Pretraining Sequence-to-Sequence Models: **BART + T5**

Antoine Bosselut

EPFL

nlp lab

Announcements

- **Course Feedback:** Indicative Feedback this week! Please fill it out!
- **Assignment 1:** Due tomorrow, 03/24/2023 at 11:59 PM
    - Additional Office Hours (today)
    - Please provide feedback on Ed discussion board (anonymously if you wish)
- **Assignment 2:** Released Saturday, 03/25/2023 at 12:00 AM
- **Course Project:** Description to be released next week!

Today's Outline

- **Lecture**
  - **Quick Recap:** Pretraining + Finetuning
  - **Pretraining sequence-to-sequence models:** BART + T5
  - A1 Review
- **Exercise Session**
  - **Review of Week 4 Exercise Session:** Transformers + Decoding
  - **Week 5 Exercise Session:** Finetuning pretrained models

Some slides adapted from Greg Durrett

Transfer Learning

Pretraining

Learn embeddings that can be used to seed a downstream model (ELMo)

-or-

Learn a model that can be fine-tuned for many downstream tasks (GPT, BERT)

Fine-tuning

Design a new model architecture whose embeddings are initialised with pretrained embeddings. Train this model on a task of interest

- or -

Take a pretrained model and train it further on data from a task of interest

Transfer Learning

Pretraining

Learn embeddings that can be used to seed a downstream model (ELMo)

-or-

Learn a model that can be fine-tuned for many downstream tasks (GPT, BERT)

Fine-tuning

Design a new model architecture whose embeddings are initialised with pretrained embeddings. Train this model on a task of interest

- or -

Take a pretrained model and train it further on data from a task of interest

Transfer Learning

Pretraining

Learn embeddings that can be used to seed a downstream model (ELMo)

-or-

Learn a model that can be fine-tuned for many downstream tasks (GPT, BERT)

Fine-tuning

Design a new model architecture whose embeddings are initialised with pretrained embeddings. Train this model on a task of interest

- or -

Take a pretrained model and train it further on data from a task of interest

Transfer Learning

Pretraining

Uses simple training objectives

Requires tons of data

Resultant model often not useful yet

Slow & expensive; can often only do once

Fine-tuning

Done on smaller datasets

Trained on data with a more complex structure

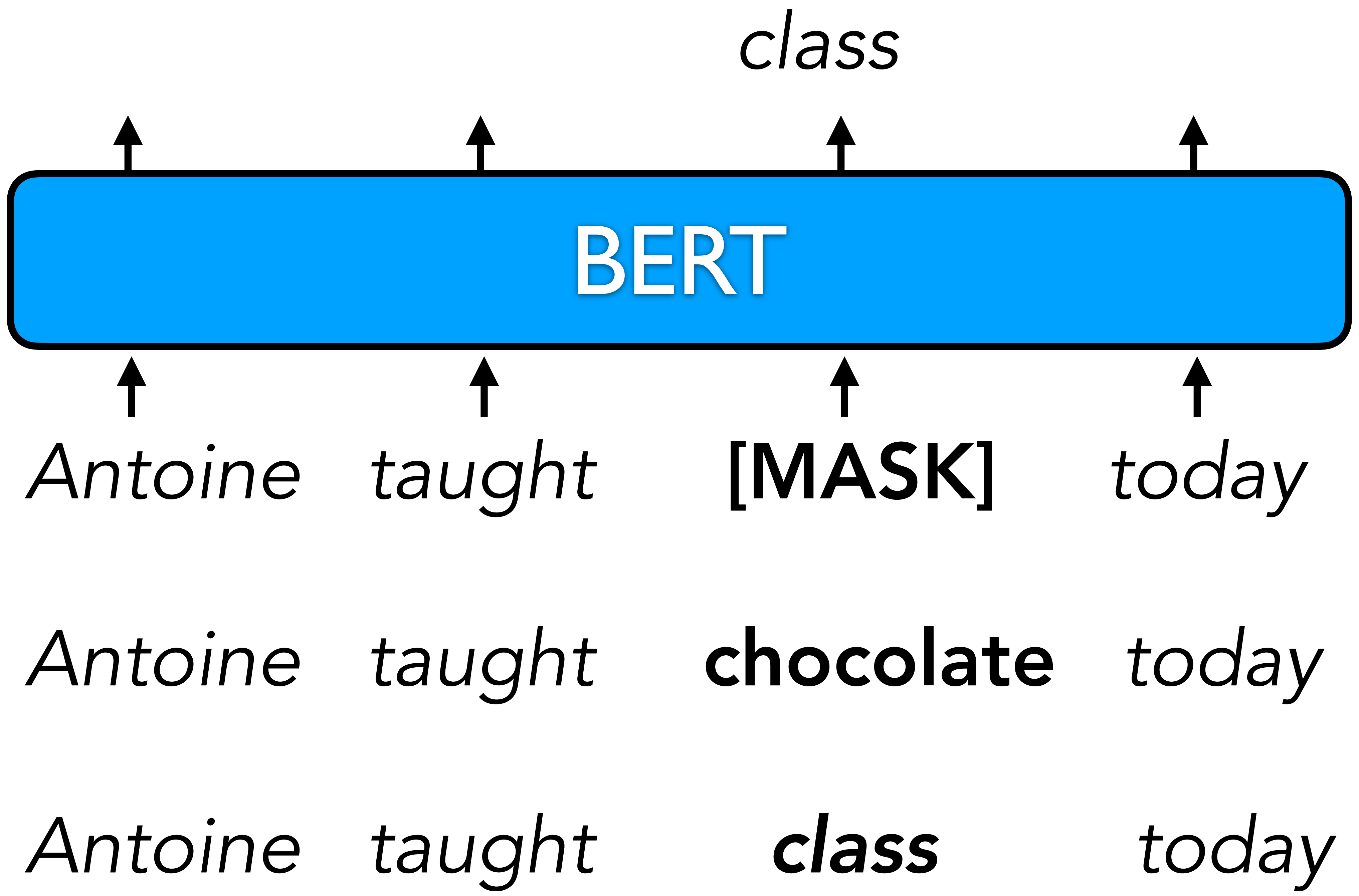
Resultant model applied to task of interest

Typically cheaper; can afford multiple runs, hyper parameter tuning, etc.

Pretraining BERT

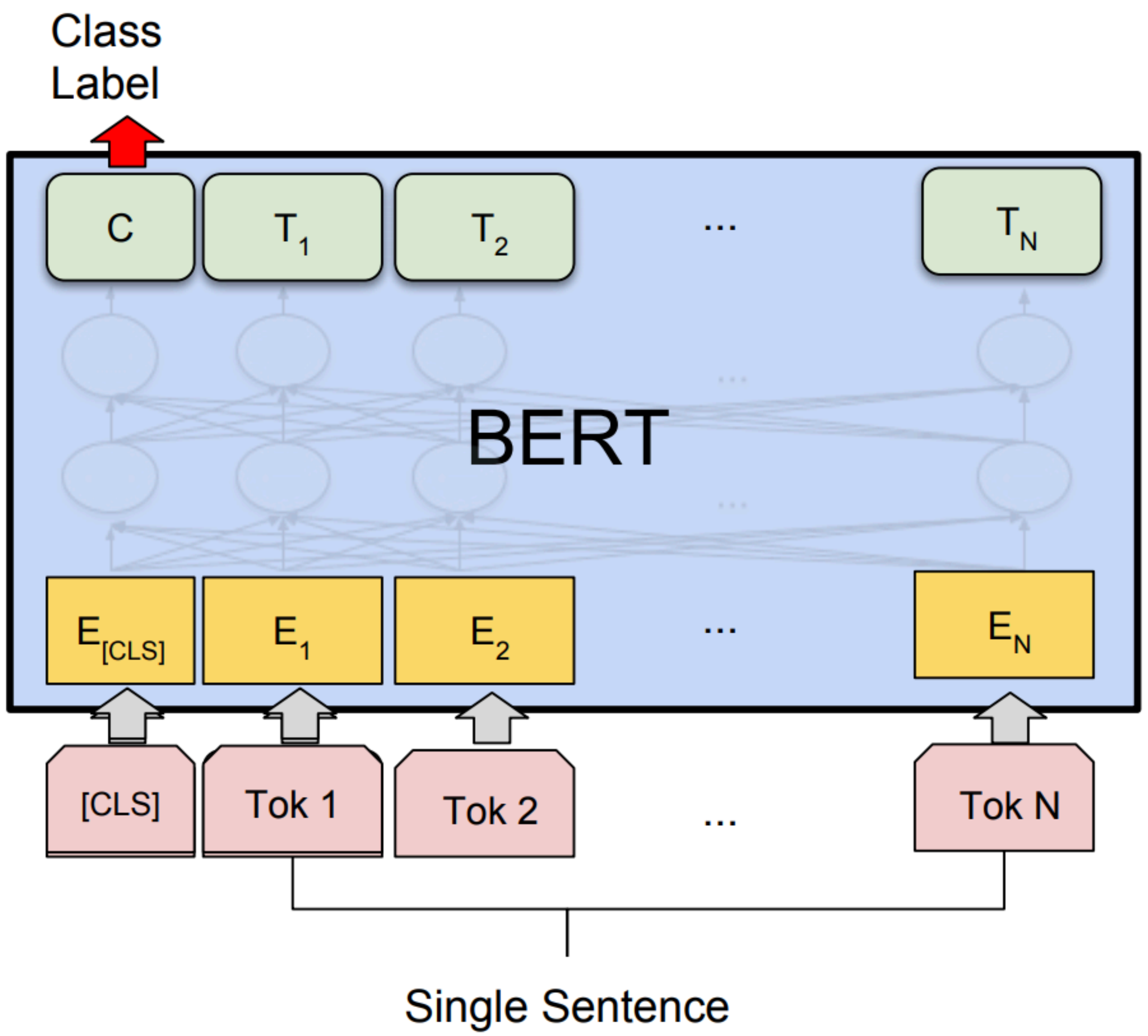
- **Pretraining (self-supervised learning)**:
  - Done at scale on natural occurring sequences of text (any large corpus of raw text)
  - Take a sequence of text, and predict 15% of the tokens

- **For 15% of tokens:**
  - Replace input token with [MASK] (80% of predictions)
  - Replace input token with a random token (10% of predictions)
  - Keep the same input token (10% of predictions)



Devlin et al. (2019)

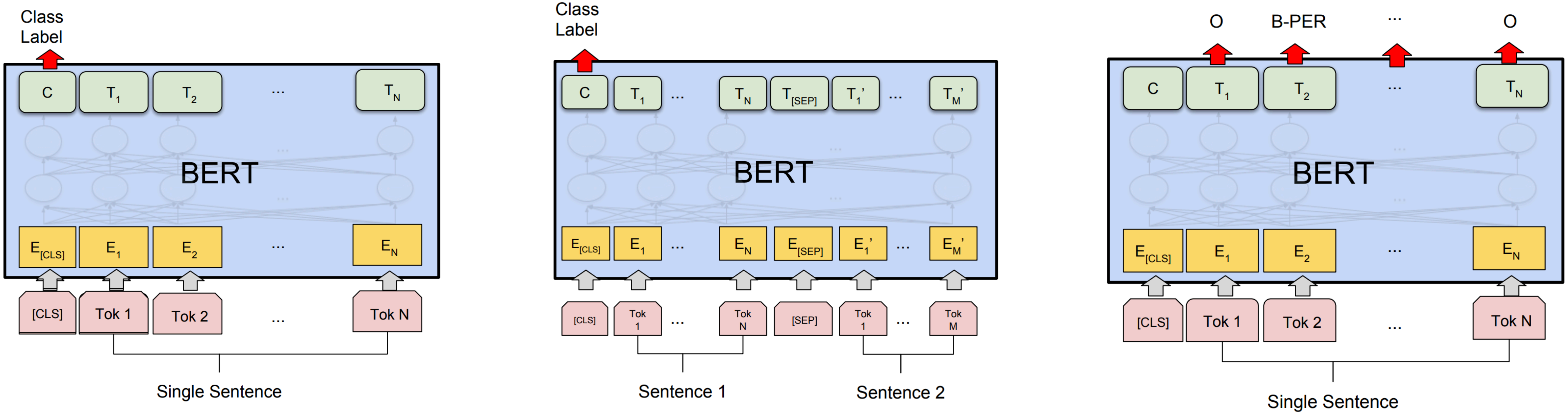
Fine-tuning BERT



- Done after BERT has been pretrained (no more pretraining objectives)
- Select a task with supervised data (i.e., classification for sentiment analysis)
- Prepend a special token **[CLS]** to the front of the sequence to classify
- **Learn** to classify the output embedding for this token
- **During fine-tuning**, we update the parameters of the BERT model to learn the task

Devlin et al. (2019)

Single model starting point for many tasks



(b) Single Sentence Classification Tasks: SST-2, CoLA

(a) Sentence Pair Classification Tasks: MNLI, QQP, QNLI, STS-B, MRPC, RTE, SWAG

(d) Single Sentence Tagging Tasks: CoNLL-2003 NER

- Re-using the same pretrained BERT model for fine-tuning on many tasks:
  - **Classification**: Take [CLS] output embedding as input features to classification model
  - **Sequence labeling**: Take output embedding for each token and classify individually

Devlin et al. (2019)

BERT on GLUE

System	MNLI-(m/mm)	QQP	QNLI	SST-2	CoLA	STS-B	MRPC	RTE	**Average**
	392k	363k	108k	67k	8.5k	5.7k	3.5k	2.5k	-
Pre-OpenAI SOTA	80.6/80.1	66.1	82.3	93.2	35.0	81.0	86.0	61.7	74.0
BiLSTM+ELMo+Attn	76.4/76.1	64.8	79.9	90.4	36.0	73.3	84.9	56.8	71.0
OpenAI GPT	82.1/81.4	70.3	88.1	91.3	45.4	80.0	82.3	56.0	75.2
$\text{BERT}_{\text{BASE}}$	84.6/83.4	71.2	90.1	93.5	52.1	85.8	88.9	66.4	79.6
$\text{BERT}_{\text{LARGE}}$	**86.7/85.9**	**72.1**	**91.1**	**94.9**	**60.5**	**86.5**	**89.3**	**70.1**	**81.9**

For each of these tasks, a different BERT model is fine-tuned on the task data

Not the same fine-tuned BERT model that gets the same performance

Devlin et al. (2019)

Recap

- **Contextual representations:** Let us model words and sequences conditioned on the context around them

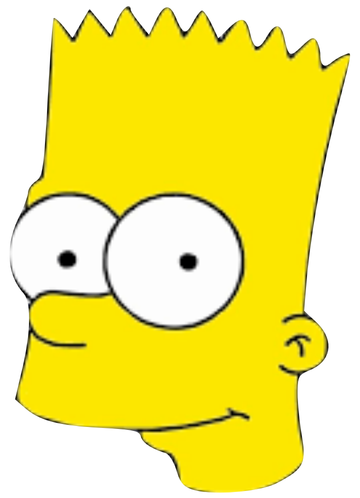
- **ELMo:** Based on bidirectional LSTMs. **Good for pretrained embeddings.**

- **GPT**: Uses a transformer decoder. **Good for generating text as a language model.**

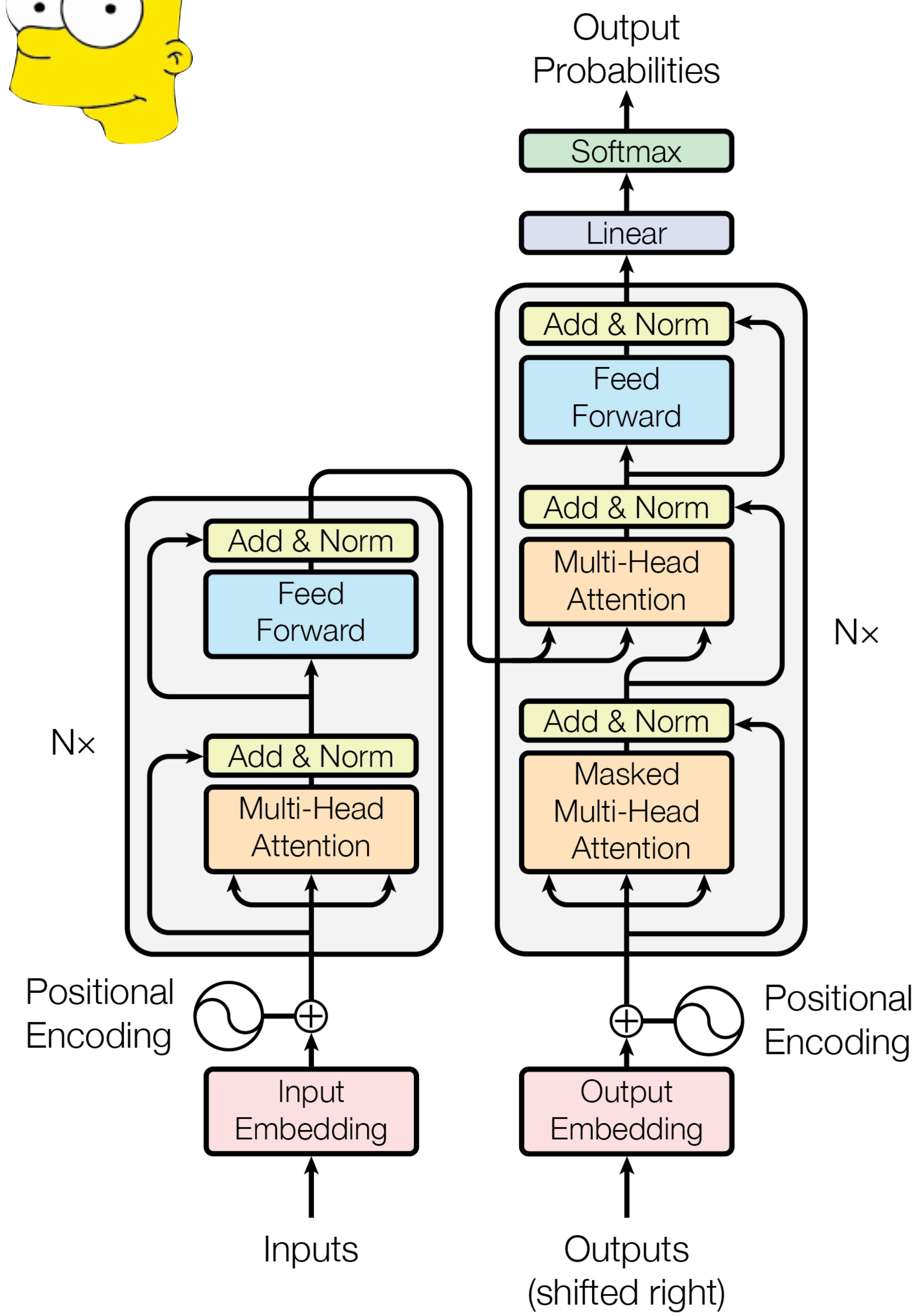
- **BERT**: Uses a transformer encoder. **Good for classification and sequence labelling.**

We've seen encoders and decoders.
What type of model have we not seen pretraining for yet?

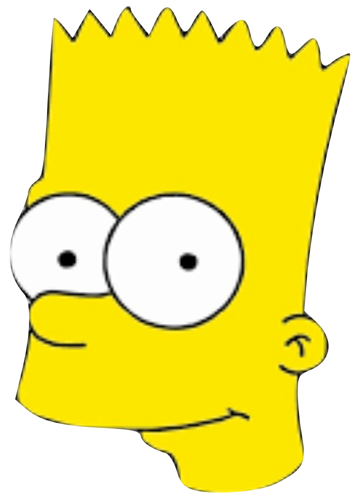
How should we pretrain sequence-to-sequence models?

BART 

- Classic transformer architecture
- Bidirectional encoder feeds into autoregressive decoder
- Cross-attention layers in decoder are back!
- **BART-base**: 6-layers each in encoder and decoder; 140M parameters
- **BART-large**: 12 layers each in encoder and decoder; 400M parameters

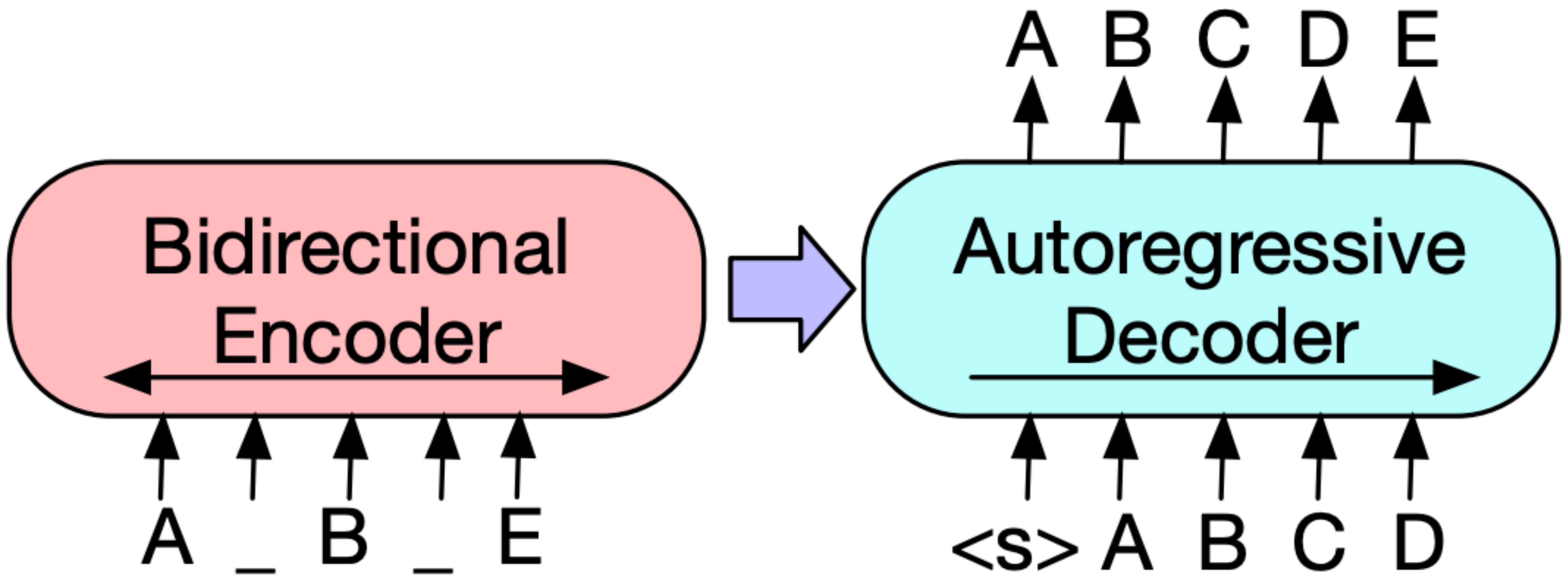


Lewis et al. (2019)

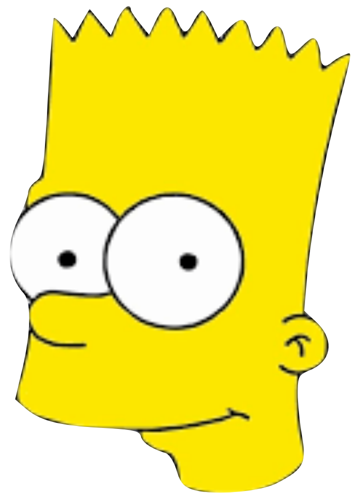


BART Pretraining

- Pretraining BART combines elements of BERT and GPT!
- **BERT-style:** input texts corrupted before they are passed to bidirectional encoder
- **GPT-style:** model is trained with a language modelling objective in the decoder: predict the next word!

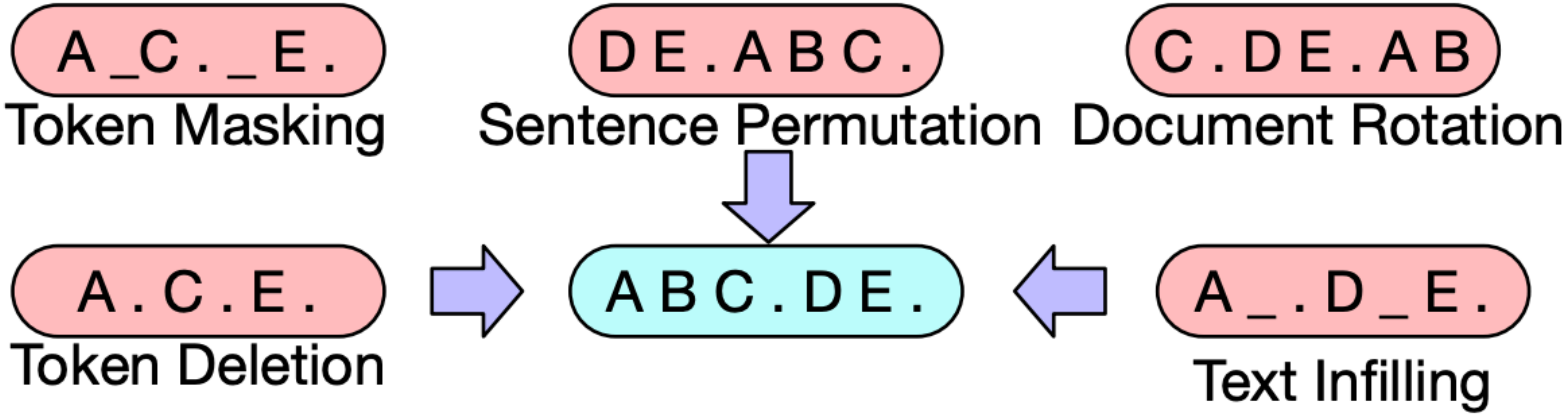


Lewis et al. (2019)



BART Pretraining

- We're not reconstructing the input the same way as BERT, so can we corrupt the input in different ways?
- Many corruption strategies can be used on the encoder side



Which one should be used?

Lewis et al. (2019)

Can do all the same tasks

- BART can also do all the tasks that BERT does!
- **Classification:**
  - Give input to both encoder AND decoder (input the sequence twice)
  - Append [CLS] token to decoder sequence and classify its output
- **Sequence Labeling:**
  - Give input to both encoder AND decoder (input the sequence twice)
  - Classify decoder output representations for each token

Can do all the same 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

Almost as good as RoBERTa

Way better than BERT! Why ?

Results: Summarization

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.
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.

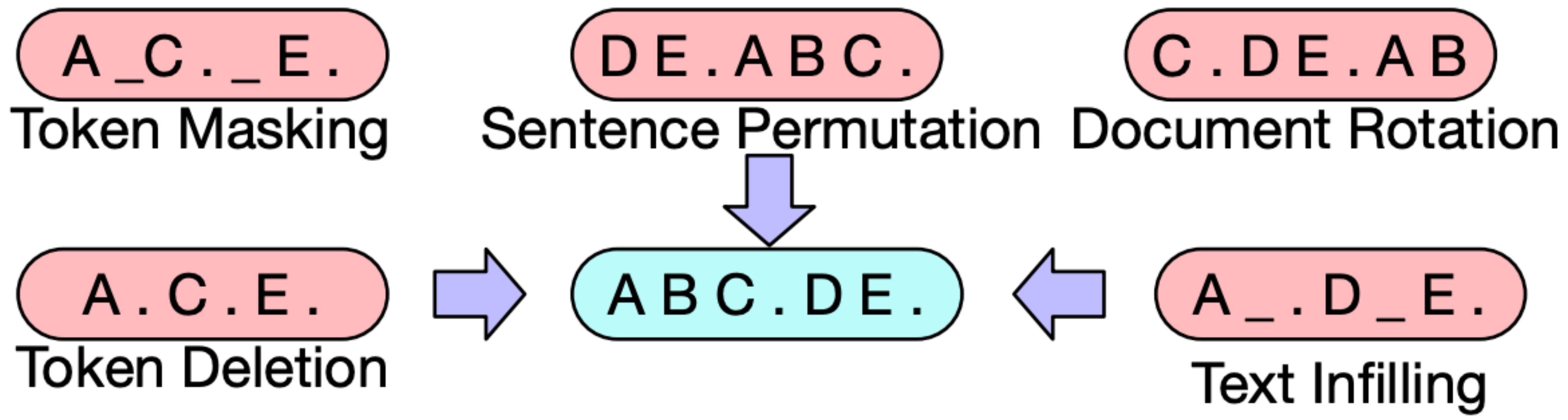
However, BART can do generation tasks too

Decoder is autoregressive!

Lewis et al. (2019)

Which encoder-side corruption?

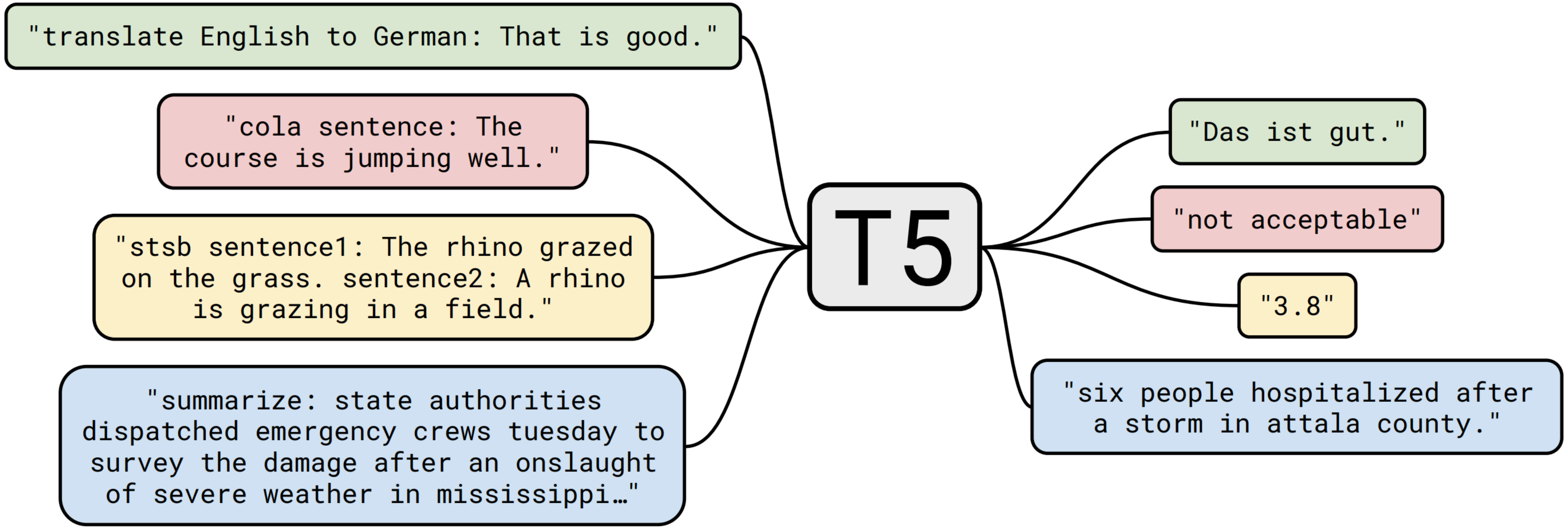
Model	SQuAD 1.1 F1	MNLI Acc	ELI5 PPL	XSum PPL	ConvAI2 PPL	CNN/DM PPL
BART Base						
w/ Token Masking	90.4	84.1	25.05	7.08	11.73	6.10
w/ Token Deletion	90.4	84.1	24.61	6.90	11.46	5.87
w/ Text Infilling	**90.8**	84.0	24.26	**6.61**	**11.05**	5.83
w/ Document Rotation	77.2	75.3	53.69	17.14	19.87	10.59
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**



- Different corruption better for transfer to different tasks
- **Use combination of text infilling + sentence permutation**

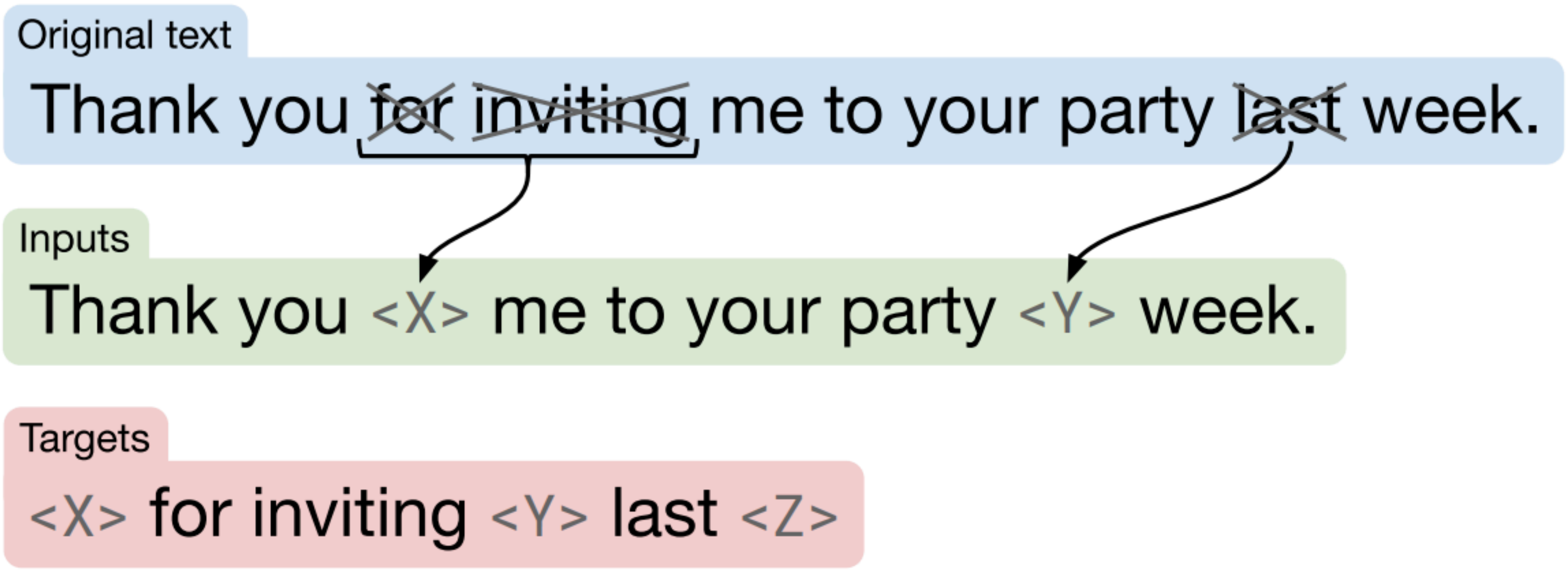
T5

- **Similar idea as BART:** Any problem can be cast as sequence-to-sequence



Raffel et al. (2019)

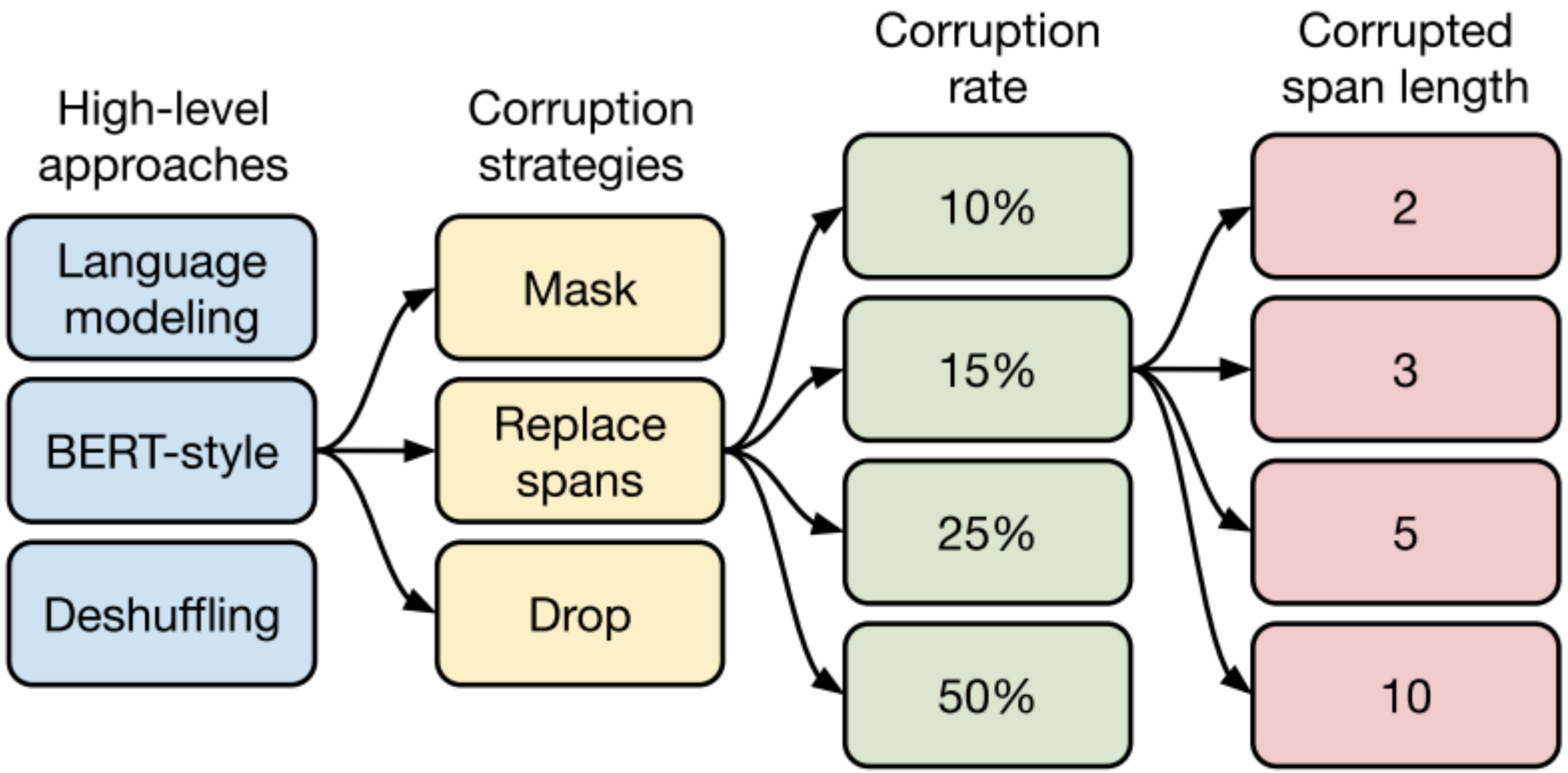
T5 Pretraining



- Similar to BART
- Uses the infilling objective where tokens are reconstructed from underspecified mask corruptions

Raffel et al. (2019)

T5 Pretraining Decisions



- Explored many dimensions of pretraining in se2seq framework
- Took findings to train much larger model — 11B parameters!

Raffel et al. (2019)

Recap

- **Contextual representations:** Let us model words and sequences conditioned on the context around them
- **ELMo:** Based on bidirectional LSTMs. **Good for pretrained embeddings.**
- **GPT**: Uses a transformer decoder. **Good for generating text as a language model.**
- **BERT**: Uses a transformer encoder. **Good for classification and sequence labelling.**
- **BART + T5:** Pretraining sequence-to-sequence transformer models. **Extendable to all task types!**

References

- Peters, M.E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018). Deep Contextualized Word Representations. *North American Chapter of the Association for Computational Linguistics*.
- Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.
- Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., & Zettlemoyer, L. (2019). Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.
- Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., & Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, *21*(1), 5485-5551.