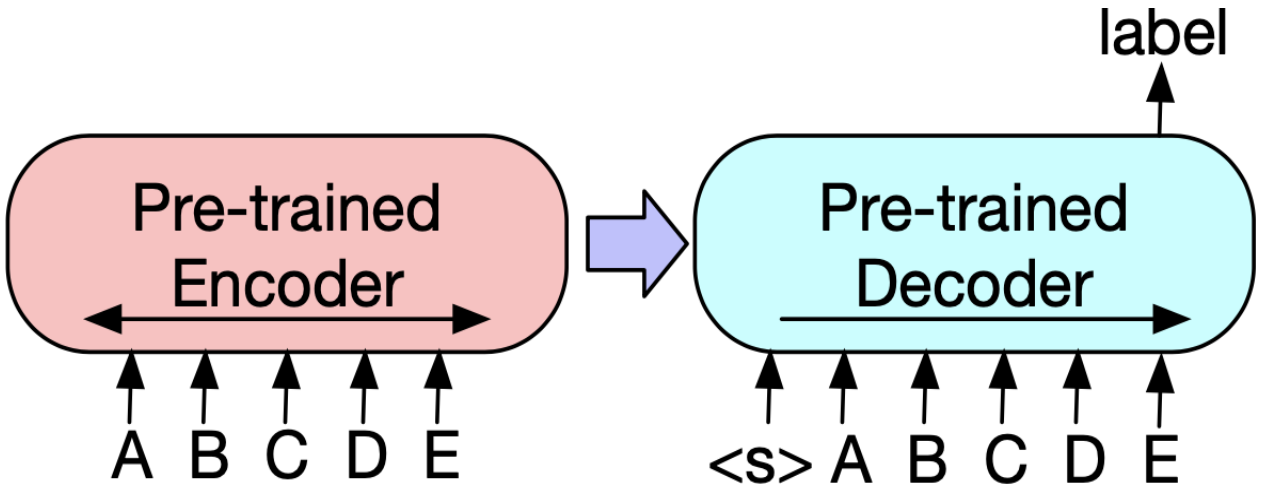
Fine-Tuning BART

Question: How do we fine-tune BART for sequence classification tasks?

Fine-Tuning BART

Sequence Classification Tasks

- The same input is fed into the encoder and the decoder
- The final hidden state of the final decoder token is fed into a multi-class linear classifier.



(a) To use BART for classification problems, the same input is fed into the encoder and decoder, and the representation from the final output is used.

BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension, [Lewis et al ACL 2020]

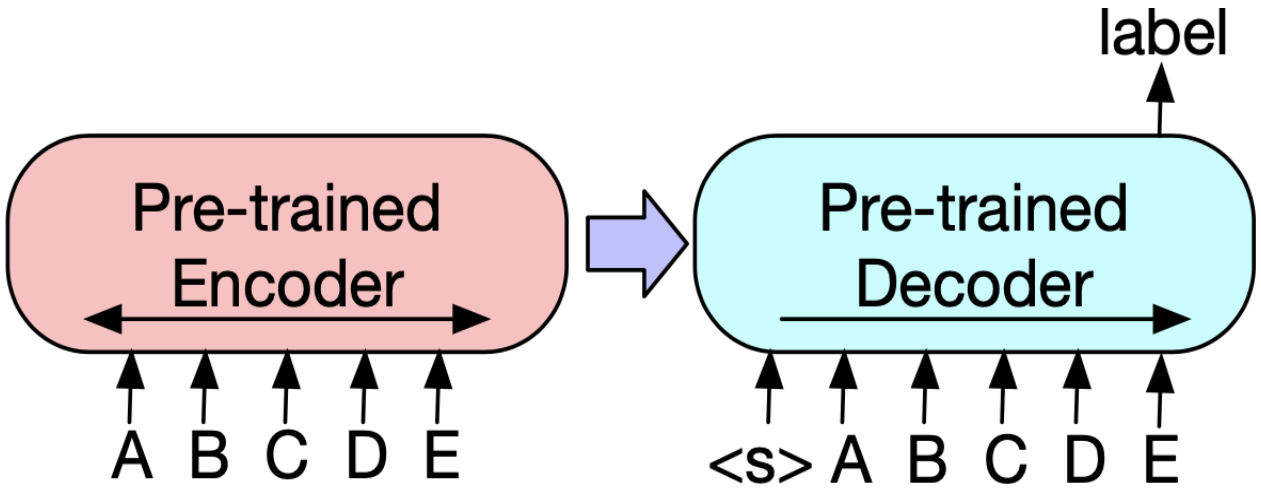
Fine-Tuning BART

Question: How do we fine-tune BART for token classification tasks?

Fine-Tuning BART

Token Classification Tasks

- The same input is fed into the encoder and the decoder
- The top hidden state of the decoder is used as a representation for each token, which is fed into a classifier.



(a) To use BART for classification problems, the same input is fed into the encoder and decoder, and the representation from the final output is used.

BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension, [Lewis et al ACL 2020]

Fine-Tuning BART

Question: How do we fine-tune BART for sequence generation tasks?

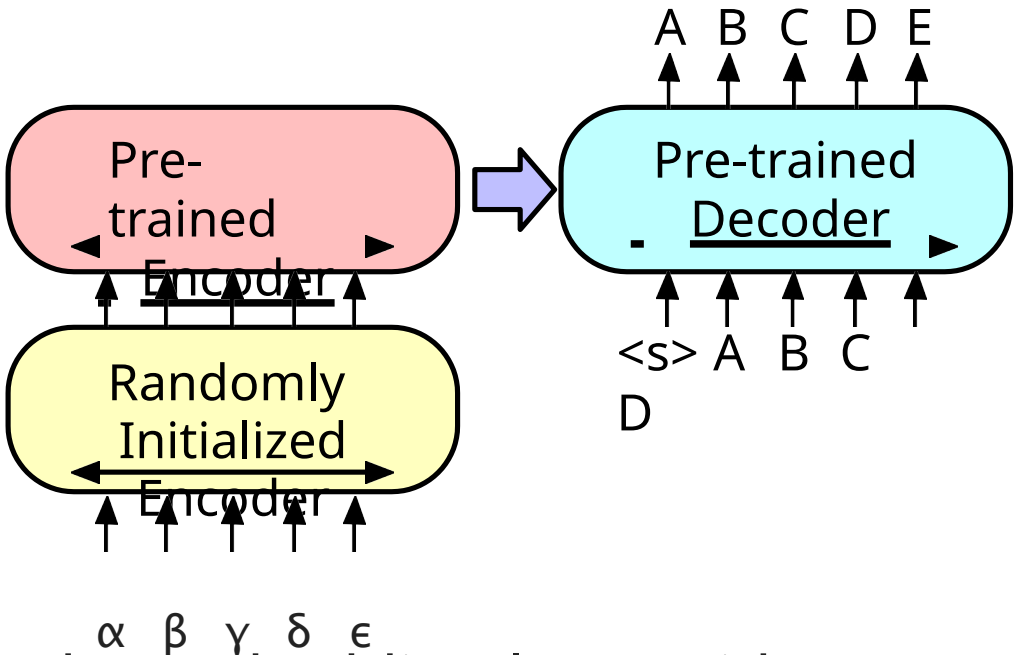
Fine-Tuning BART

Sequence Generation Tasks

- Can be directly fine-tuned for sequence generation tasks because of the autoregressive decoder.
- The input is manipulated: related to the denoising pre-training objective.
- The encoder input is the input sequence, and the decoder generates outputs autoregressively.

BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension, [Lewis et al ACL 2020]

Fine-Tuning BART for Neural Machine Translation



- Replace BART's encoder embedding layer with a new randomly initialized encoder
- The new encoder uses a separate vocabulary from the original BART mode
- First, freeze the BART parameters and only update the randomly initialized source encoder.
- Then, jointly fine-tune for a few steps

BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension, ACL 2020] [Lewis et al

BART: Impact of the Pretraining Objective

Model	SQuAD 1.1 F1	MNLI Acc	ELI5 PPL	XSum PPL	ConvAI2 PPL	CNN/DM PPL
BART Base						
w/ Token Masking	90.4	84.1	25.05	7.08	11.73	6.10
w/ Token Deletion	90.4	84.1	24.61	6.90	11.46	5.87
w/ Text Infilling	**90.8**	84.0	24.26	**6.61**	**11.05**	5.83
w/ Document Rotation	77.2	75.3	53.69	17.14	19.87	10.59
w/ Sentence Shuffling	85.4	81.5	41.87	10.93	16.67	7.89
w/ Text Infilling + Sentence Shuffling	**90.8**	83.8	24.17	6.62	11.12	**5.41**

- Infilling is all-around a bit better than masking or deletion
- Final system: combination of infilling and sentence permutation

BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension, ACL 2020] [Lewis et al

BART for Classification Tasks

	SQuAD 1.1 EM/F1	SQuAD 2.0 EM/F1	MNLI m/mm	SST Acc	QQP Acc	QNLI Acc	STS-B Acc	RTE Acc	MRPC Acc	CoLA Mcc
BERT	84.1/90.9	79.0/81.8	86.6/-	93.2	91.3	92.3	90.0	70.4	88.0	60.6
UniLM	-/-	80.5/83.4	87.0/85.9	94.5	-	92.7	-	70.9	-	61.1
XLNet	**89.0**/94.5	86.1/88.8	89.8/-	95.6	91.8	93.9	91.8	83.8	89.2	63.6
RoBERTa	88.9/**94.6**	**86.5/89.4**	**90.2/90.2**	96.4	92.2	94.7	**92.4**	86.6	**90.9**	**68.0**
BART	88.8/**94.6**	86.1/89.2	89.9/90.1	**96.6**	**92.5**	**94.9**	91.2	**87.0**	90.4	62.8

- The results on GLUE are comparable to RoBERTa
  - suggesting that BART's uni-directional decoder layers do not reduce the performance for discriminative tasks

BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension, ACL 2020] [Lewis et al

BART for NMT

	RO-EN
Baseline	36.80
Fixed BART	36.29
Tuned BART	**37.96**

- Results on IWSLT 2016 English-Romanian
- BART improves over a strong back-translation baseline by using monolingual English pre-training.

BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension, ACL 2020] [Lewis et al

BART for Summarization

	CNN/DailyMail			XSum		
	R1	R2	RL	R1	R2	RL
Lead-3	40.42	17.62	36.67	16.30	1.60	11.95
PTGEN (See et al., 2017)	36.44	15.66	33.42	29.70	9.21	23.24
PTGEN+COV (See et al., 2017)	39.53	17.28	36.38	28.10	8.02	21.72
UniLM	43.33	20.21	40.51	-	-	-
BERTSUMABS (Liu & Lapata, 2019)	41.72	19.39	38.76	38.76	16.33	31.15
BERTSUMEXTABS (Liu & Lapata, 2019)	42.13	19.60	39.18	38.81	16.50	31.27
BART	**44.16**	**21.28**	**40.90**	**45.14**	**22.27**	**37.25**

- This is where BART shines

BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension, ACL 2020] [Lewis et al

BART for Summarization: Example (1)

This is the first time anyone has been recorded to run a full marathon of 42.195 kilometers (approximately 26 miles) under this pursued landmark time. It was not, however, an officially sanctioned world record, as it was not an ”open race” of the IAAF. His time was 1 hour 59 minutes 40.2 seconds. Kipchoge ran in Vienna, Austria. It was an event specifically designed to help Kipchoge break the two hour barrier.

Kenyan runner Eliud Kipchoge has run a marathon in less than two hours.

BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension, ACL 2020] [Lewis et al

BART for Summarization: Example (2)

PG&E stated it scheduled the blackouts in response to forecasts for high winds amid dry conditions. The aim is to reduce the risk of wildfires. Nearly 800 thousand customers were scheduled to be affected by the shutoffs which were expected to last through at least midday tomorrow.

Power has been turned off to millions of customers in California as part of a power shutoff plan.

BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension, ACL 2020] [Lewis et al

From Fine-Tuning to Few-Shot Learning

OpenAI GPT/GPT2

- Very large language models using the Transformer architecture
- Straightforward left-to-right language model, trained on raw text
- GPT2: trained on 40GB of text collected from upvoted links from Reddit
- 1.5B parameters — by far the largest of these models trained when it came out in March 2019
- Because it's a language model, we can **generate** from it

Parameters	Layers	d_{model}
117M	12	768
345M	24	1024
762M	36	1280
1542M	48	1600

Radford et al. (2019)

Pre-Training Cost (with Google/AWS)

- BERT: Base $500, Large $7000
- GPT-2 (as reported in other work): $25,000
- This is for a single pre-training run…developing new pre-training techniques may require many runs
- *Fine-tuning* these models can typically be done with a single GPU (but may take 1-3 days for medium-sized datasets)

https://syncedreview.com/2019/06/27/the-staggering-cost-of-training-sota-ai-models/

Pre-GPT-3: Fine-Tuning

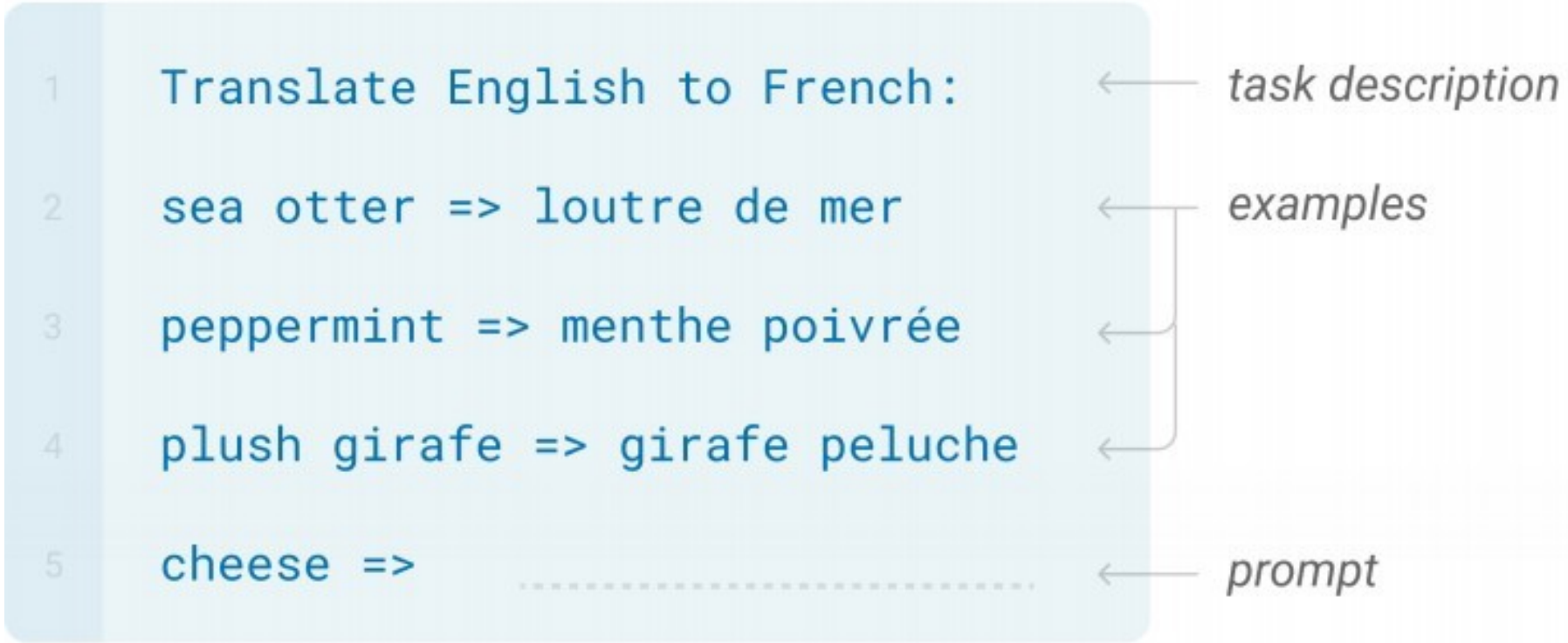
- Fine-tuning: this is the "normal way" of doing learning in models like GPT-2
- Requires computing the gradient and applying a parameter update on every example
- **This is super expensive with 175B parameters**

Brown et al. (2020)

GPT-3: Few-Shot Learning

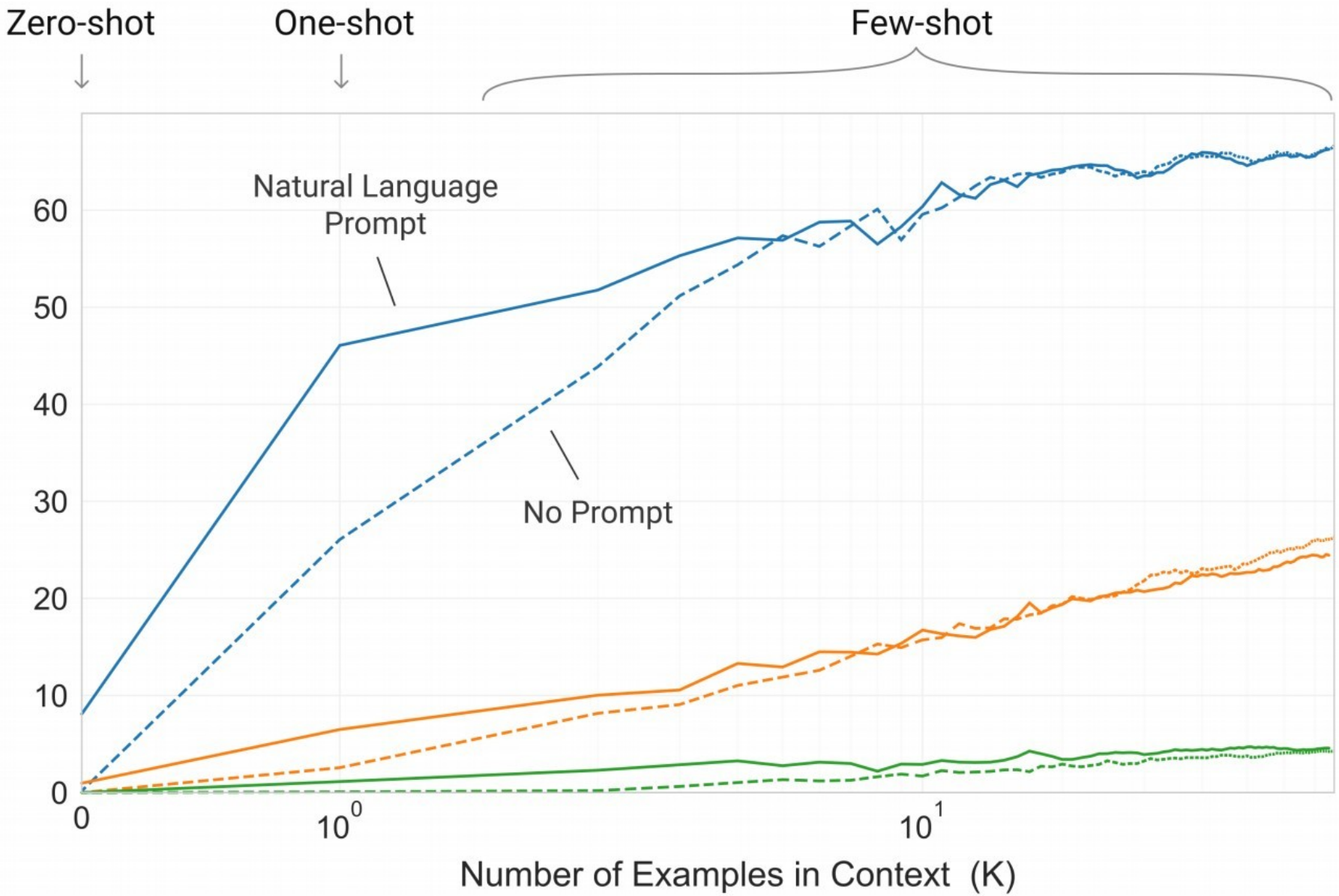
- GPT-3 proposes an alternative: **in-context learning.** Just uses the off-the-shelf model, no gradient updates

- This procedure depends heavily on the examples you pick as well as on the prompt (“*Translate English to French*”)



Brown et al. (2020)

GPT-3



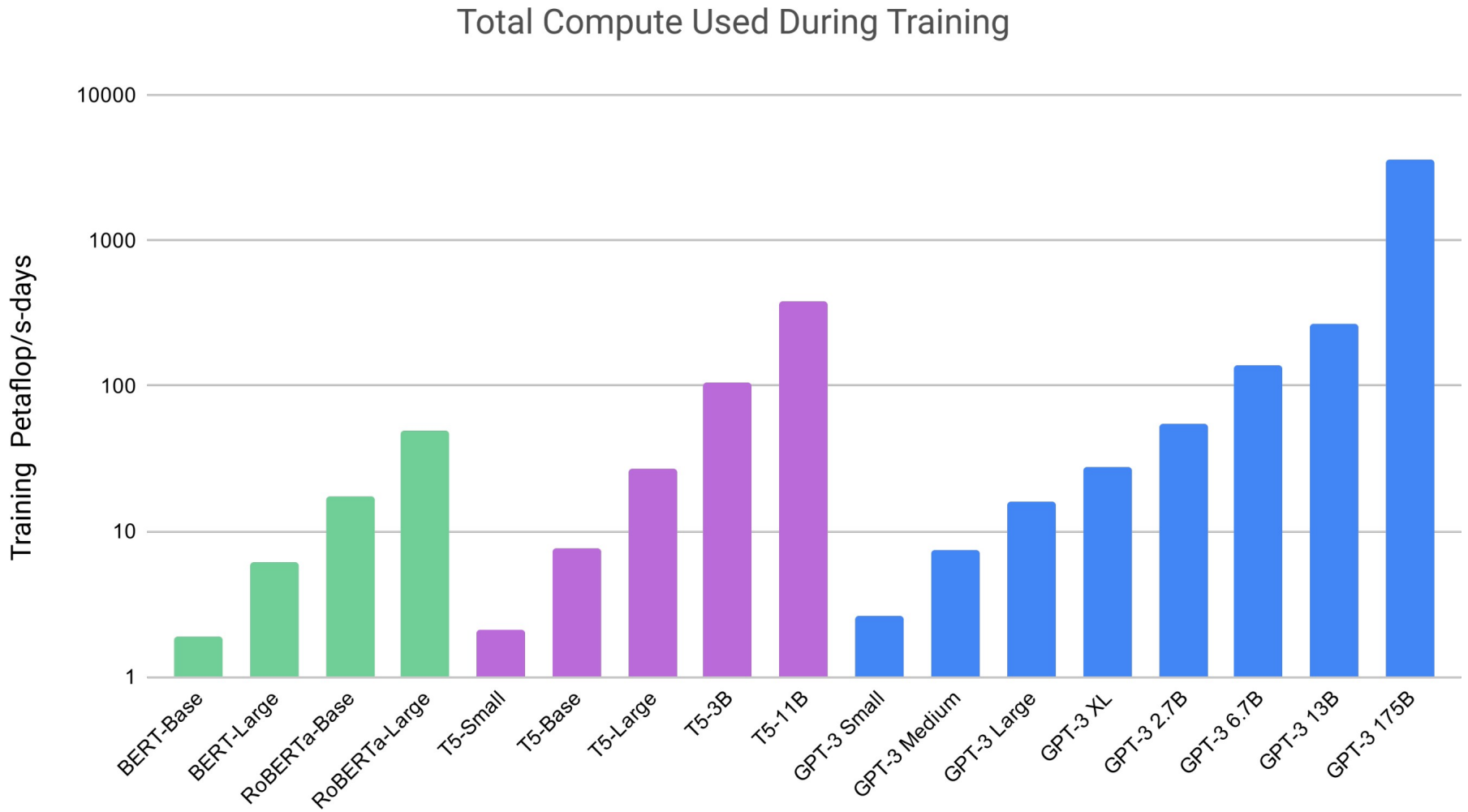
- **Key observation:** few-shot learning only works with huge models!

Brown et al. (2020)

GPT-3: Size and Cost

- GPT-3: 175B parameter model: 96 layers, 96 heads, 12k-dim vectors

- Trained on Microsoft Azure, estimated to cost roughly $10M



Brown et al. (2020)

GPT-3 Results

	SuperGLUE Average	BoolQ Accuracy	CB Accuracy	CB F1	COPA Accuracy	RTE Accuracy
Fine-tuned SOTA	**89.0**	**91.0**	**96.9**	**93.9**	**94.8**	**92.5**
Fine-tuned BERT-Large	69.0	77.4	83.6	75.7	70.6	71.7
GPT-3 Few-Shot	71.8	76.4	75.6	52.0	92.0	69.0

	WiC Accuracy	WSC Accuracy	MultiRC Accuracy	MultiRC F1a	ReCoRD Accuracy	ReCoRD F1
Fine-tuned SOTA	**76.1**	**93.8**	**62.3**	**88.2**	**92.5**	**93.3**
Fine-tuned BERT-Large	69.6	64.6	24.1	70.0	71.3	72.0
GPT-3 Few-Shot	49.4	80.1	30.5	75.4	90.2	91.1

- Sometimes very impressive, (MultiRC, ReCoRD), sometimes very bad
- Results on other datasets are equally mixed — but still strong for a few-shot model!

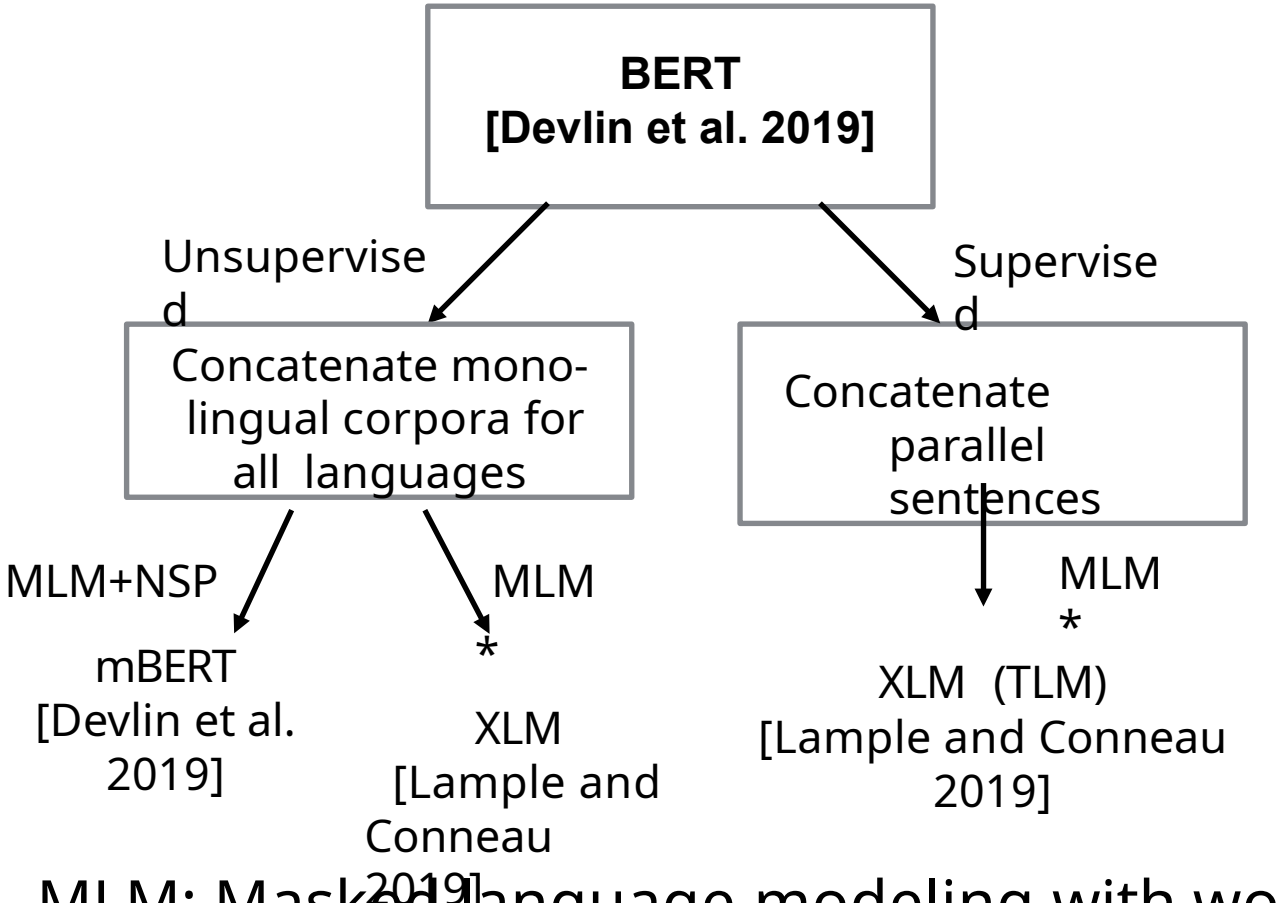
Brown et al. (2020)

Multilingual Transformers: mBERT

Multilingual Pre-Training

- Language model pre-training has shown to be effective for many NLP tasks, e.g., BERT
- BERT uses masked language model (MLM) and next sentence prediction (NSP) objective.
- Models such as mBERT, XLM, XLM-R extend BERT for multi-lingual pre-training

Multilingual Pre-Training



MLM: Masked language modeling with word-piece MLM* : MLM + byte-pair encoding

mBERT: Multilingual BERT

- BERT trained on text from 104 languages
- Wikipedia content
- shared vocabulary across all languages
- to combat the imbalance, small languages were oversampled and large languages undersampled

https://github.com/google-research/bert/blob/master/multilingual.md

Multilingual Transformers: XLM

XLM

- Extended BERT to multiple languages and showed the effectiveness of cross-lingual pretraining
- Introduced a new unsupervised method for learning cross-lingual representations
- Significantly outperformed the previous state of the art on cross-lingual classification, unsupervised MT, and supervised MT

Alexis Conneau, Guillaume Lample. Cross-lingual Language Model Pretraining. NeurIPS 2019: 7057-7067

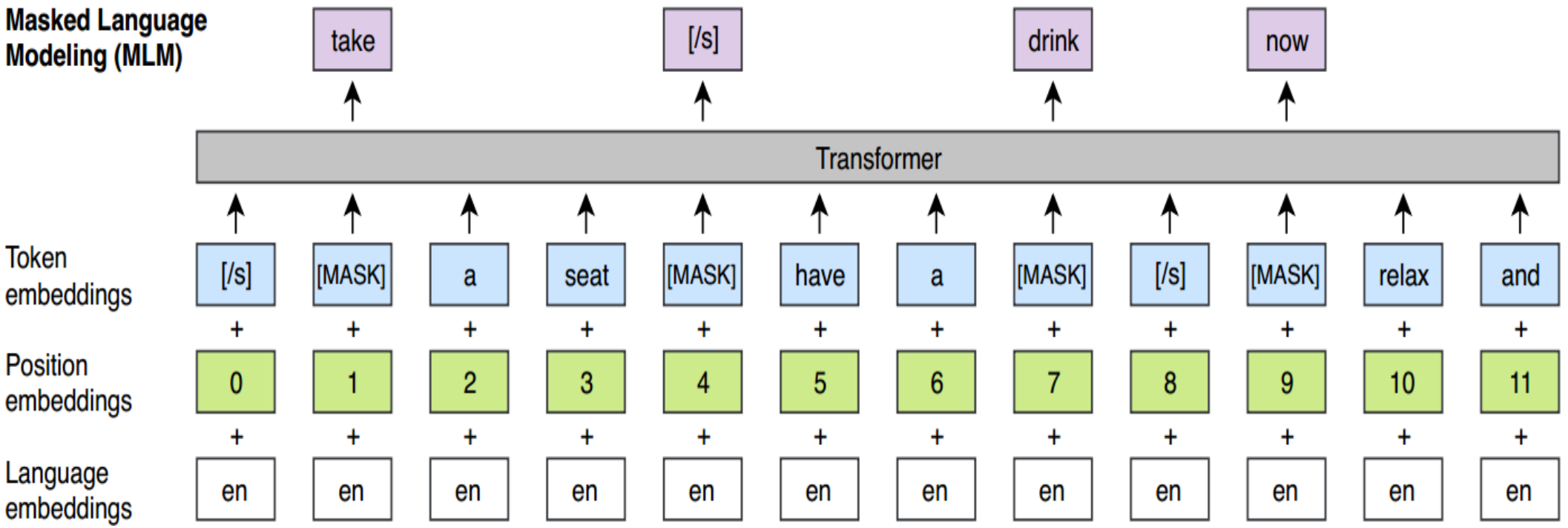
XLM

- Proposed three language modeling objectives
- Two of them only require monolingual data (unsupervised)
- The third one requires parallel sentences (supervised)

XLM

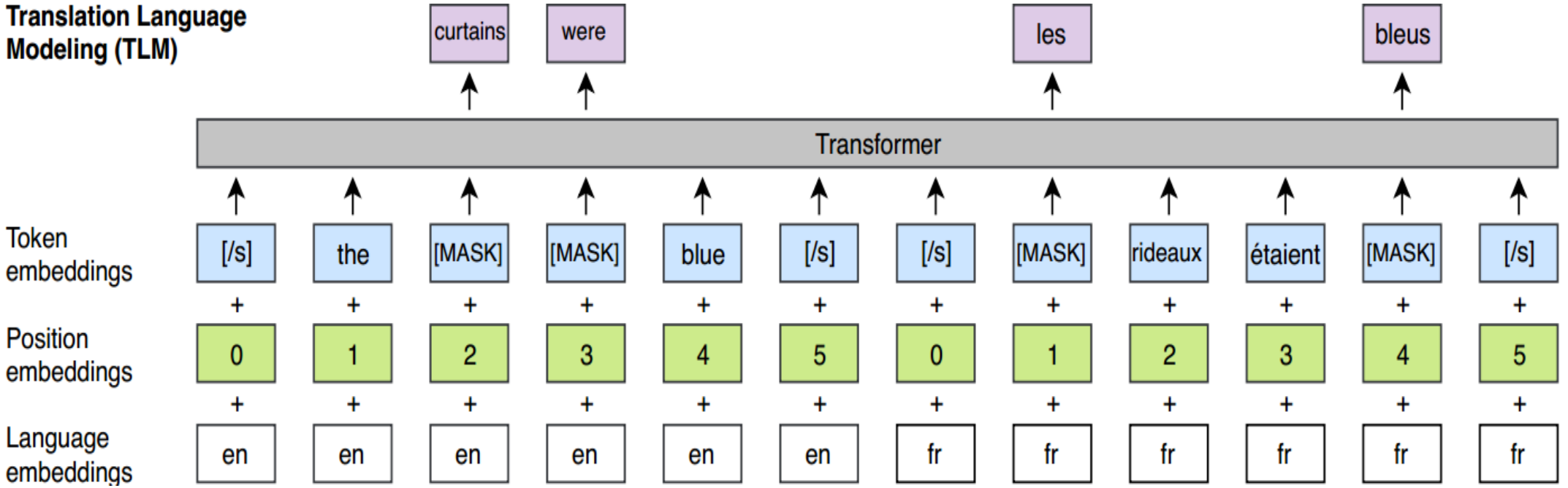
- Process all languages with the same shared vocabulary created through BPE
- Learn BPE splits on the concatenation of sentences sampled randomly from monolingual corpora.
- Sampling increases the number of tokens associated with low-resource languages and alleviates the bias towards high-resource languages

XLM Pre-Training: MLM Objective



- Instead of using pairs of sentences, use a text stream of arbitrary number of sentences (truncated at 256 tokens)
- To counter the imbalance between rare and frequent tokens, subsample the frequent outputs

XLM Pre-Training: TLM Objective



- Extension of MLM: instead of considering monolingual text streams, concatenate parallel sentences
- Randomly mask words in both the source and the target sentences
- For prediction, the model can either attend to surrounding same language words or to the other language words
- This encourages the model to align the English and the French representations

XLM: Fine-Tuning for Cross-Language Classification

- XLM works as a better initialization of sentence encoders for zero-shot cross-lingual classification
- Add a linear classifier on top of the first hidden state of the pretrained Transformer, and fine-tune all parameters on the English NLI training dataset
- Evaluate the capacity of the model to make correct NLI predictions in the 15 XNLI languages

XNLI: Evaluating Cross-lingual Sentence Representations

Language	Premise / Hypothesis	Genre	Label
English	You don’t have to stay there. You can leave.	Face-To-Face	Entailment
French	La figure 4 montre la courbe d’offre des services de partage de travaux. Les services de partage de travaux ont une offre variable.	Government	Entailment
Spanish	Y se estremeció con el recuerdo. El pensamiento sobre el acontecimiento hizo su estremecimiento.	Fiction	Entailment
German	Während der Depression war es die ärmste Gegend, kurz vor dem Hungertod. Die Weltwirtschaftskrise dauerte mehr als zehn Jahre an.	Travel	Neutral
Swahili	Ni silaha ya plastiki ya moja kwa moja inayopiga risasi. Inadumu zaidi kuliko silaha ya chuma.	Telephone	Neutral
Russian	И мы занимаемся этим уже на протяжении 85 лет. Мы только начали этим заниматься.	Letters	Contradiction
Chinese	让我告诉你，美国人最终如何看待你作为独立顾问的表现。 美国人完全不知道您是独立律师。	Slate	Contradiction
Arabic	تحتاج الوكالات لأن تكون قادرة على قياس مستويات النجاح. لا يمكن للوكالات أن تعرف ما إذا كانت ناجحة أم لا.	Nine-Eleven	Contradiction

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel R. Bowman, Holger Schwenk, Veselin Stoyanov. XNLI: Evaluating Cross-lingual Sentence Representations. EMNLP 2018: 2475-2485

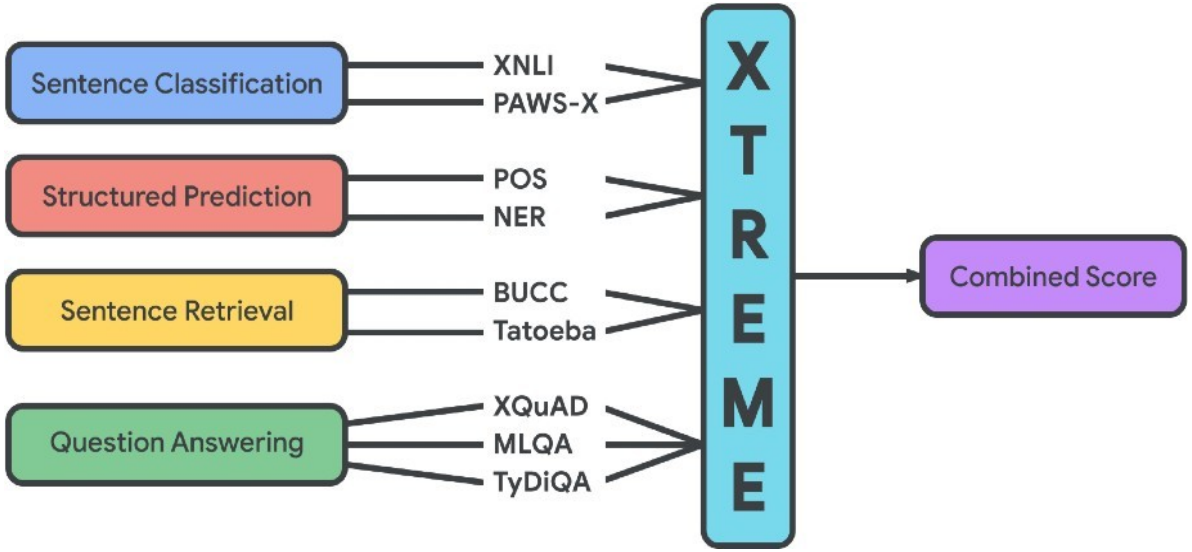
XLM: Evaluation on XNLI

	en	fr	es	de	el	bg	ru	tr	ar	vi	th	zh	hi	sw	ur	Δ
Machine translation baselines (TRANSLATE-TRAIN)																
Devlin et al. (2018)	81.9	-	77.8	75.9	-	-	-	-	70.7	-	-	76.6	-	-	61.6	-
XLM (MLM+TLM)	85.0	80.2	80.8	80.3	78.1	79.3	78.1	74.7	76.5	76.6	75.5	78.6	72.3	70.9	63.2	76.7
Machine translation baselines (TRANSLATE-TEST)																
Devlin et al. (2018)	81.4	-	74.9	74.4	-	-	-	-	70.4	-	-	70.1	-	-	62.1	-
XLM (MLM+TLM)	85.0	79.0	79.5	78.1	77.8	77.6	75.5	73.7	73.7	70.8	70.4	73.6	69.0	64.7	65.1	74.2
Evaluation of cross-lingual sentence encoders																
Conneau et al. (2018b)	73.7	67.7	68.7	67.7	68.9	67.9	65.4	64.2	64.8	66.4	64.1	65.8	64.1	55.7	58.4	65.6
Devlin et al. (2018)	81.4	-	74.3	70.5	-	-	-	-	62.1	-	-	63.8	-	-	58.3	-
Artetxe and Schwenk (2018)	73.9	71.9	72.9	72.6	73.1	74.2	71.5	69.7	71.4	72.0	69.2	71.4	65.5	62.2	61.0	70.2
XLM (MLM)	83.2	76.5	76.3	74.2	73.1	74.0	73.1	67.8	68.5	71.2	69.2	71.9	65.7	64.6	63.4	71.5
XLM (MLM+TLM)	**85.0**	**78.7**	**78.9**	**77.8**	**76.6**	**77.4**	**75.3**	**72.5**	**73.1**	**76.1**	**73.2**	**76.5**	**69.6**	**68.4**	**67.3**	**75.1**

Multilingual Representation Evaluation

Large-scale benchmarks that cover many tasks

XTREME: 40 languages, 9 tasks (Hu et al. 2020)



XGLUE: less typologically diverse, but contains generation (Liang et al. 2020)

XTREME-R harder version based on XTREME (Ruder et al. 2021)

XLM as Pre-Training for NMT

Neural Machine Translation (NMT)

- XLM works as a better initialization of supervised and unsupervised NMT systems
- Pretrain entire encoder and decoder with a cross-lingual language model

XLM as Pre-Training for Unsupervised NMT

		en-fr	fr-en	en-de	de-en	en-ro	ro-en
Previous state-of-the-art - Lample et al. (2018b)							
NMT		25.1	24.2	17.2	21.0	21.2	19.4
PBSMT		28.1	27.2	17.8	22.7	21.3	23.0
PBSMT + NMT		27.6	27.7	20.2	25.2	25.1	23.9
Our results for different encoder and decoder initializations							
EMB	EMB	29.4	29.4	21.3	27.3	27.5	26.6
-	-	13.0	15.8	6.7	15.3	18.9	18.3
-	CLM	25.3	26.4	19.2	26.0	25.7	24.6
-	MLM	29.2	29.1	21.6	28.6	28.2	27.3
CLM	-	28.7	28.2	24.4	30.3	29.2	28.0
CLM	CLM	30.4	30.0	22.7	30.5	29.0	27.8
CLM	MLM	32.3	31.6	24.3	32.5	31.6	29.8
MLM	-	31.6	32.1	**27.0**	33.2	31.8	30.5
MLM	CLM	**33.4**	32.3	24.9	32.9	31.7	30.4
MLM	MLM	**33.4**	**33.3**	26.4	**34.3**	**33.3**	**31.8**

BLEU scores on WMT'14 English-French, WMT'16 German-English and WMT'16 RomanianEnglish. For our results, the first two columns indicate the model used to pretrain the encoder and the decoder. " - " means the model was randomly initialized. EMB corresponds to pretraining the loodsdkup table with cross-lingual embeddings, CLM and MLM correspond to pretraining with models trained on the CLM or MLM objectives. **CLM is Conditional LM.**

XLM as Pre-Training for Supervised NMT

Pretraining	-	CLM	MLM
Sennrich et al. (2016)	33.9	-	-
ro → en	28.4	31.5	35.3
ro ↔ en	28.5	31.5	35.6
ro ↔ en + BT	34.4	37.0	**38.5**

BLEU scores on WMT'16 Romanian-English. The previous state-of-the-art of Sennrich et al. (2016) uses both back-translation and an ensemble model. ro ↔ en corresponds to models trained on both directions.

Notes on Multilinguality

Cross-Lingual Zero-Shot Learning

- We are given labeled training data for task X only in language A. Can we build a model that can make predictions for task X in a different language B?

- **Idea**: leverage information from high-resource languages to help improve performance on low-resource languages.

- **Zero-shot** learning: no labeled data is available for the target task X in language B, although unlabeled data in language B might be available for pretraining

Zero-Shot XNLI: Training on English Only

Model	Sentence pair	
	XNLI	PAWS-X
Metrics	Acc.	Acc.
Cross-lingual zero-shot transfer (models fine-tu		
mBERT	65.4	81.9
XLM	69.1	80.9
InfoXLM	81.4	-
X-STILTs	80.4	87.7
XLM-R	79.2	86.4
VECO	79.9	88.7
RemBERT	80.8	87.5
mT5-Small	67.5	82.4
mT5-Base	75.4	86.4
mT5-Large	81.1	88.9
mT5-XL	82.9	89.6
mT5-XXL	**85.0**	**90.0**

What if we use a machine translation system to get more labeled data (e.g., translate all the labeled English text to other languages)?

Adding Translations Does Not Improve Much Over Zero-Shot Setting!

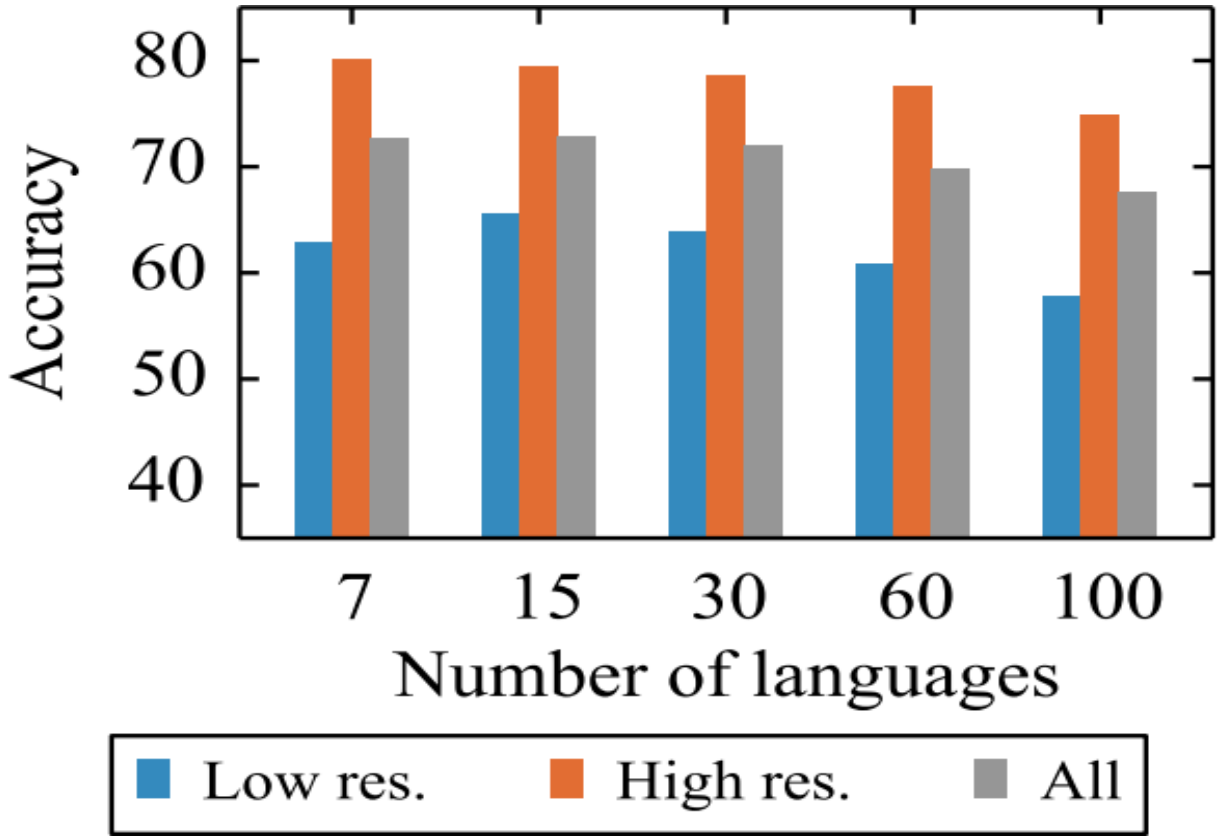
Model	Sentence pair	
	XNLI	PAWS-X
Metrics	Acc.	Acc.
Cross-lingual zero-shot transfer (models fine-tu		
mBERT	65.4	81.9
XLM	69.1	80.9
InfoXLM	81.4	-
X-STILTs	80.4	87.7
XLM-R	79.2	86.4
VECO	79.9	88.7
RemBERT	80.8	87.5
mT5-Small	67.5	82.4
mT5-Base	75.4	86.4
mT5-Large	81.1	88.9
mT5-XL	82.9	89.6
mT5-XXL	**85.0**	**90.0**

Translate-train (models fine-tuned on English		
XLM-R	82.6	90.4
FILTER + Self-Teaching	83.9	91.4
VECO	83.0	91.1
mT5-Small	64.7	79.9
mT5-Base	75.9	89.3
mT5-Large	81.8	91.2
mT5-XL	84.8	91.0
mT5-XXL	**87.8**	**91.5**

What if a language is unseen or poorly represented during pretraining?

The “Curse of Multilinguality” (Conneau et al., 2020)

For a fixed-size model, the per-language capacity decreases as we increase the number of languages…



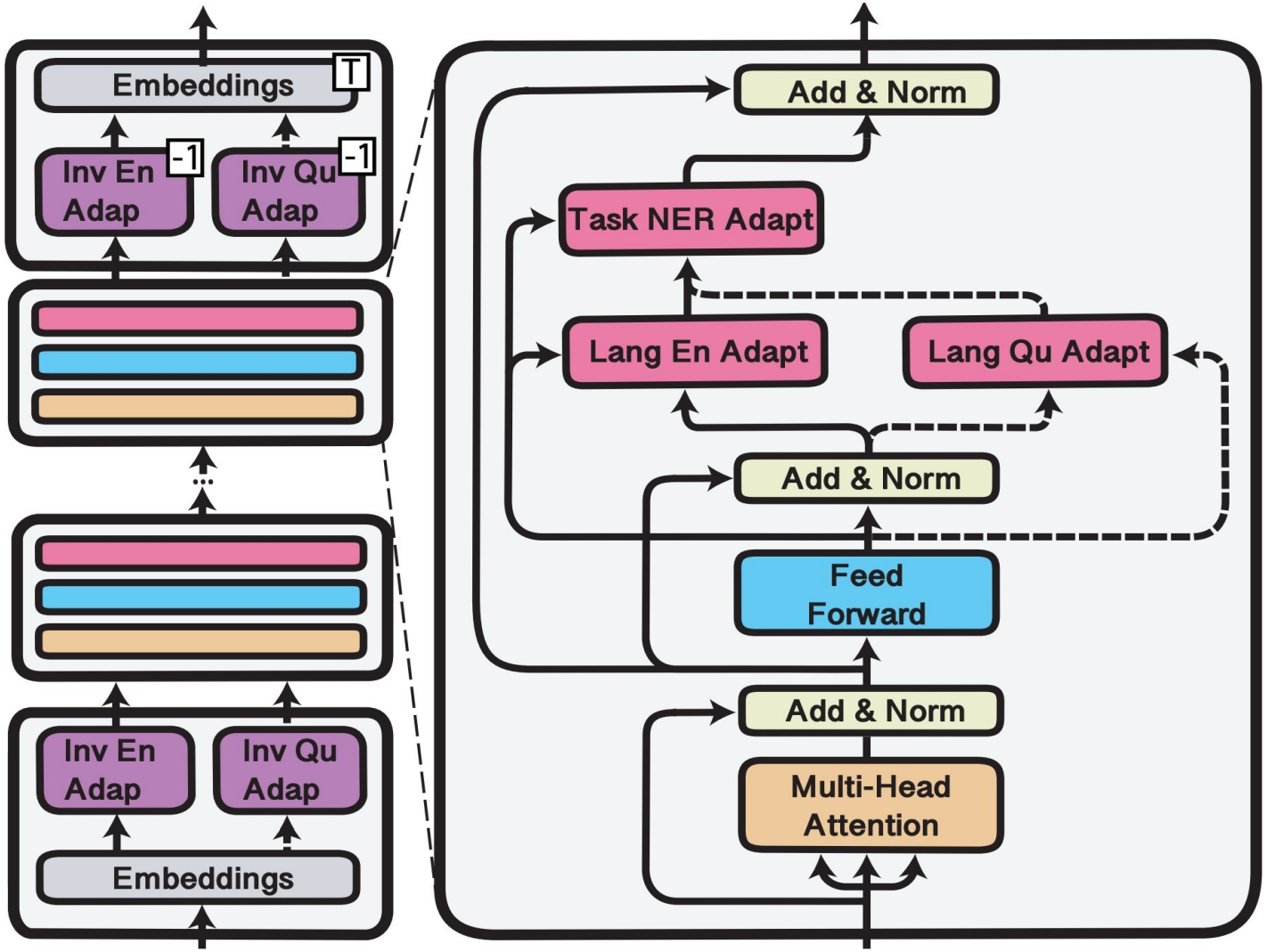
Target Language Adaptation

- If we only care about transferring to a specific target language B, then after normal pretraining on many languages, we can perform a second phase of fine-tuning on only unlabeled data from language B

- However, doing this might result in *catastrophic forgetting* of multilingual knowledge learned during the first stage of pretraining.

- **Solution:** train a small number of parameters in the second phase!

Target Language Adaptation with Adapters



MAD-X paper: (Pfeiffer et al., EMNLP 2020)

Target Language Adaptation with Adapters

Language	ISO code	Language family	# of Wiki articles	Covered by SOTA?
English	en	Indo-European	6.0M	✓
Japanese	ja	Japonic	1.2M	✓
Chinese	zh	Sino-Tibetan	1.1M	✓
Arabic	ar	Afro-Asiatic	1.0M	✓
Javanese	jv	Austronesian	57k	✓
Swahili	sw	Niger-Congo	56k	✓
Icelandic	is	Indo-European	49k	✓
Burmese	my	Sino-Tibetan	45k	✓
Quechua	qu	Quechua	22k	
Min Dong	cdo	Sino-Tibetan	15k	
Ilokano	ilo	Austronesian	14k	
Mingrelian	xmf	Kartvelian	13k	
Meadow Mari	mhr	Uralic	10k	
Maori	mi	Austronesian	7k	
Turkmen	tk	Turkic	6k	
Guarani	gn	Tupian	4k	

Table 1: Languages in our NER evaluation.

MAD-X paper: (Pfeiffer et al., EMNLP 2020)

Target Language Adaptation with Adapters

Model	en	ja	zh	ar	jv	sw	is	my	qu	cdo	ilo	xmf	mi	mhr	tk	gn	avg
XLM-R^{Base}	44.2	38.2	40.4	36.4	37.4	42.8	47.1	**26.3**	27.4	18.1	28.8	**35.0**	16.7	**31.7**	20.6	**31.2**	32.6
XLM-R^{Base} MLM-SRC	39.5	45.2	34.7	17.7	34.5	35.3	43.1	20.8	26.6	21.4	28.7	22.4	18.1	25.0	27.6	24.0	29.0
XLM-R^{Base} MLM-TRG	54.8	**47.4**	**54.7**	51.1	38.7	48.1	53.0	20.0	29.3	16.6	27.4	24.7	15.9	26.4	26.5	28.5	35.2
MAD-X^{Base} – LAD – INV	44.5	38.6	40.6	42.8	32.4	43.1	48.6	23.9	22.0	10.6	23.9	27.9	13.2	24.6	18.8	21.9	29.8
MAD-X^{Base} – INV	52.3	46.0	46.2	56.3	**41.6**	48.6	52.4	23.2	32.4	**27.2**	30.8	33.0	**23.5**	29.3	30.4	28.4	37.6
MAD-X^{Base}	**55.0**	46.7	47.3	**58.2**	39.2	**50.4**	**54.5**	24.9	**32.6**	24.2	**33.8**	34.3	16.8	**31.7**	**31.9**	30.4	**38.2**
mBERT	48.6	50.5	50.6	50.9	45.3	48.7	51.2	17.7	31.8	20.7	33.3	26.1	**20.9**	31.3	**34.8**	**30.9**	37.1
MAD-X^{mBERT}	**52.8**	**53.1**	**53.2**	**55.5**	**46.3**	**50.9**	**51.4**	**21.0**	**37.7**	**22.1**	**35.0**	**30.0**	18.6	**31.8**	33.0	25.1	**38.6**
XLM-R^{Large}	47.10	46.52	46.43	45.15	39.21	43.96	48.69	**26.18**	26.39	15.12	22.80	**33.67**	19.86	27.70	29.56	**33.78**	34.6
MAD-X^{Large}	**56.30**	**53.37**	**55.6**	**59.41**	**40.40**	**50.57**	**53.22**	24.55	**33.89**	**26.54**	**30.98**	33.37	**24.31**	**28.03**	**30.82**	26.38	**39.2**

Table 2: NER F1 scores averaged over all 16 target languages when transferring from each source language (i.e. the columns are source languages). The vertical dashed line distinguishes between languages seen in multilingual pretraining and the unseen ones (see also Table 1).

MAD-X paper: (Pfeiffer et al., EMNLP 2020)

Target Language Adaptation with Adapters

Model	en	et	ht	id	it	qu	sw	ta	th	tr	vi	zh	avg
XLM-R^{Base}	66.8	58.0	51.4	65.0	60.2	51.2	52.0	58.4	62.0	56.6	65.6	**68.8**	59.7
XLM-R^{Base} MLM-TRG	66.8	59.4	50.0	**71.0**	61.6	46.0	**58.8**	60.0	**63.2**	**62.2**	**67.6**	67.4	61.2
MAD-X^{Base}	**68.3**	**61.3**	**53.7**	65.8	**63.0**	**52.5**	56.3	**61.9**	61.8	60.3	66.1	67.6	**61.5**

Table 3: Accuracy scores of all models on the XCOPA test sets when transferring from English. Models are first fine-tuned on SIQA and then on the COPA training set.

	en	ar	de	el	es	hi	ru	th	tr	vi	zh	avg
XLM-R^{Base}	83.6 / 72.1	66.8 / 49.1	**74.4 / 60.1**	73.0 / **55.7**	76.4 / **58.3**	68.2 / 51.7	**74.3 / 58.1**	66.5 / **56.7**	68.3 / 52.8	73.7 / 53.8	51.3 / 42.0	70.6 / 55.5
XLM-R^{Base} MLM-TRG	**84.7 / 72.6**	**67.0 / 49.2**	73.7 / 58.8	**73.2** / 55.7	**76.6 / 58.3**	**69.8 / 53.6**	**74.3** / 57.9	67.0 / 55.8	**68.6 / 53.0**	**75.5 / 54.9**	52.2 / **43.1**	**71.1 / 55.7**
MAD-X^{Base} – INV	83.3 / 72.1	64.0 / 47.1	72.0 / 55.8	71.0 / 52.9	74.6 / 55.5	67.3 / 51.0	72.1 / 55.1	64.1 / 51.8	66.2 / 49.6	73.0 / 53.6	50.9 / 40.6	67.0 / 53.2
MAD-X^{Base}	83.5 / **72.6**	65.5 / 48.2	72.9 / 56.0	72.9 / 54.6	75.9 / 56.9	68.2 / 51.3	73.1 / 56.7	**67.8** / 55.9	67.0 / 49.8	73.7 / 53.3	**52.7** / 42.8	70.3 / 54.4

Table 4: F_1 / EM scores on XQuAD with English as the source language for each target language.

MAD-X paper: (Pfeiffer et al., EMNLP 2020)