BERT (Encoder-only Architecture)

Pretrained Models 2024 S | MSc CogSys

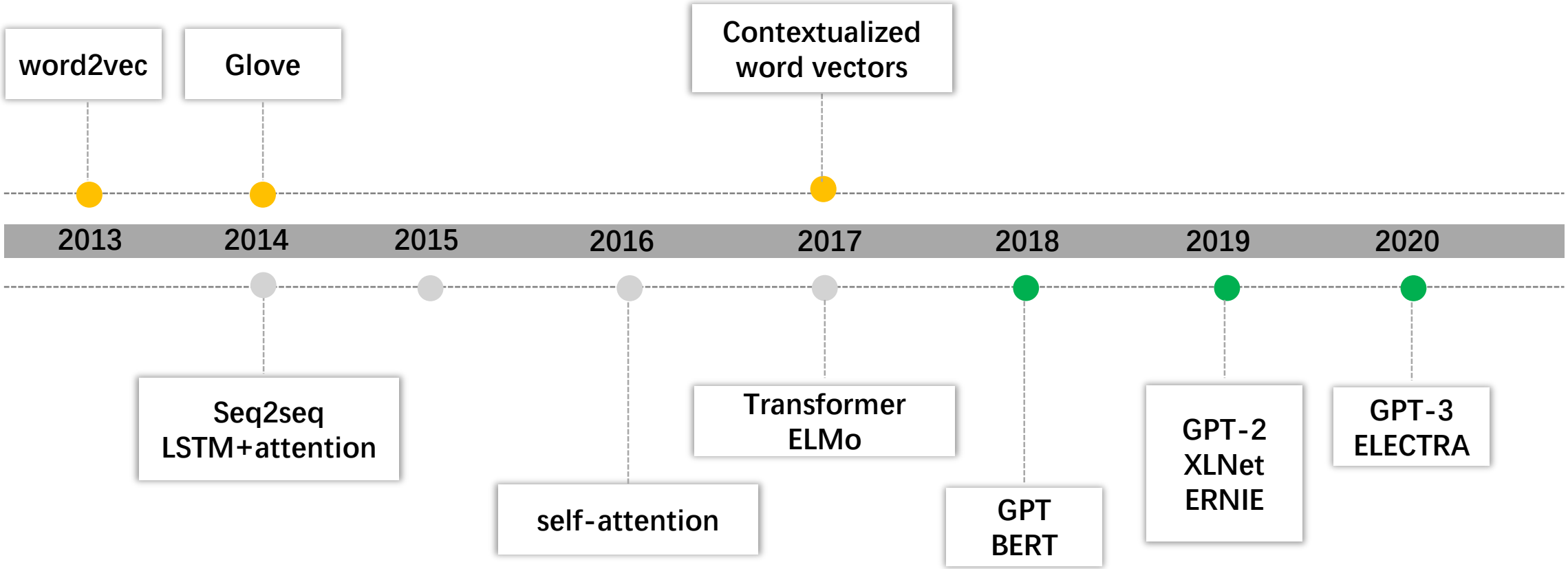
Meng Li

meng.li@uni-potsdam.de

(Adapted from *Jacob Devlin*, *Danqi Chen*)

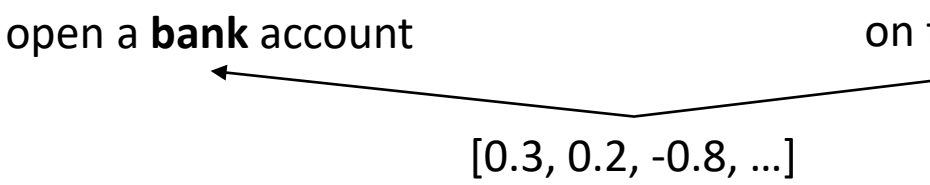
Announcement

- Presenters are updated in syllabus (https://limengnlp.github.io/teaching/pretrain_24s/)
  - Some topics are cancelled;
  - Q: move date forward or leave several breaks?


word2vec
Glove
Contextualized word vectors
2013
2014
2015
2016
2017
2018
2019
2020
Seq2seq LSTM+attention
self-attention
Transformer ELMo
GPT BERT
GPT-2 XLNet ERNIE
GPT-3 ELECTRA

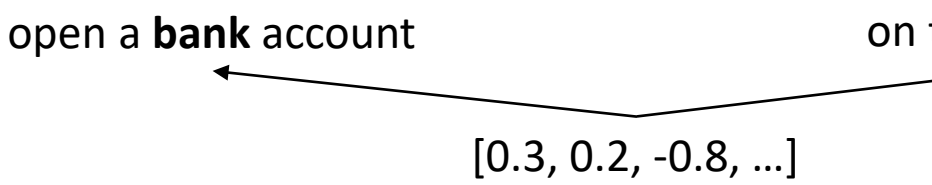
Pre-training in NLP

- Non-contextual word embedding
  - Continuous Bag-of-Words (CBOW) and Skip-Gram (SG) models;
  - Popular implementations: Word2vec and GloVe
  - Limitation: The embedding for a word does is always the same regardless of its context. It fails to model polysemy.



Pre-training in NLP

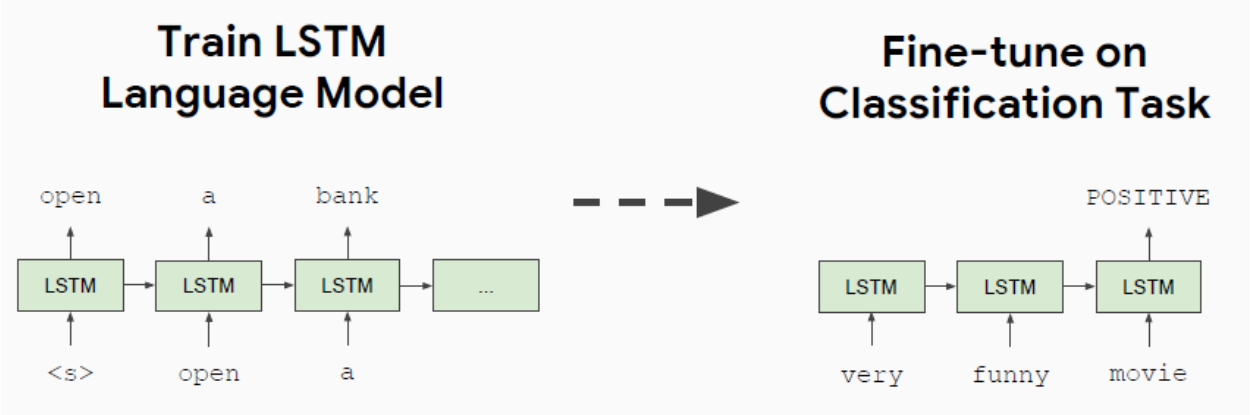
- Non-contextual word embedding
  - Continuous Bag-of-Words (CBOW) and Skip-Gram (SG) models;
  - Popular implementations: Word2vec and GloVe
  - Limitation: The embedding for a word does is always the same regardless of its context. It fails to model polysemy.



  - Solution: Train **contextual** representations on text corpus ?

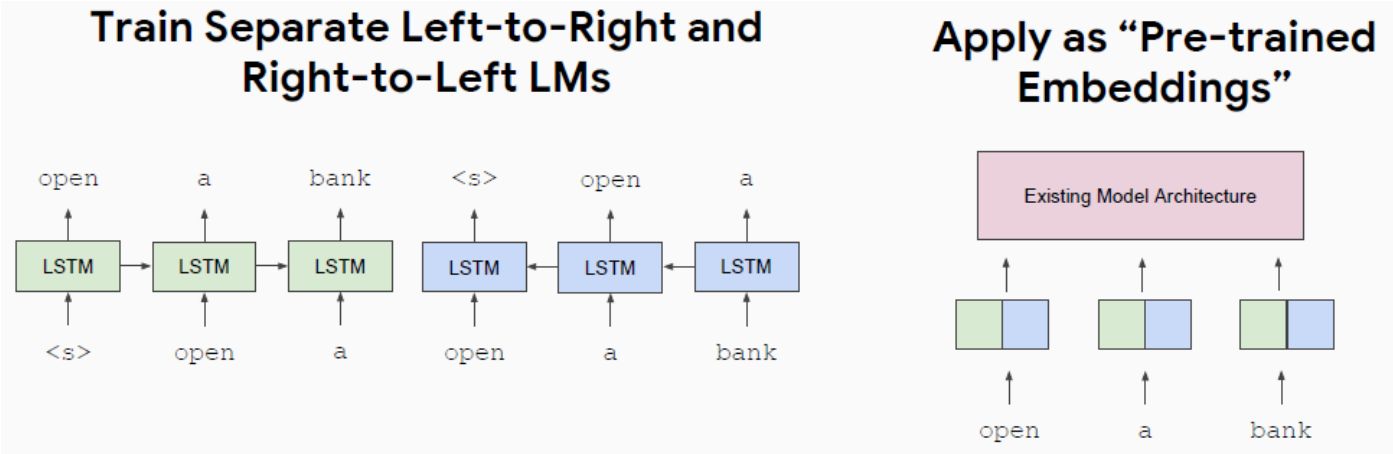
Pre-training in NLP

- Pre-trained contextual representations
  - [Andrew M. Dai, Quoc V. Le 2015]



Pre-training in NLP

- Pre-trained contextual representations
  - ELMo 2017
  - GPT-1 2018



BERT

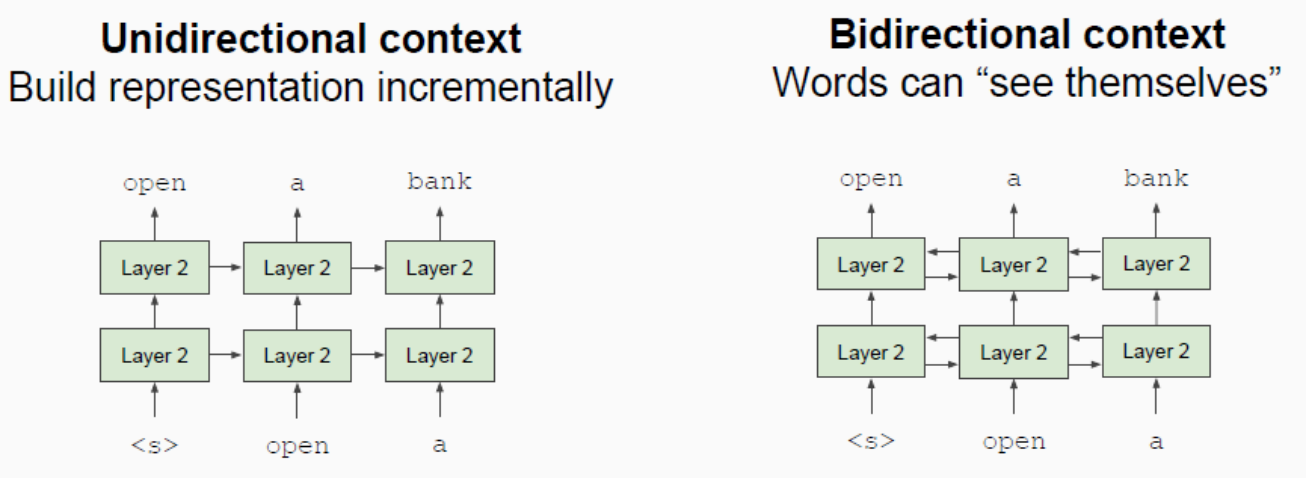
- It is a fine-tuning approach based on a deep **Transformer encoder**
- The key: learn representations based on **bidirectional context**

Why? Because both left and right contexts are important to understand the meaning of words.
Example #1: we went to the river bank.
Example #2: I need to go to bank to make a deposit.

- **Pre-training objectives**: masked language modeling + next sentence prediction
- State-of-the-art performance on a large set of sentence-level and token-level tasks

BERT

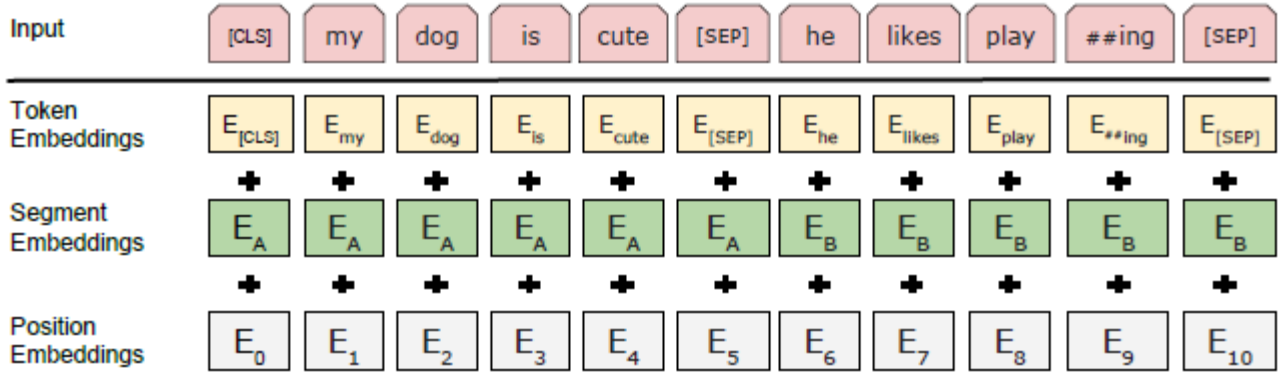
- Motivation: Unidirectionality -> Bidirectionality
  - Why unidirectional LMs? (only use left or right context)
  - Reason 1: Directionality is needed to generate a well-formed probability distribution.
  - Reason 2: Words can “see themselves” in a bidirectional encoder.



  - Q: the author claims “language understanding is bidirectional” in his talk at Stanford. Is it cognitive plausible? How to understand the role of bidirectionality in language model pre-training?

BERT

- Input/Output Representations
  - **WordPiece** embeddings with a 30,000 token vocabulary.
  - [CLS] sent / [CLS] sent A [SEP] sent B
  - a learned embedding to every token indicating whether it belongs to sentence A or sentence B



BERT

- Task #1: Masked LM
  - Mask out k (k=15%) of the input words, and then predict the masked words

store gallon
the man went to the [MASK] to buy a [MASK] of milk

  - 80-10-10 corruption (For the 15% predicted words)
    - 80% of the time, they replace it with [MASK] token
    - 10% of the time, they replace it with a random word in the vocabulary
    - 10% of the time, they keep it unchanged

went to the store ⟶ went to the [MASK]
went to the store ⟶ went to the running
went to the store ⟶ went to the store

    - Why? Because [MASK] tokens are never seen during fine-tuning

BERT

- Task #1: Masked LM

  - Masking rate (too little masking: too expensive to train; too much masking: not enough context)
    - Should you mask 15% in masked language modeling?
    - Masked autoencoders are scalable vision learners (Vision pre-training in MAE: **75%**) different semantic density in vision and language
  - Masking strategy
    - 15% tokens are uniformly sampled
    - Improve: Span masking and PMI masking

BERT

- Task #2: Next Sentence Prediction (NSP)
  - Motivation: downstream tasks such as Question Answering (QA) and Natural Language Inference (NLI) are based on understanding the *relationship* between two sentences.
  - Description: predict whether Sentence B is actual sentence that proceeds Sentence A, or a random sentence.

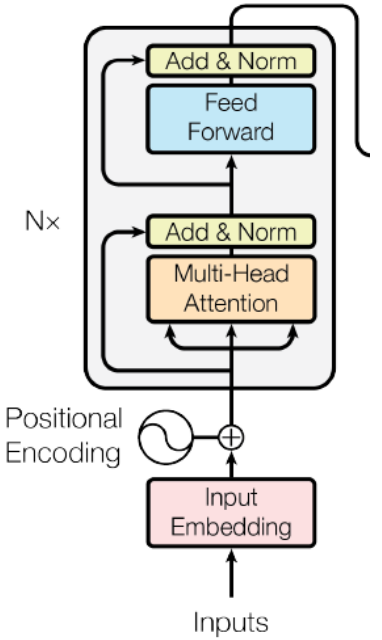
Sentence A = The man went to the store.
Sentence B = He bought a gallon of milk.
Label = IsNextSentence

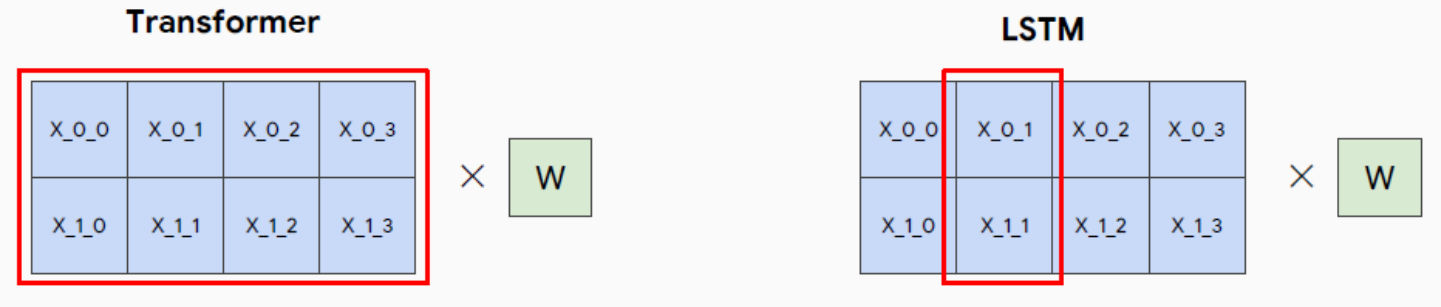
Sentence A = The man went to the store.
Sentence B = Penguins are flightless.
Label = NotNextSentence

  - The final model achieves **97%-98%** accuracy on NSP.

BERT

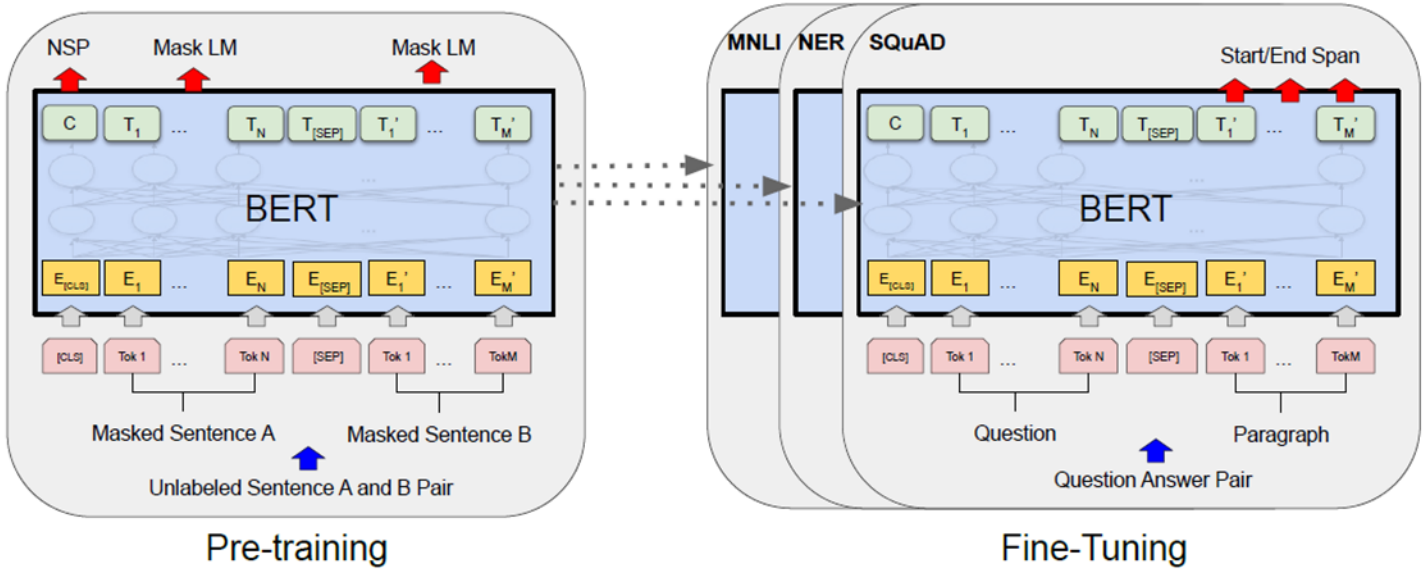
- Model Architecture
  - Transformer encoder;
  - Unified architecture across different tasks;
  - Empirical advantages of Transformer vs. LSTM:
    - Self-attention -> capture long dependency
    - Training efficiency: single multiplication per layer; effective batch size is number of words, not sequences





BERT

- Details
  - Data: Wikipedia (2.5B words) + BookCorpus (800M words)
  - Batch Size: 131,072 words (1024 sequences * 128 length or 256 sequences * 512 length)
  - Training Time: 1M steps (~40 epochs)
  - Optimizer: AdamW, 1e-4 learning rate, linear decay
  - BERT-Base: 12-layer, 768-hidden, 12-head, 110M parameters;
  - BERT-Large: 24-layer, 1024-hidden, 16-head, 340M parameters;
  - Trained on 4x4 or 8x8 TPU slice for 4 days



BERT

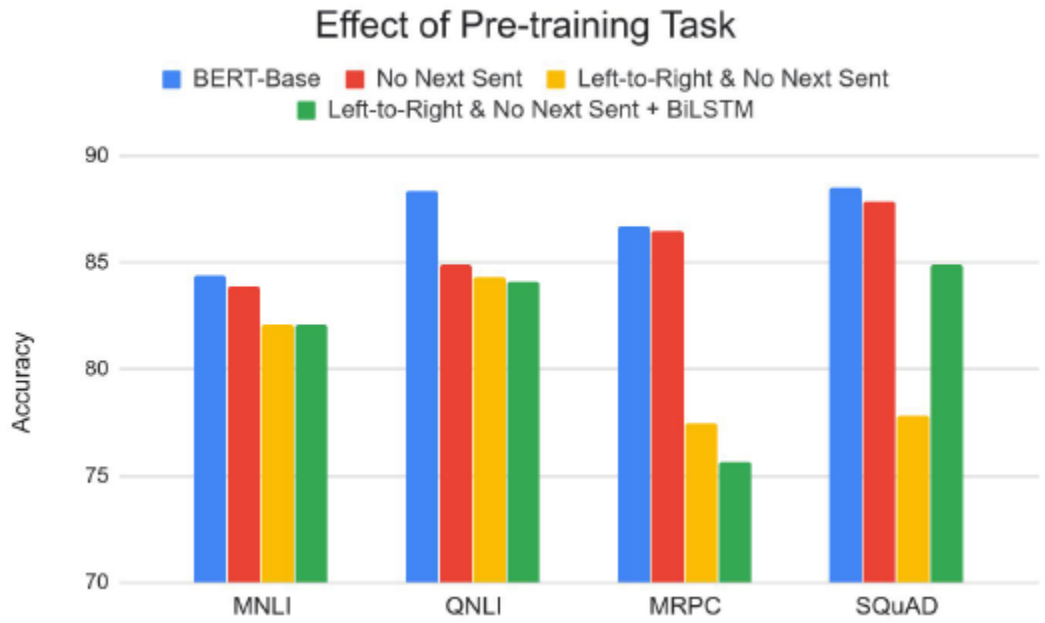
System	MNLI-(m/mm) 392k	QQP 363k	QNLI 108k	SST-2 67k	CoLA 8.5k	STS-B 5.7k	MRPC 3.5k	RTE 2.5k	Average -
Pre-OpenAI SOTA	80.6/80.1	66.1	82.3	93.2	35.0	81.0	86.0	61.7	74.0
BiLSTM+ELMo+Attn	76.4/76.1	64.8	79.8	90.4	36.0	73.3	84.9	56.8	71.0
OpenAI GPT	82.1/81.4	70.3	87.4	91.3	45.4	80.0	82.3	56.0	75.1
$\text{BERT}_{\text{BASE}}$	84.6/83.4	71.2	90.5	93.5	52.1	85.8	88.9	66.4	79.6
$\text{BERT}_{\text{LARGE}}$	**86.7/85.9**	**72.1**	**92.7**	**94.9**	**60.5**	**86.5**	**89.3**	**70.1**	**82.1**

Table 1: GLUE Test results, scored by the evaluation server (https://gluebenchmark.com/leaderboard). The number below each task denotes the number of training examples. The "Average" column is slightly different than the official GLUE score, since we exclude the problematic WNLI set.[8] BERT and OpenAI GPT are single-model, single task. F1 scores are reported for QQP and MRPC, Spearman correlations are reported for STS-B, and accuracy scores are reported for the other tasks. We exclude entries that use BERT as one of their components.

MultiNLI
Premise: Hills and mountains are especially sanctified in Jainism.
Hypothesis: Jainism hates nature.
Label: Contradiction

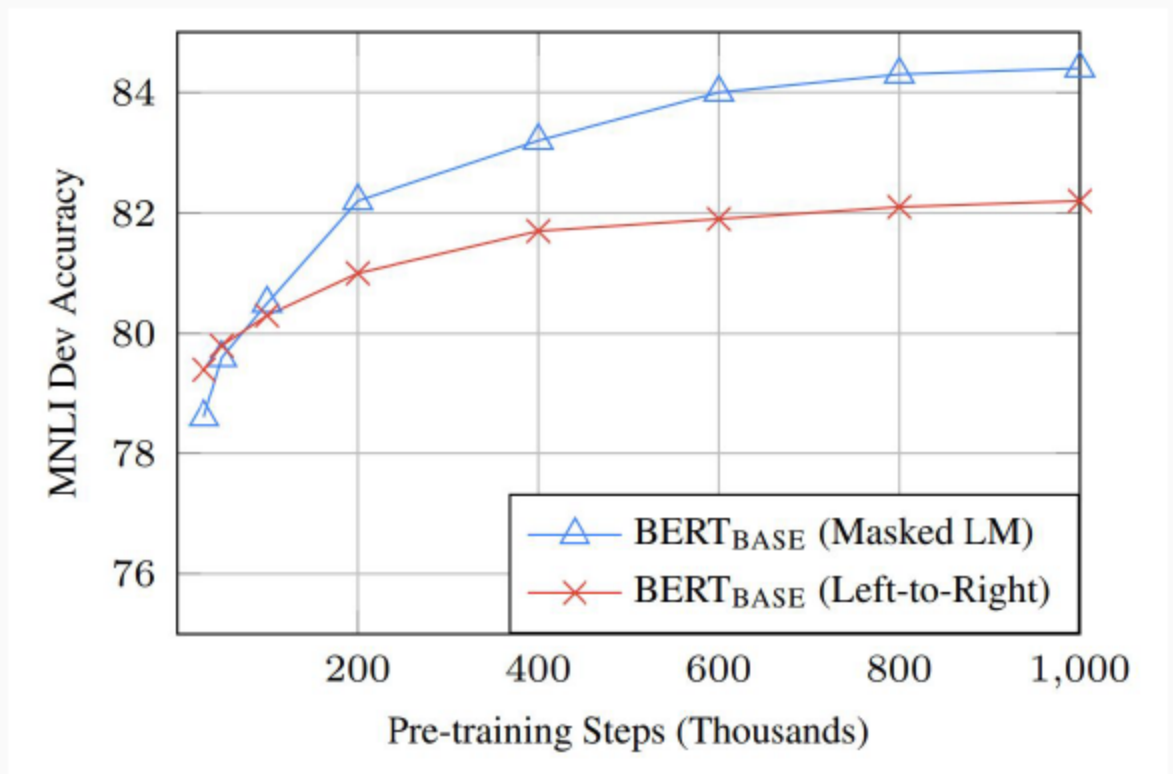
CoLa
Sentence: The wagon rumbled down the road.
Label: Acceptable
Sentence: The car honked down the road.
Label: Unacceptable

BERT



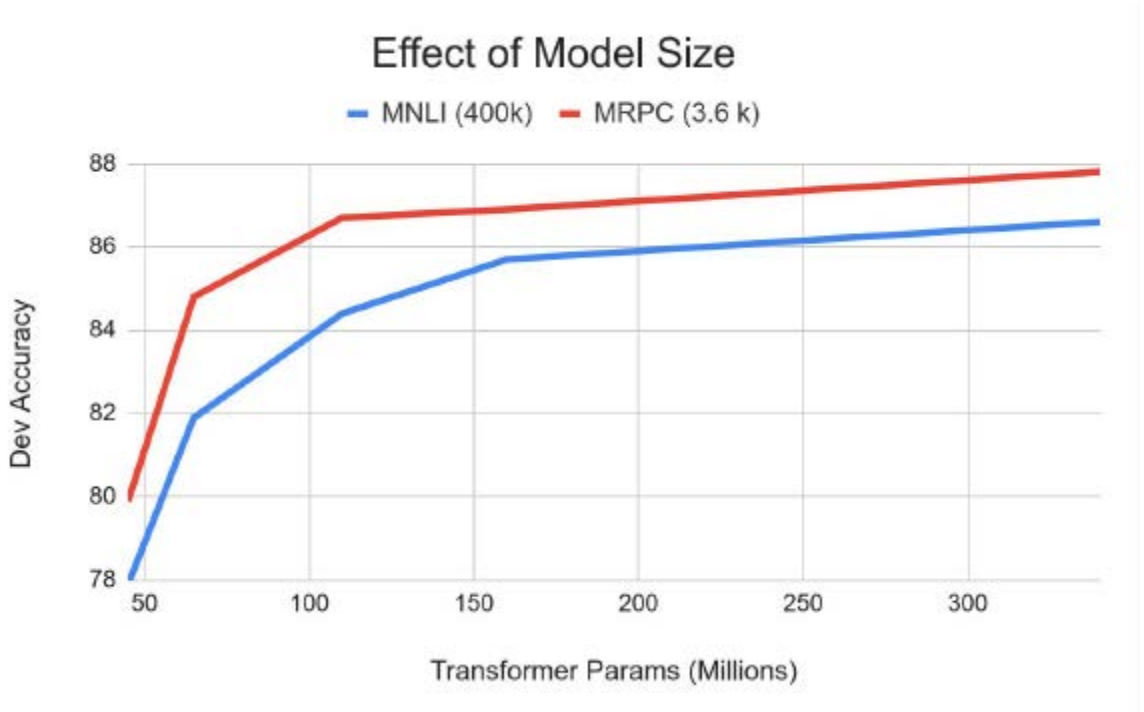
- Effect of Pre-training Task
- Masked LM (compared to left-to-right LM) is very important on some tasks, Next Sentence Prediction is important on other tasks.
- Left-to-right model does very poorly on word-level task (SQuAD), although this is mitigated by BiLSTM

BERT



- Effect of Directionality and Training Time
- Masked LM takes slightly longer to converge because we only predict 15% instead of 100%
- But absolute results are much better almost immediately

BERT



- Effect of Model Size
- Big models help a lot
- Going from 110M -> 340M params helps even on datasets with 3,600 labeled examples
- Improvements have not asymptoted

BERT

Masking Rates			Dev Set Results		
MASK	SAME	RND	MNLI	NER	
			Fine-tune	Fine-tune	Feature-based
80%	10%	10%	84.2	95.4	94.9
100%	0%	0%	84.3	94.9	94.0
80%	0%	20%	84.1	95.2	94.6
80%	20%	0%	84.4	95.2	94.7
0%	20%	80%	83.7	94.8	94.6
0%	0%	100%	83.6	94.9	94.6

Table 8: Ablation over different masking strategies.

- Effect of Masking Strategy
- Masking 100% of the time hurts on feature-based approach
- Using random word 100% of time hurts slightly

BERT

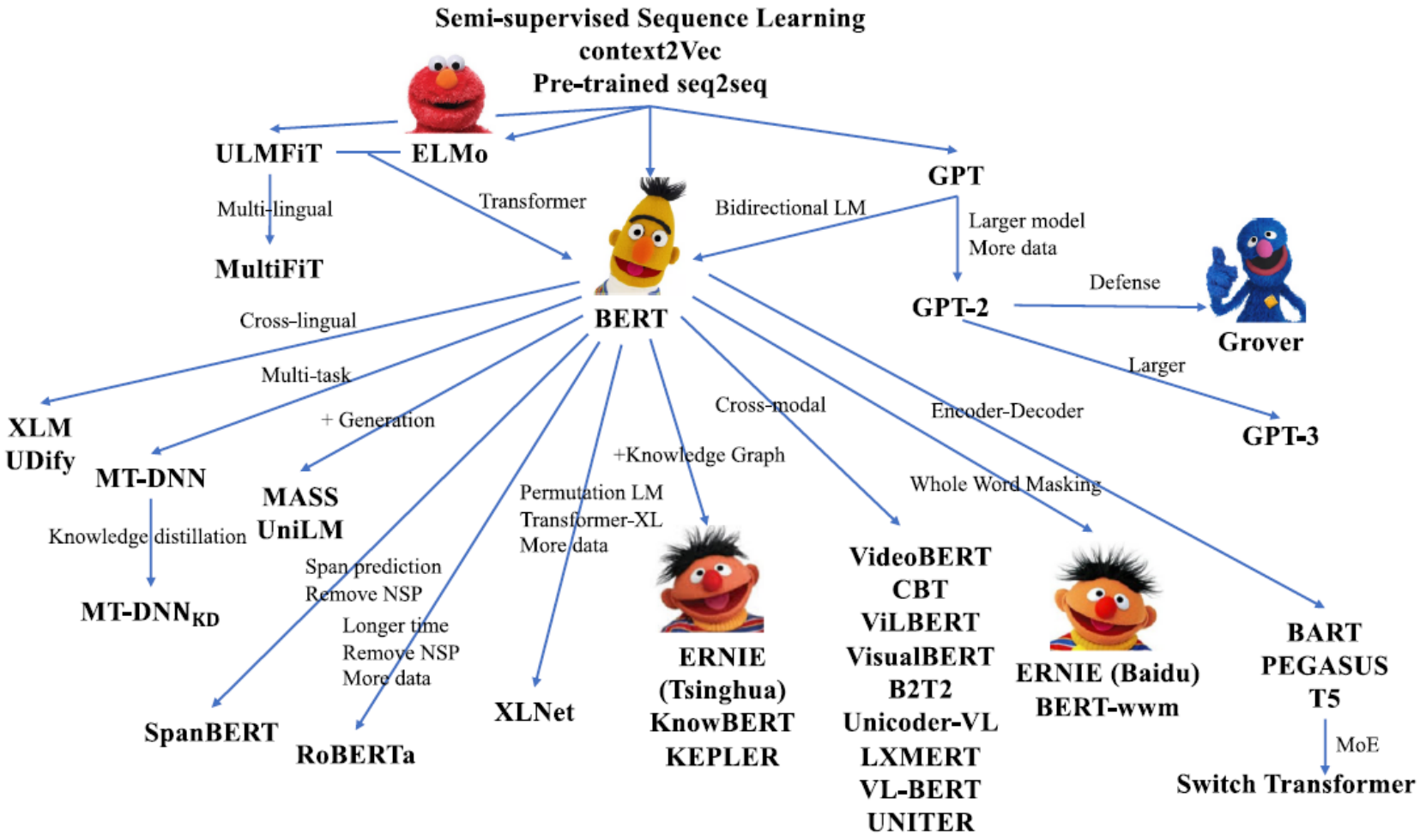
- What can BERT NOT do ?
- BERT cannot generate text (at least not in an obvious way)
- Intended to be used for “analysis” tasks
- Can fill in MASK tokens, but can’t generate left-to-right (repeated put MASK at the end, but it is slow)

BERT

- Empirical results from BERT are great, but biggest impact on the field is:
- With pre-training, bigger == better, without clear limits (so far).
- Unclear if adding things on top of BERT really helps by very much.
  - Good for people and companies building NLP systems.
  - Not necessary a “good thing” for researchers, but important.

After BERT

- How to build better PTMs
  - Masking;
  - Multi-lingual
  - Multi-modal
  - …
- How these model works
  - World knowledge
  - Linguistic knowledge
- How to efficiently use them



source

RoBERTa

- a replication study of BERT pre-training
- Trained on 10x data & longer
- remove NSP objective
- dynamically changing the masking pattern applied to the training data
- Much stronger performance than BERT (e.g., 94.6 vs 90.9 on SQuAD)
- Citation 10000+, rejected by ICLR

SpanBERT

- masking contiguous random spans, rather than random tokens
- training the span boundary representations to predict the entire content of the masked span, without relying on the individual token representations within it.
- substantial gains on span selection tasks such as question answering and co-reference resolution.

$$\mathcal{L}(\text{football}) = \mathcal{L}_{\text{MLM}}(\text{football}) + \mathcal{L}_{\text{SBO}}(\text{football})$$
$$= -\log P(\text{football} \mid \mathbf{x}_7) - \log P(\text{football} \mid \mathbf{x}_4, \mathbf{x}_9, \mathbf{p}_3)$$

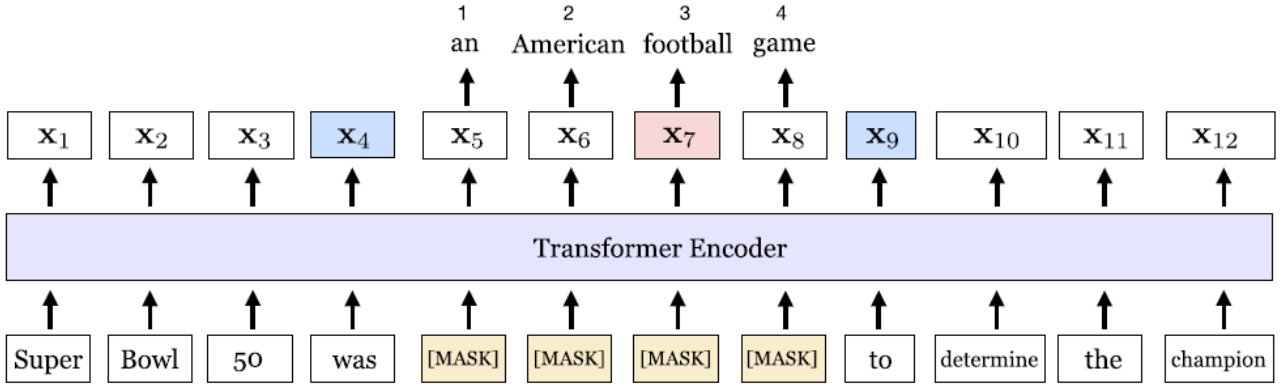


Figure 1: An illustration of SpanBERT training. The span *an American football game* is masked. The SBO uses the output representations of the boundary tokens, $\mathbf{x}_4$ and $\mathbf{x}_9$ (in blue), to predict each token in the masked span. The equation shows the MLM and SBO loss terms for predicting the token, *football* (in pink), which as marked by the position embedding $\mathbf{p}_3$, is the *third* token from x_4.

ALBERT

- Interesting observations on reducing parameters:
  - it factorizes the input word embedding matrix into two smaller ones.
  - it enforces parameter-sharing between all Transformer layers to significantly reduce parameters.
  - it proposes the sentence order prediction (SOP) task to substitute BERT's NSP task.
- ALBERT has a slower fine-tuning and inference speed.

XLNet

- unify GPT-style unidirectional generation and BERT- style bidirectional understanding;
- XLNet maximizes the expected log likelihood of a sequence w.r.t. all possible permutations of the factorization order. In expectation, each position learns to utilize contextual information from all positions, i.e., capturing bidirectional context.
- XLNet does not rely on data corruption. -> XLNet does not suffer from the pretrain-finetune discrepancy in BERT. Meanwhile, the autoregressive objective also provides a natural way to use the product rule for factorizing the joint probability of the predicted tokens.

Scientific debt

- Conducting rigorous experiments and extensive ablation studies
- Reward and encourage a line of work that focuses on understanding (not just those that chase a new state-of-the-art), even when they are imperfect
- Establish standard, publicly available pre-training corpora at multiple data scales

Q&A