BERT and GPT

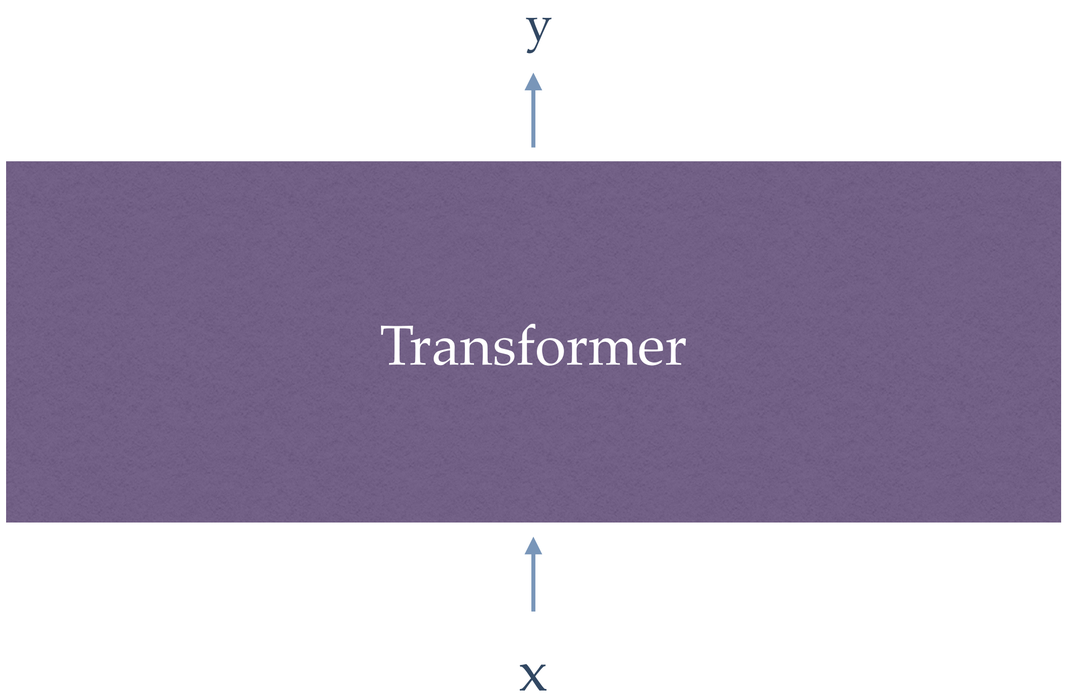
Sandeep Soni

10/10/2024

Story so far

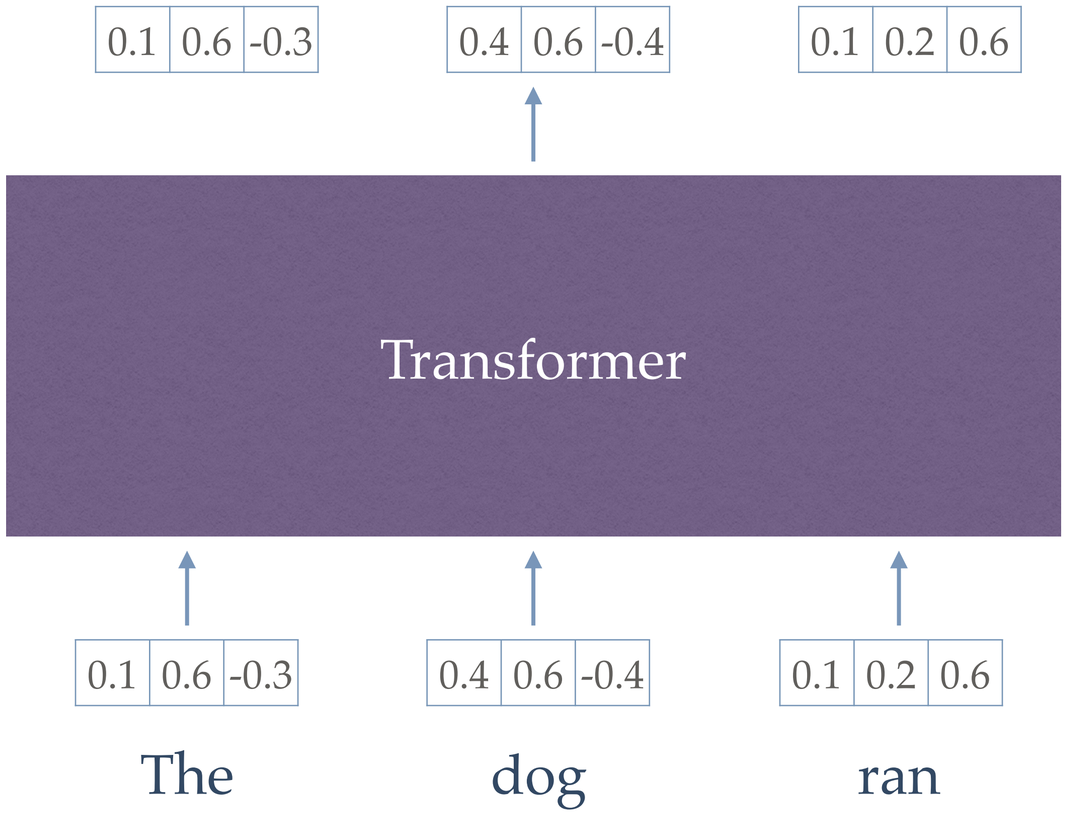
- Language model estimates P(x) for a sequence x
- Count and normalize language models (e.g., N-gram LMs) assume fixed-length conditioning context
- RNN LM can condition on arbitrarily long context but has a sequential bottleneck

Transformers



- Transform an input sequence of vectors into an output sequence of vectors

Transformers



- Transform an input sequence of vectors into an output sequence of vectors

Key ideas

- Every token in a sequence gets an embedding that depends on the embeddings of tokens in the rest of the sequence.
- Multiple transformations of vector sequences.
- Parallel computation

Few more things about transformers

- Cross attention
- Beyond text

Image captioning

Image captioning

A little girl in a pink shirt is swinging.

Image captioning

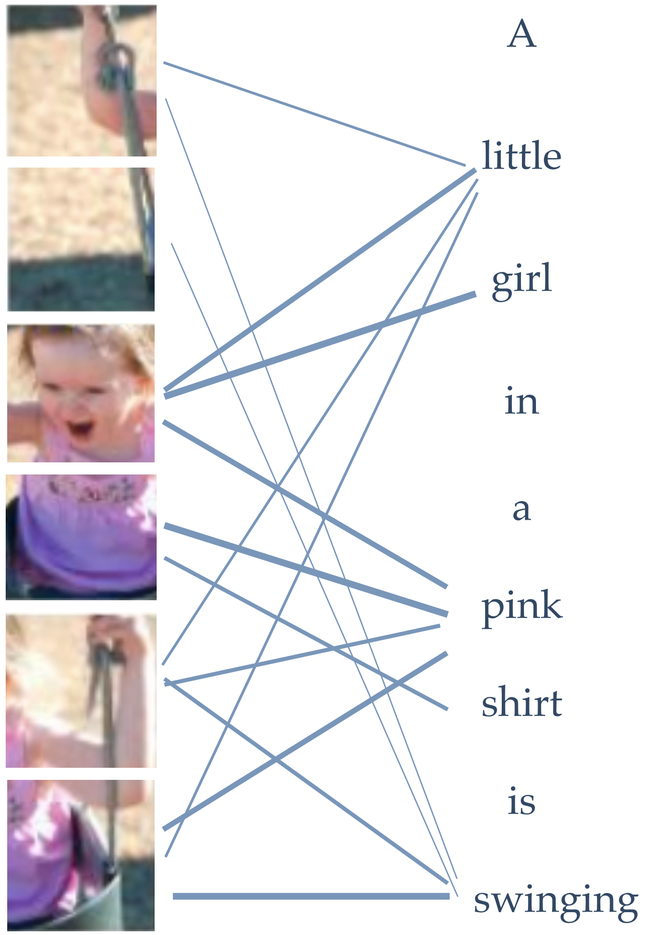
A little girl in a pink shirt is swinging.

A little girl in a pink shirt is swinging.

Transformer

Cross attention

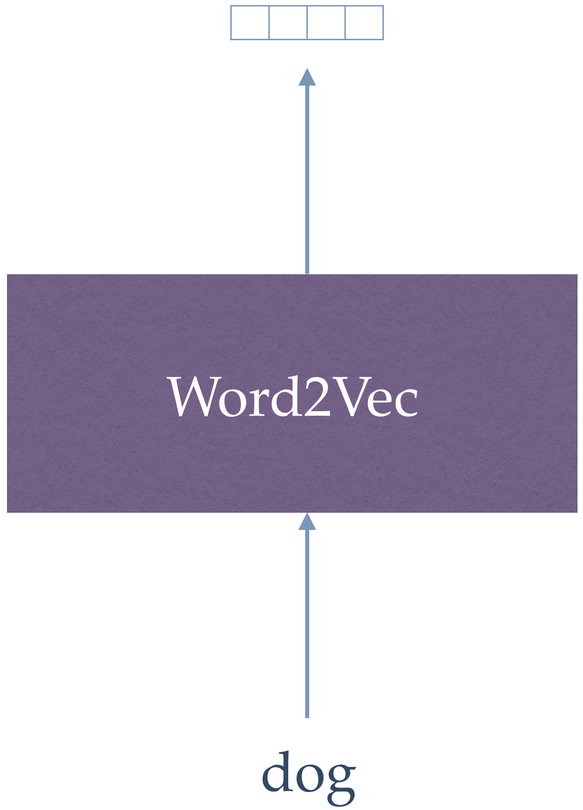
Cross attention



Few more things about transformers

- Cross attention
- Beyond text

Word2vec abstraction



Input: word

Output:embedding

Corpus

N1	The players won the game
N2	The coach praised the players
N3	The players were victorious
…	…
Nn	The players won

Corpus

N1	The players won the game
N2	The coach praised the players
N3	The players were victorious
…	…
Nn	The players won

Data in the form of (tagged) sequences

How to use Word2Vec to measure the similarity between songs without knowing their lyrics?

How to use Word2Vec to measure the similarity between products without knowing their description?

How to use Word2Vec to measure the similarity between things without knowing their stuff?

I stay out too late
Got nothing in my brain
That's what people say, mm-mm
That's what people say, mm-mm
...

Tryna rain, tryna rain on the thunder
Tell the storm I'm new
I'm a wall, come and march on the regular
Painting white flags blue
...

I stay out too late
Got nothing in my brain
That's what people say, mm-mm
That's what people say, mm-mm
...

ts_song1

Tryna rain, tryna rain on the thunder
Tell the storm I'm new
I'm a wall, come and march on the regular
Painting white flags blue
...

b_song1

Corpus

playlist1	ts_song1 b_song4 e_song3 …
playlist2	ts_song5 b_song3 j_song1 …
playlist3	sd_song1 ts_song1 e_song5 …
…	…
playlistn	ts_song1 b_song1 …

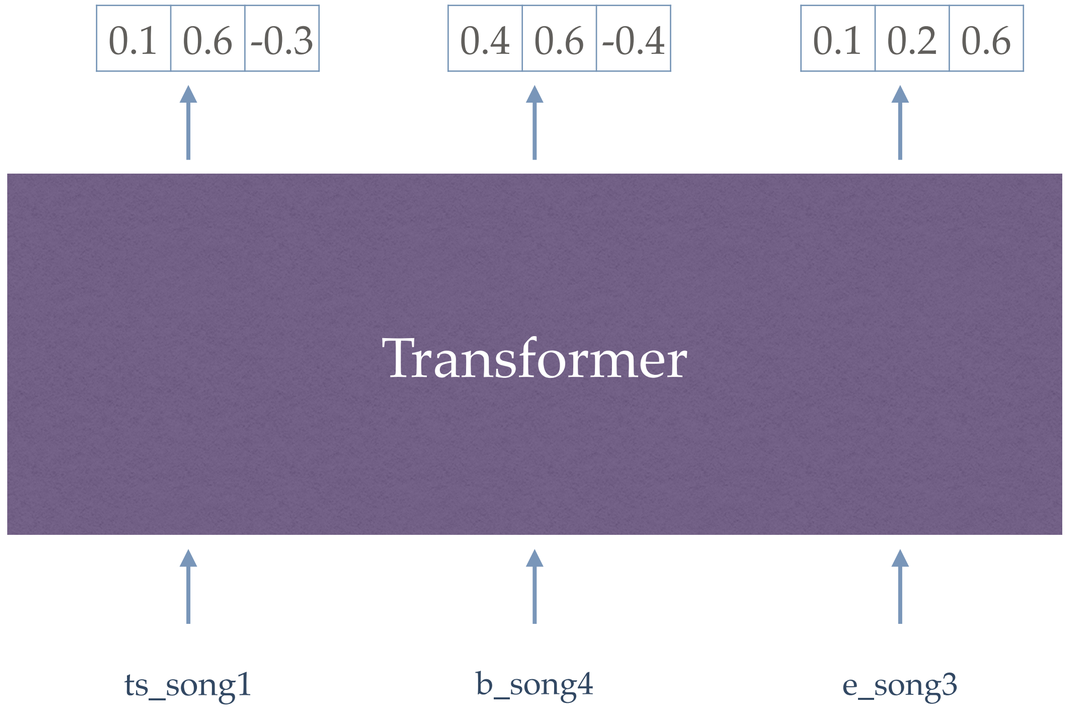
Corpus

playlist1	ts_song1 b_song4 e_song3 …
playlist2	ts_song5 b_song3 j_song1 …
playlist3	sd_song1 ts_song1 e_song5 …
…	…
playlistn	ts_song1 b_song1 …

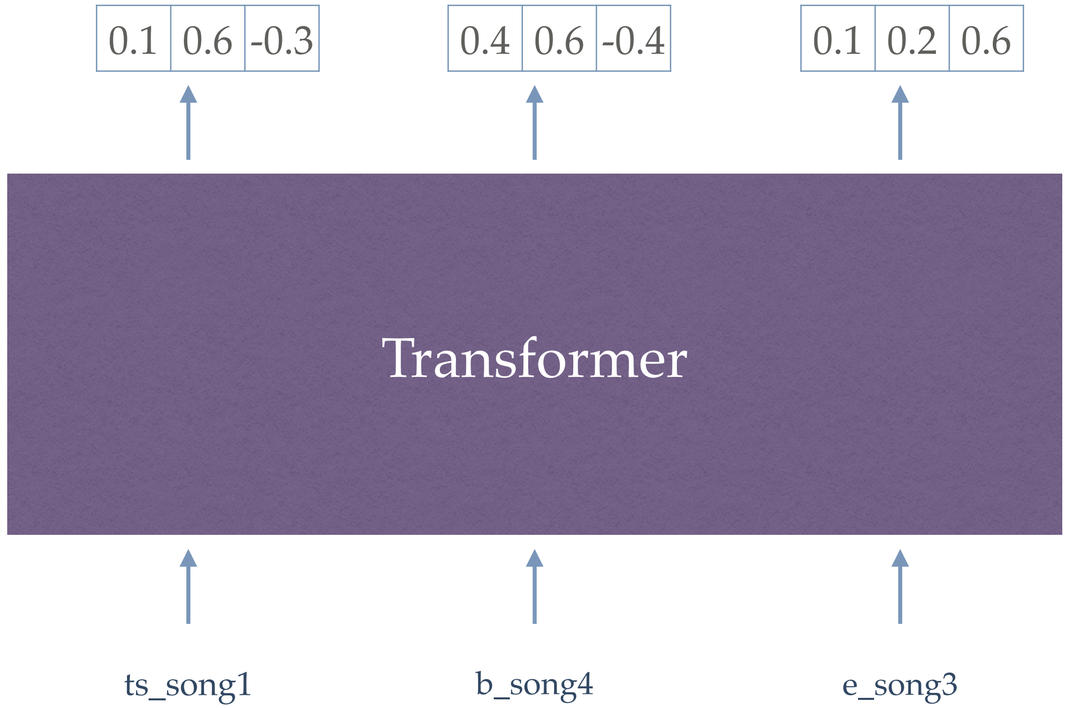
How to measure the similarity between b_song4 and ts_song1?

How to use Transformers to measure the similarity between songs without knowing their lyrics?

Transformers



Transformers



How to measure the similarity between b_song4 and ts_song1 in/across any playlist(s)?

Language modeling

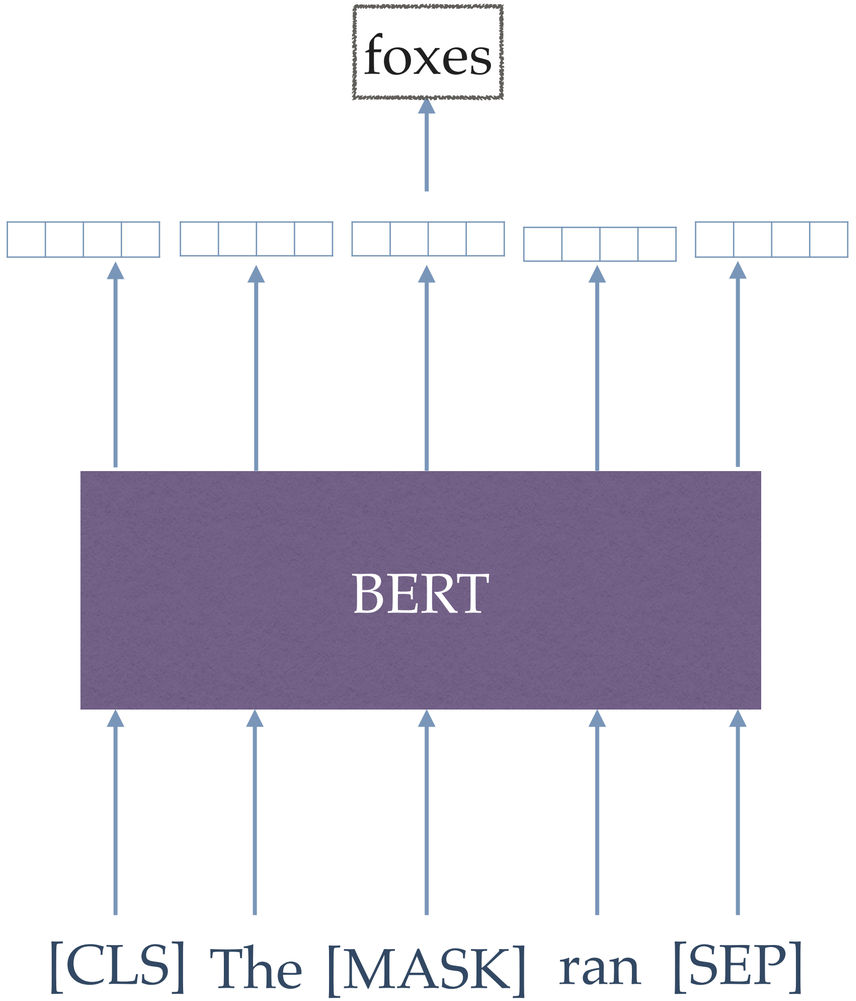
- Instead of modeling P(x), model P(w|c).
- Language modeling as a self-supervised learning task
- This is just one way to set up a language model

Masked Language modeling

Fill in the blank by using the surrounding context

The _____ ran

$$P(w_t \mid \neg w_t)$$



BERT is a language model trained to predict the missing word

BERT

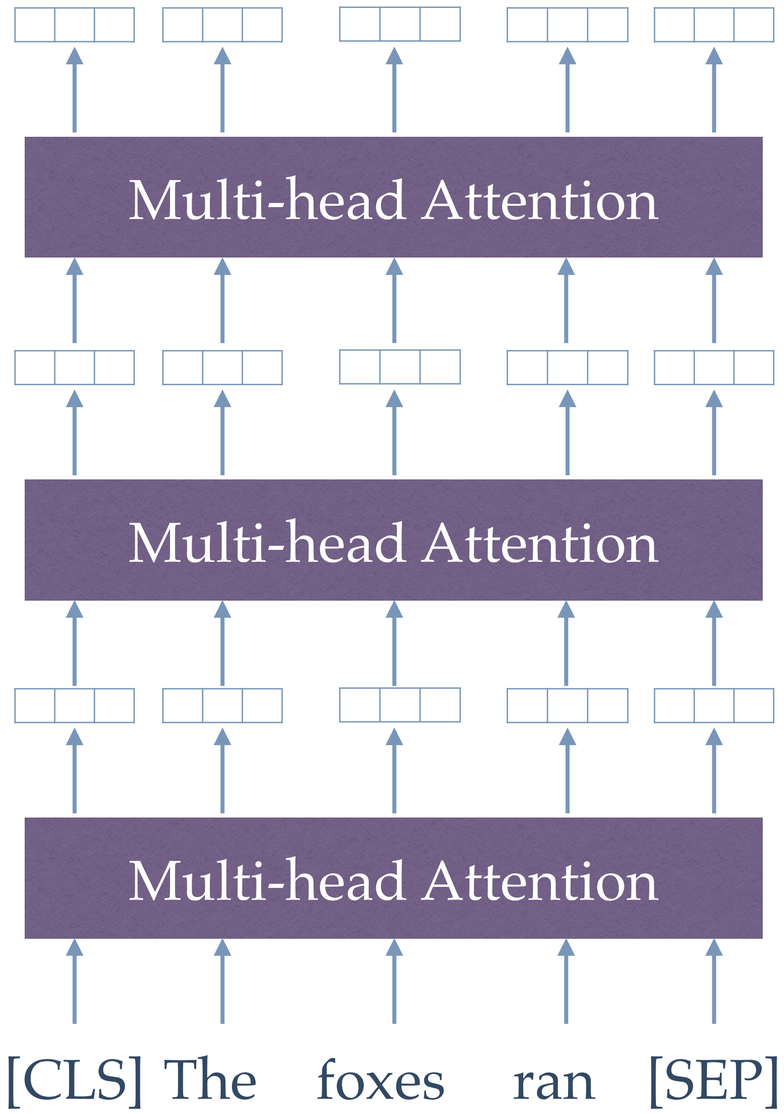
- Masked LM

$$P(x) = \prod_{i=1}^{n} P(x_i \mid x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n)$$

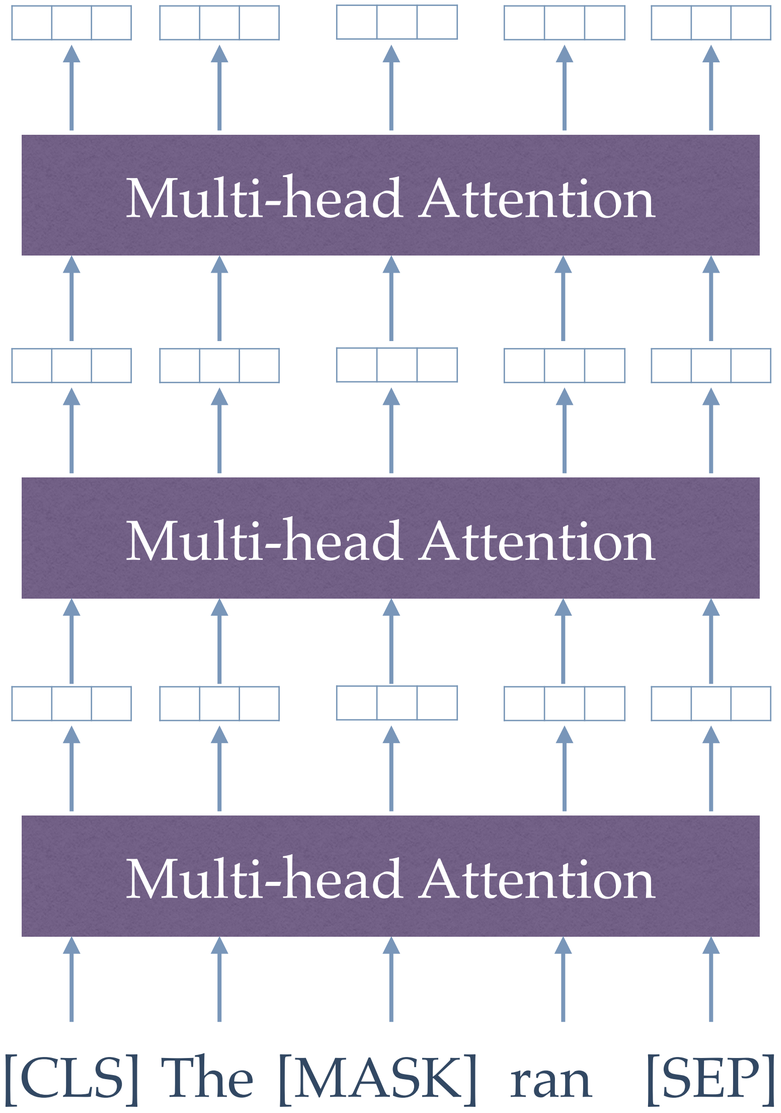
- Pretrained on large amounts of English text such as Wikipedia (2.5B words) and BooksCorpus (800M words)

Special Tokens

- Every sentence is appended with a special token [CLS] in the beginning and [SEP] at the end
- At the time of training, a special token [MASK] replaces randomly picked words and the model parameters are updated till model learns to predict the masked word
- Embeddings for [CLS] and [SEP] tokens are also learned and can be used as vector representations of the entire sequence at inference



Many such transformer blocks staked together make the BERT model



At the time of training, we try to predict the original token in place of MASK token

Training setup

Training setup

Fill in the blank by using the surrounding context

Training setup

Fill in the blank by using the surrounding context

[CLS] The [MASK] ran [SEP]

Training setup

Fill in the blank by using the surrounding context

Predict SEP if two sentences are in sequence

[CLS] The [MASK] ran [SEP]

Training setup

Fill in the blank by using the surrounding context

[CLS] The [MASK] ran [SEP]

Predict SEP if two sentences are in sequence

[CLS] The foxes ran ? The humans were relieved [SEP]

Wordpieces

- Tokens are called wordpieces which allows to limit the vocabulary size and share subword information

this	this
grow	grow
growing	grow + #ing

BERT

- Deep networks (12 layers for BERT base, 24 for BERT large)
- Token representations are high dimensional (768 dims for BERT base, 1024 for BERT large)

Encoder-decoder language models

The foxes ran in the forest after seeing the humans

Encoder-decoder language models

Encoder-decoder language models

Encode a sequence by
hiding some parts …

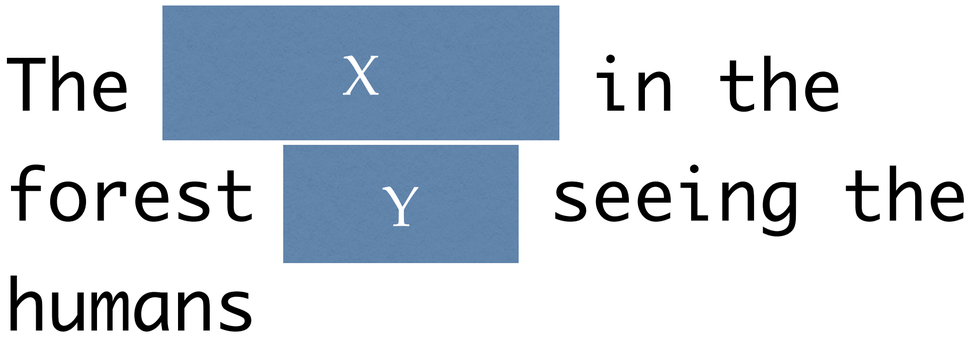
Encoder-decoder language models

Encode a sequence by hiding some parts …

The ~~foxes ran~~ in the forest ~~after~~ seeing the humans

Encoder-decoder language models

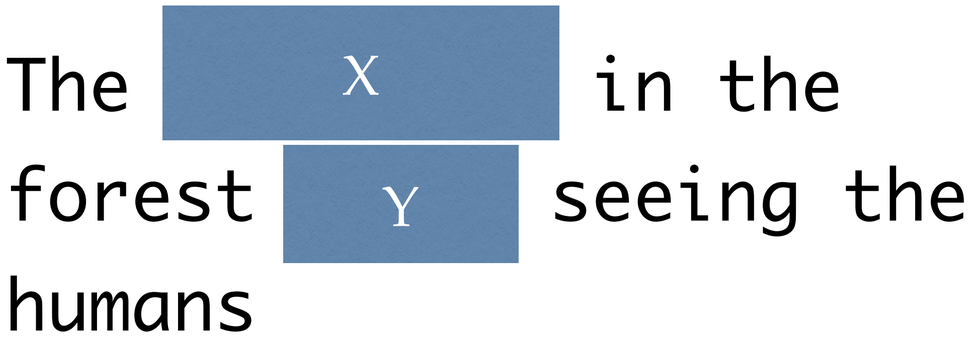
Encode a sequence by hiding some parts …



Encoder-decoder language models

Encode a sequence by hiding some parts …

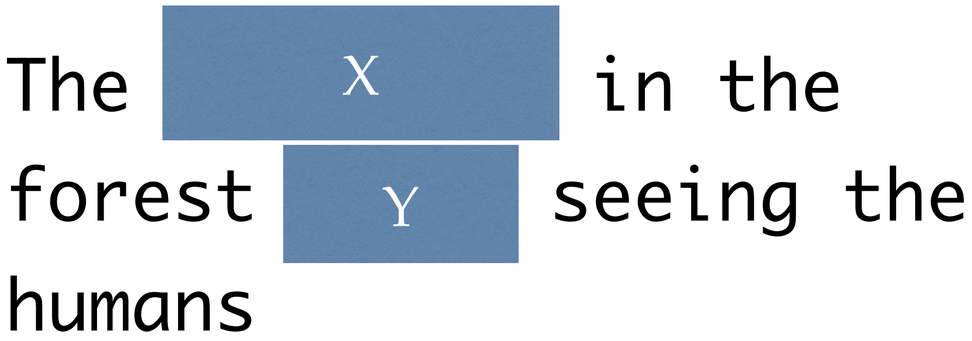
… then decode the missing parts.



Encoder-decoder language models

Encode a sequence by hiding some parts …

… then decode the missing parts.

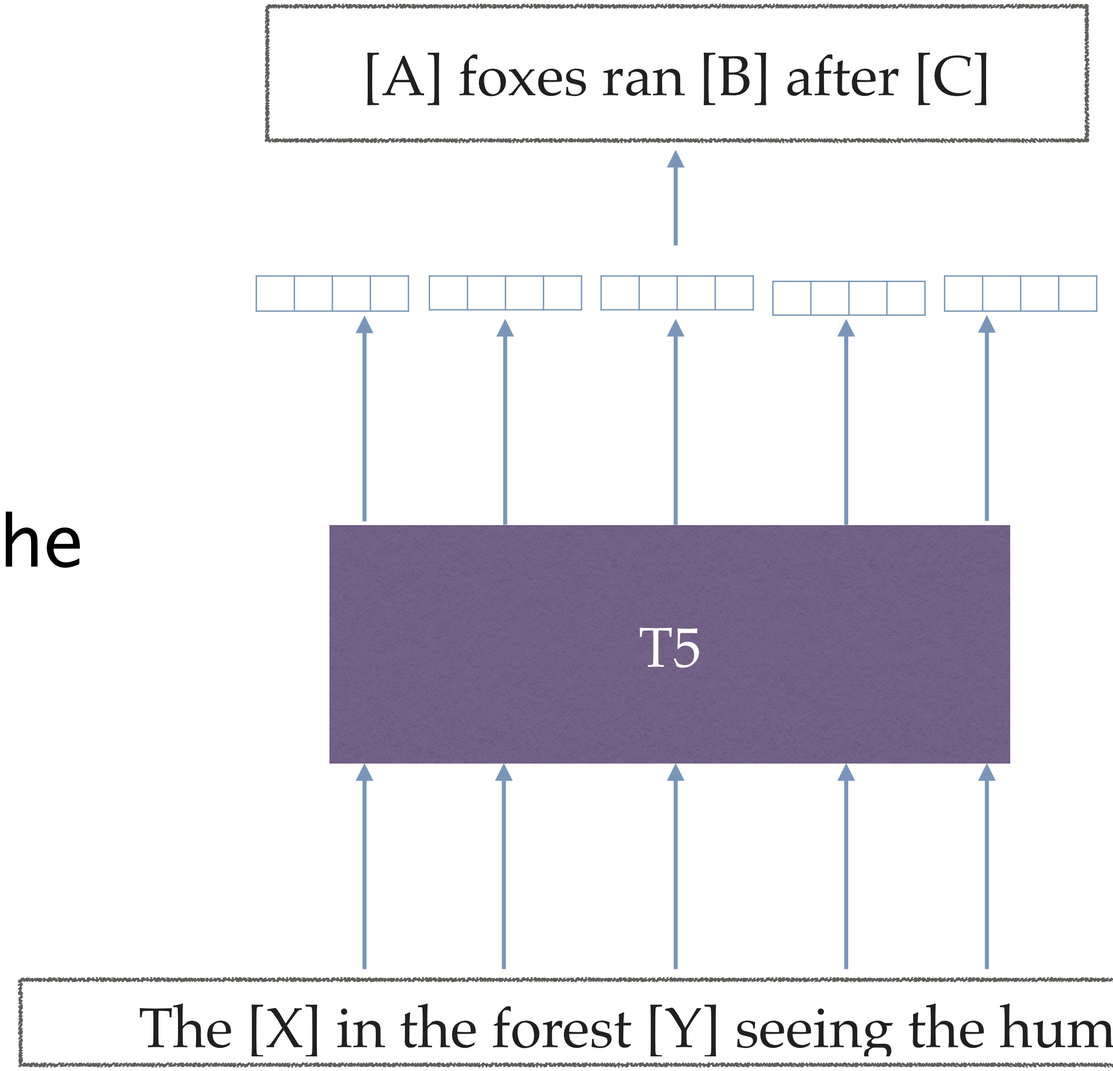


[A] foxes ran [B] after [C]

Encoder-decoder language models

Encode a sequence and try to decode it back

The ~~foxes ran~~ in the forest ~~after~~ seeing the humans



T5 is a language model trained to decode missing spans

T5

- Encoder-Decoder LM: $P(y) = \prod_{i=1}^{n} P(y_i \mid y_1, \ldots, y_{i-1}, x)$
- Pretrained on 750GB of web text

GPT

- Causal LM: $$P(x) = \prod_{i=1}^{n} P(x_i \mid x_1, x_2, \ldots, x_{i-1})$$
- Also called as left-to-right or autoregressive language modeling
- Fill in the blank but always predict the next word in the sequence (e.g., The foxes ___)
- GPT-3 has 175B parameters and is trained on 570GB of web text, books, wikipedia.

Language modeling

Masked LM:

$$P(x) = \prod_{i=1}^{n} P(x_i \mid x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n)$$

Encoder-Decoder LM:

$$P(y) = \prod_{i=1}^{n} P(y_i \mid y_1, \ldots, y_{i-1}, x)$$

Causal LM:

$$P(x) = \prod_{i=1}^{n} P(x_i \mid x_1, x_2, \ldots, x_{i-1})$$

How can we use these new language models?

Task performance

Plugging in the contextual embeddings from these language models can improve performance on linguistic tasks

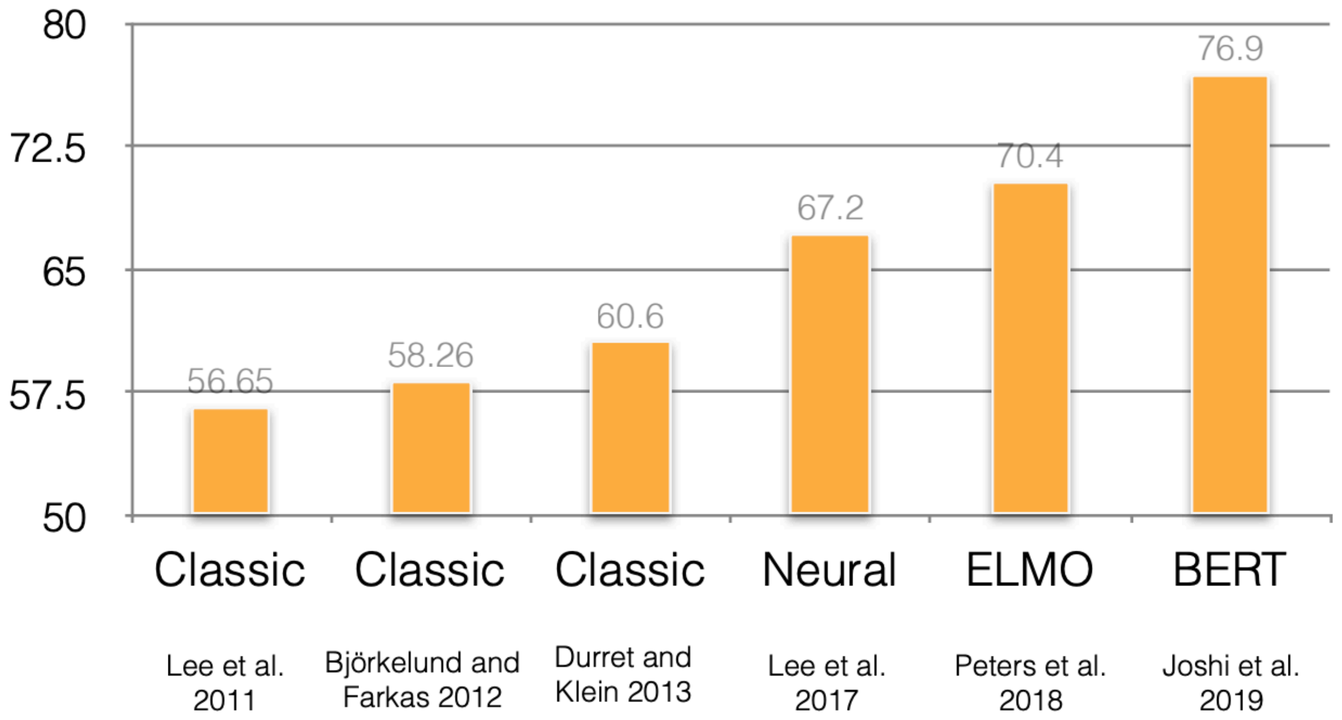


Figure taken from David Bamman's slide

Finetune

- Take a pretrained language model and then fine-tune it on your task and data
- Pretrain + finetune is much better than learning individualized models
- Also allows to do domain adaptation

In class

- bert_exploration