

Transformers Variants

Course Instructor: Chandresh
AI Lab Coordinator @IIT Indore

Course Content

Module I: History of Deep Learning, Sigmoid Neurons, Perceptrons, and learning algorithms. Multilayer Perceptrons (MLPs), Representation Power of MLPs.,

Module II: Feedforward Neural Networks. Backpropagation. first and second-order training methods. NN Training tricks, Regularization

Module III: Introduction to Autoencoders and their characteristics, relation to PCA, Regularization in autoencoders, and Types of autoencoders, Variational Autoencoder

Module IV: Architecture of Convolutional Neural Networks (CNN), types of CNNs. Image classification, Pre-training vs fine-tuning.- representation learning, Object Detection and Semantic Segmentation

Module V: Architecture of Recurrent Neural Networks (RNN), Word Embeddings, Encoder-Decoder Models, Attention Mechanism. Advanced Topics: Transformers, **BERT, GPT and variants**. Nodule VI: Gen AI- Deep generative models: VAE, GAN,

Acknowledgement

- Russ Salakhutdinov and Hugo Larochelle's class on Neural Networks
- Neural Networks and Deep Learning course by Danna Gurari University of Colorado Boulder

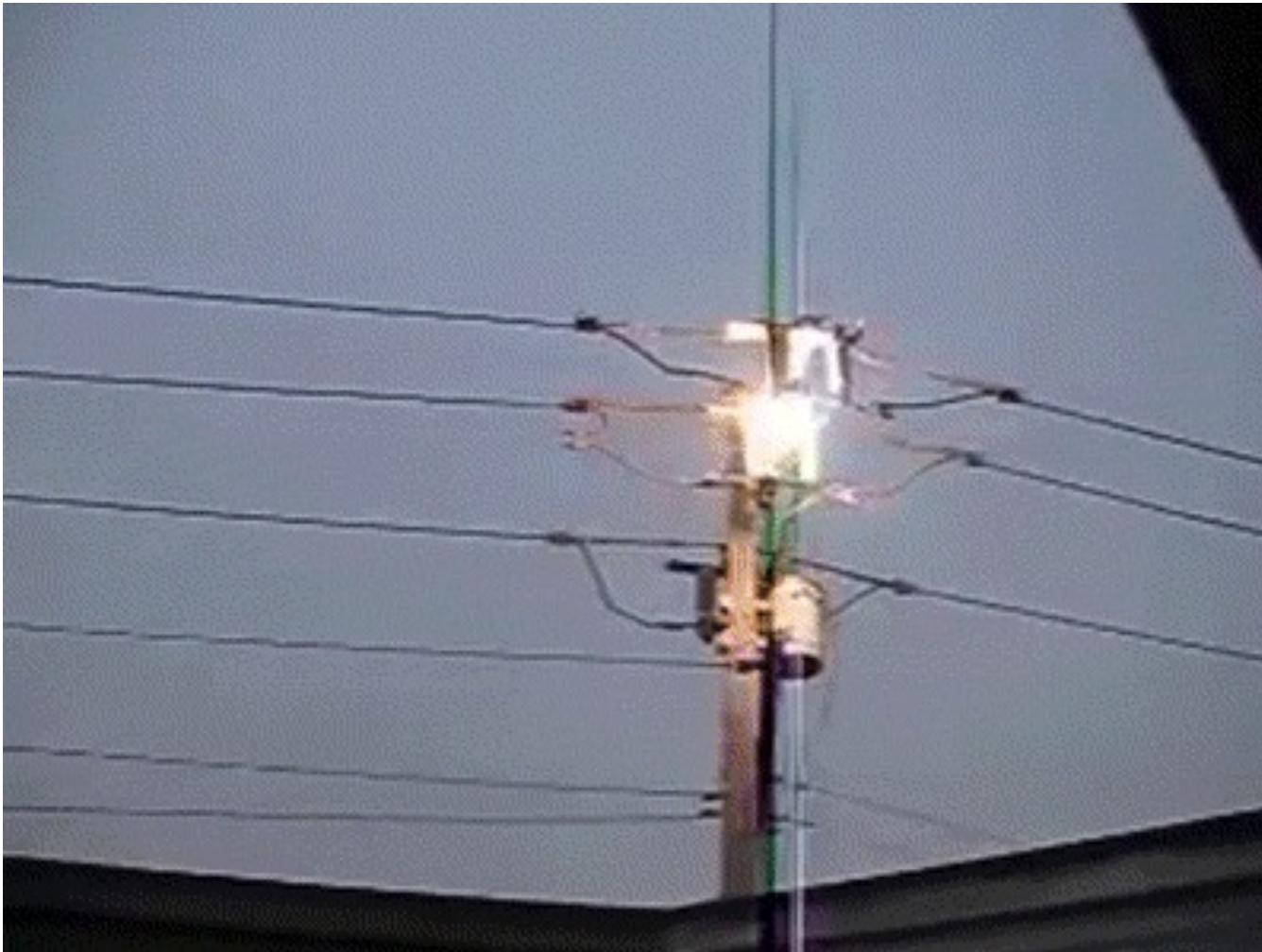
Today's Topics

- Explosion of transformers
- GPT
- BERT
- ViT
- Limitations of transformer models

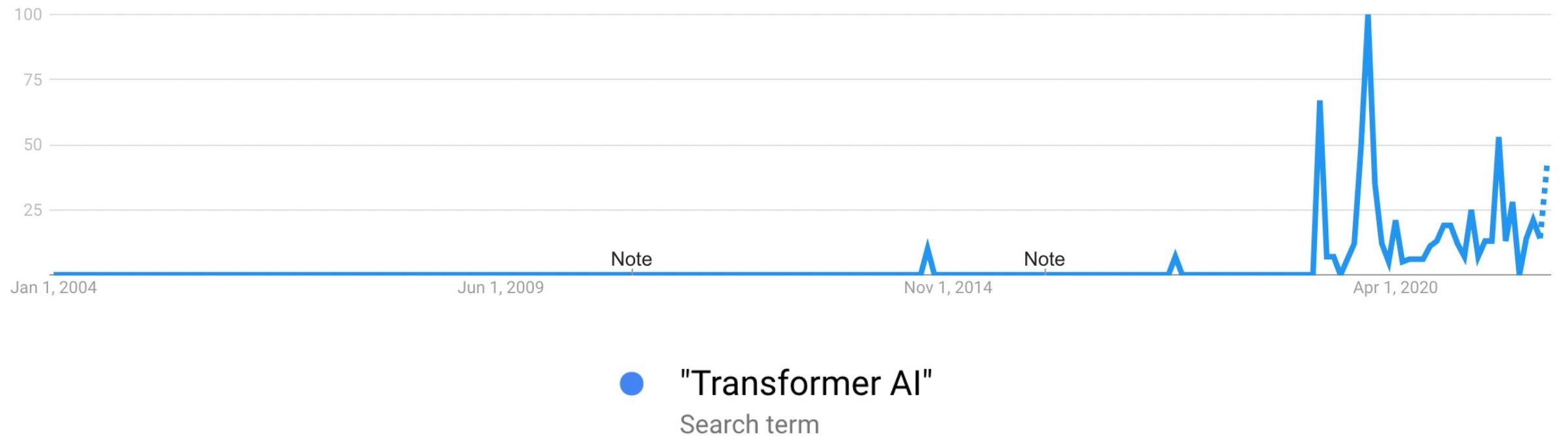
Today's Topics

- Explosion of transformers
- GPT
- BERT
- ViT
- Limitations of transformer models

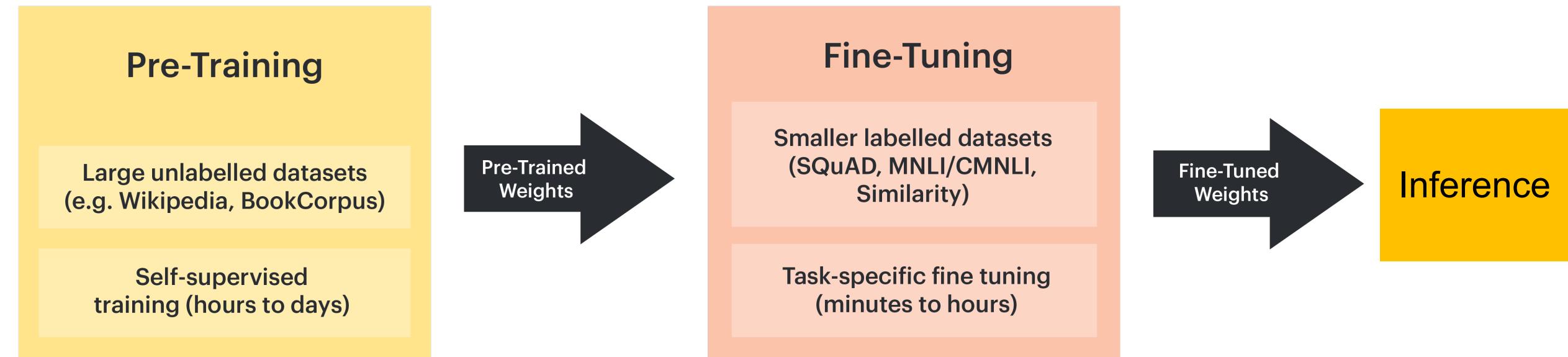
Explosion of Transformers in Society



Popularity of Transformers in Society

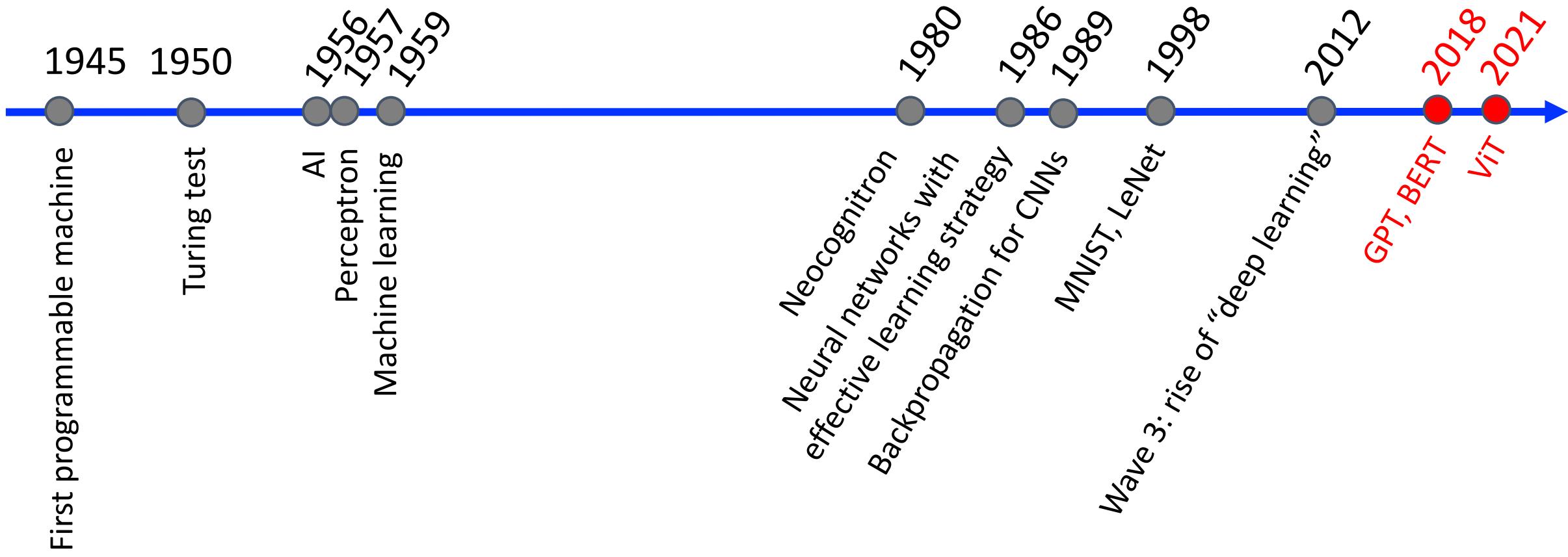


Today's Focus: Methods that Perform Pretraining and then Fine-tuning



Transformers can provide better embeddings for downstream tasks since they capture context (i.e., unlike context-free embeddings such as word2vec, the word embedding is different for a word used in different contexts... e.g., “I arrived at the bank after crossing the river/street”)

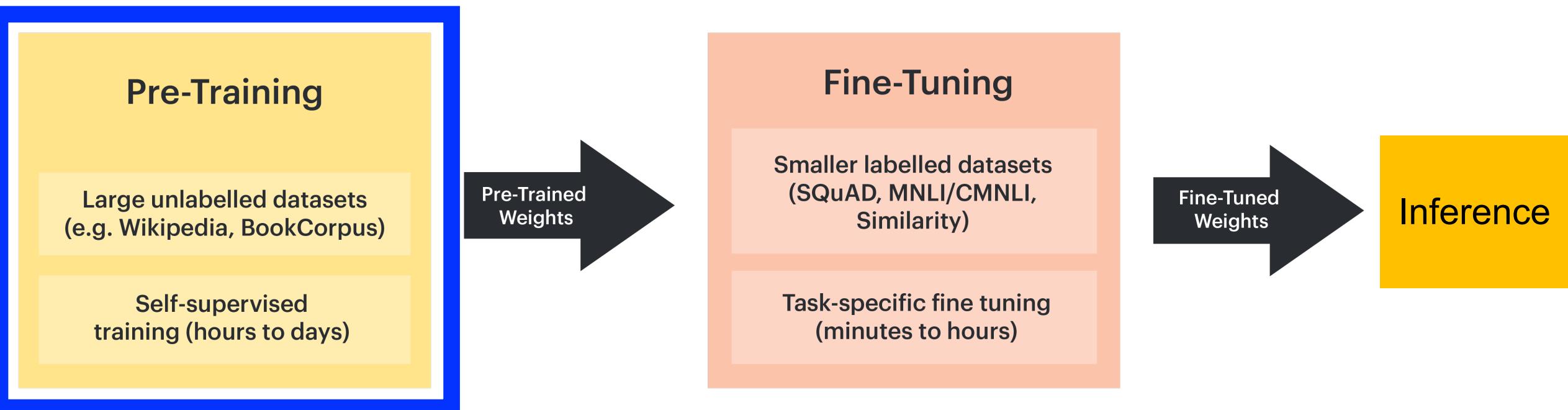
Today's Focus: Historical Context



Today's Topics

- Explosion of transformers
- GPT
- BERT
- ViT
- Limitations of transformer models

GPT: Generative Pre-Training

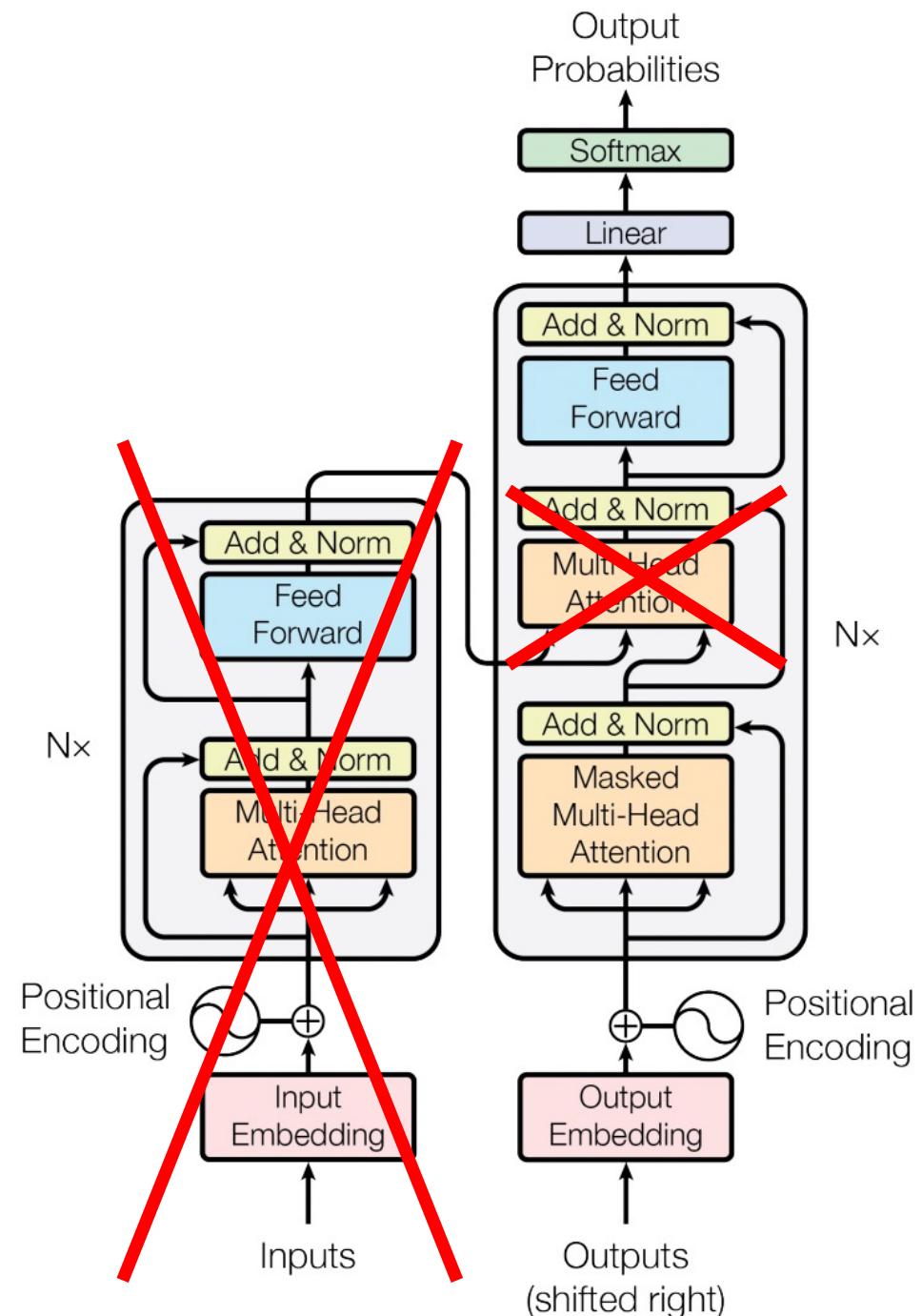


Task: Predict Next Word Given Previous Ones

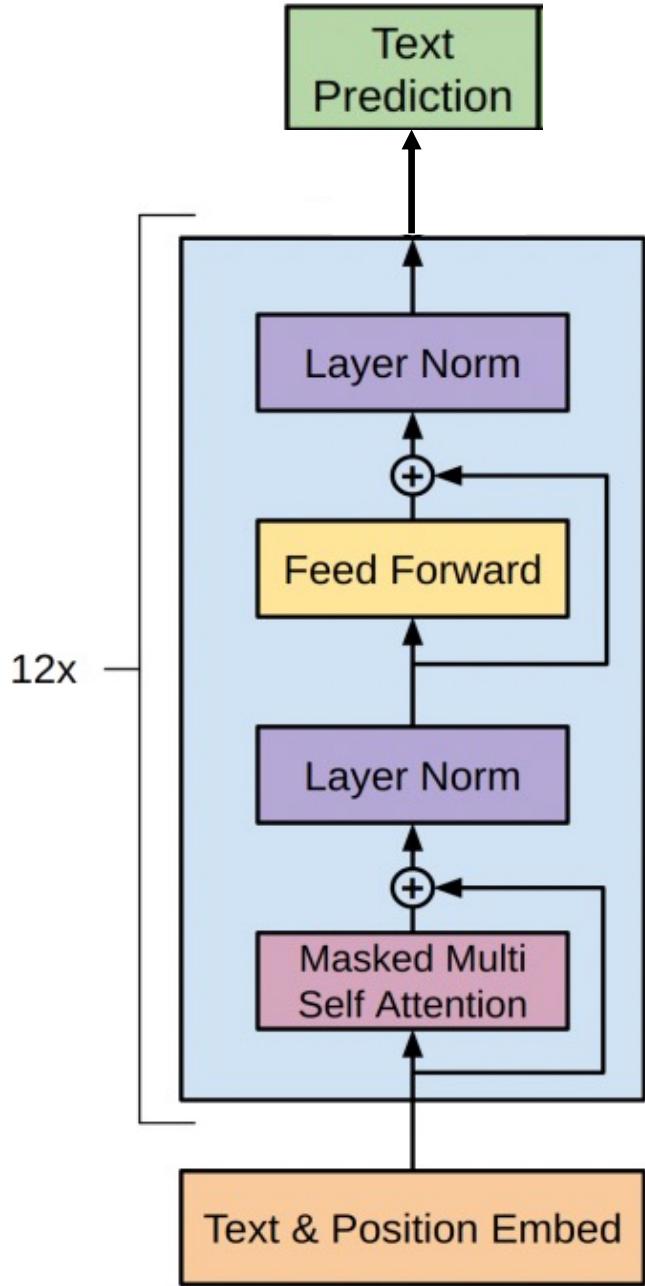
e.g.,

1. Background music from a _____
2. Many people danced around the _____
3. I practiced for many years to learn how to play the _____

Architecture: Decoder from Pioneering Transformer



Architecture



Architecture: Input & Output

EMBEDDING
WITH TIME
SIGNAL

$$\mathbf{x}_1 = \begin{array}{|c|c|c|c|}\hline \textcolor{lightgreen}{\square} & \textcolor{lightgreen}{\square} & \textcolor{lightgreen}{\square} & \textcolor{lightgreen}{\square} \\ \hline \end{array}$$

$$\mathbf{x}_2 = \begin{array}{|c|c|c|c|}\hline \textcolor{lightgreen}{\square} & \textcolor{lightgreen}{\square} & \textcolor{lightgreen}{\square} & \textcolor{lightgreen}{\square} \\ \hline \end{array}$$

$$\mathbf{x}_3 = \begin{array}{|c|c|c|c|}\hline \textcolor{lightgreen}{\square} & \textcolor{lightgreen}{\square} & \textcolor{lightgreen}{\square} & \textcolor{lightgreen}{\square} \\ \hline \end{array}$$

POSITIONAL
ENCODING

$$\mathbf{t}_1 = \begin{array}{|c|c|c|c|}\hline \textcolor{yellow}{\square} & \textcolor{yellow}{\square} & \textcolor{yellow}{\square} & \textcolor{yellow}{\square} \\ \hline \end{array}$$

$$\mathbf{t}_2 = \begin{array}{|c|c|c|c|}\hline \textcolor{yellow}{\square} & \textcolor{yellow}{\square} & \textcolor{yellow}{\square} & \textcolor{yellow}{\square} \\ \hline \end{array}$$

$$\mathbf{t}_3 = \begin{array}{|c|c|c|c|}\hline \textcolor{yellow}{\square} & \textcolor{yellow}{\square} & \textcolor{yellow}{\square} & \textcolor{yellow}{\square} \\ \hline \end{array}$$

EMBEDDINGS

$$\mathbf{x}_1 + \begin{array}{|c|c|c|c|}\hline \textcolor{lightgreen}{\square} & \textcolor{lightgreen}{\square} & \textcolor{lightgreen}{\square} & \textcolor{lightgreen}{\square} \\ \hline \end{array}$$

$$\mathbf{x}_2 + \begin{array}{|c|c|c|c|}\hline \textcolor{lightgreen}{\square} & \textcolor{lightgreen}{\square} & \textcolor{lightgreen}{\square} & \textcolor{lightgreen}{\square} \\ \hline \end{array}$$

$$\mathbf{x}_3 + \begin{array}{|c|c|c|c|}\hline \textcolor{lightgreen}{\square} & \textcolor{lightgreen}{\square} & \textcolor{lightgreen}{\square} & \textcolor{lightgreen}{\square} \\ \hline \end{array}$$

INPUT

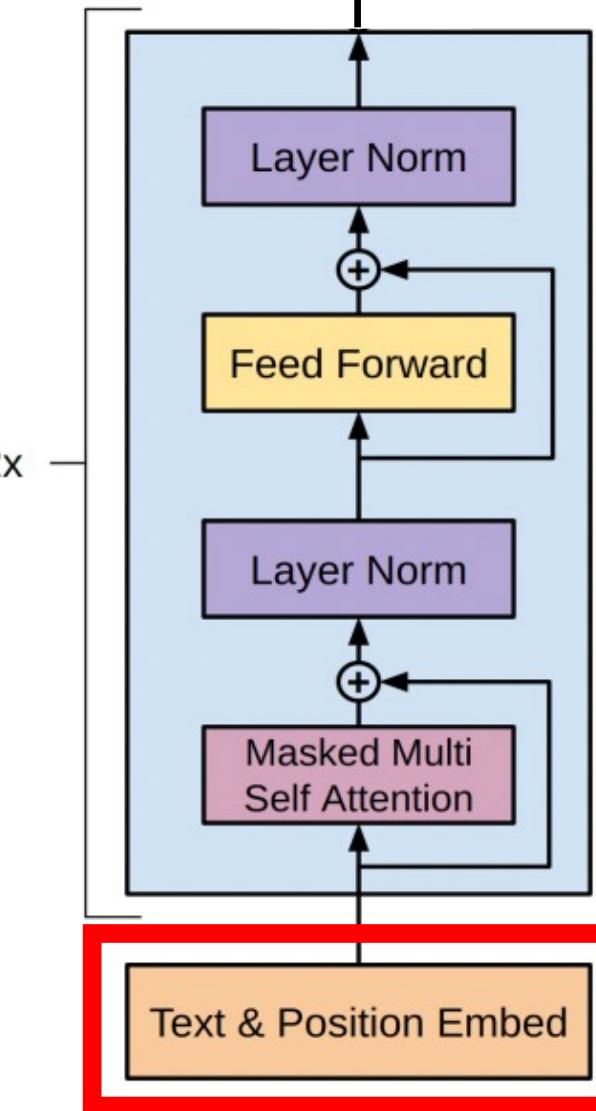
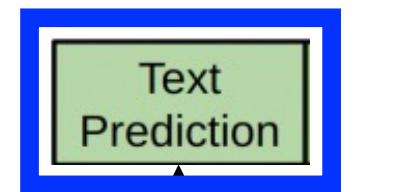
Je

suis

étudiant

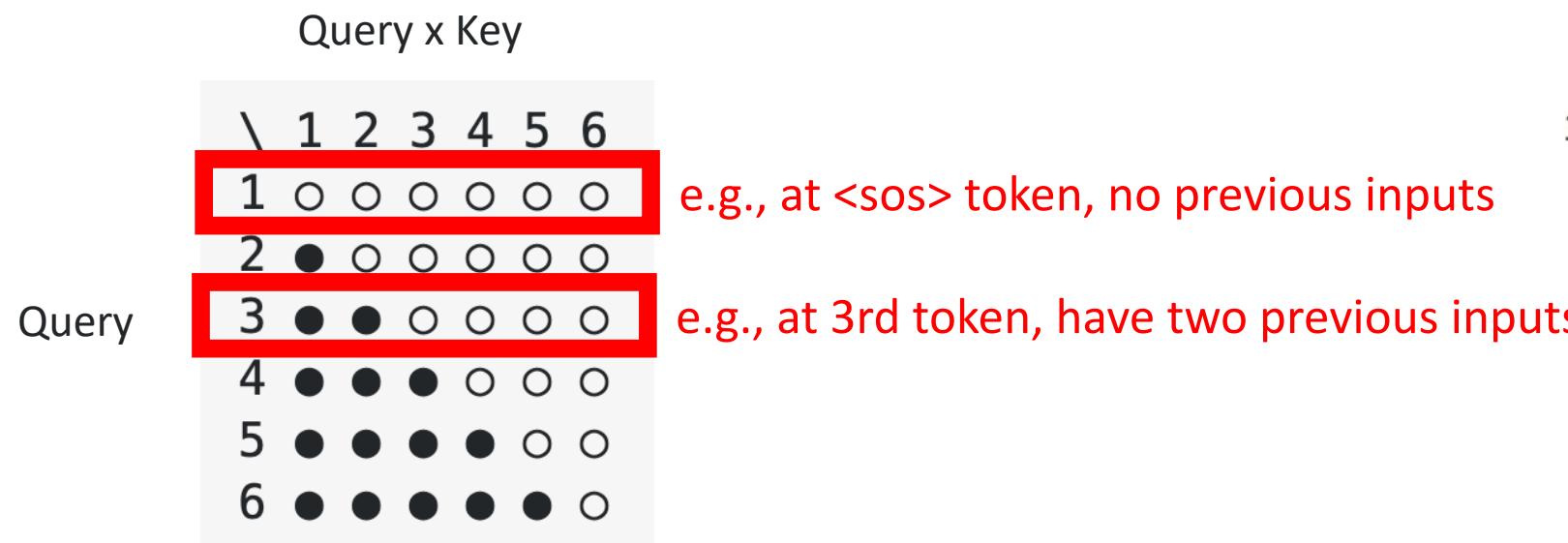
<http://jalammar.github.io/illustrated-transformer/>

Softmax layer determines
next predicted word at
each time step

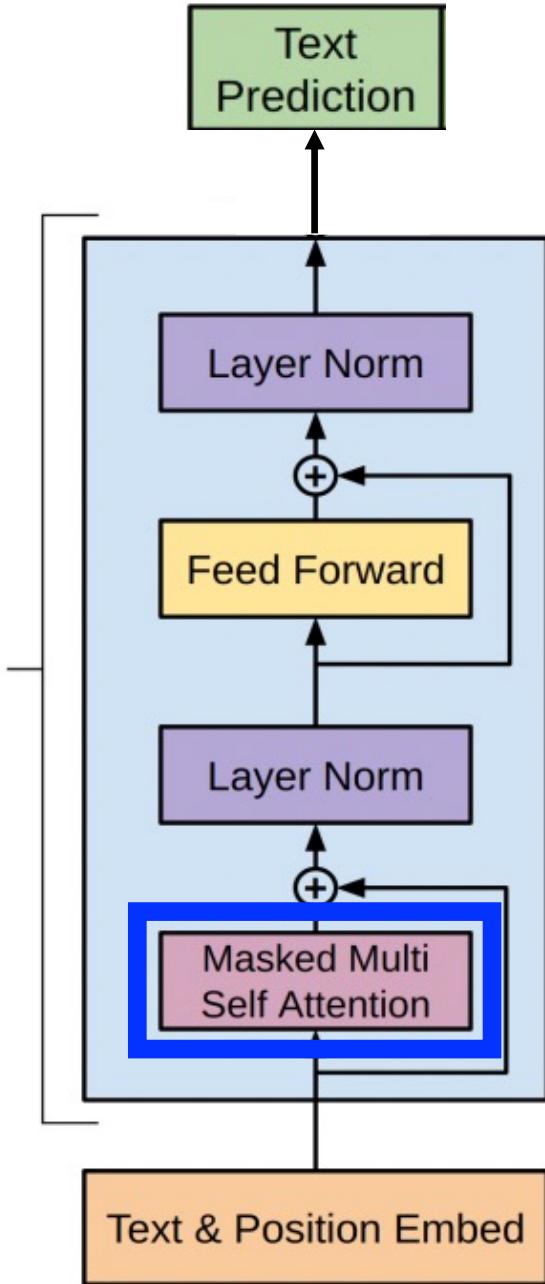


Architecture: Masked Attention

Limit each word's new representation to only reflect earlier words
(mimics inference time when only previous tokens can be seen):



<https://stackoverflow.com/questions/64799622/how-is-the-gpts-masked-self-attention-is-utilized-on-fine-tuning-inference>

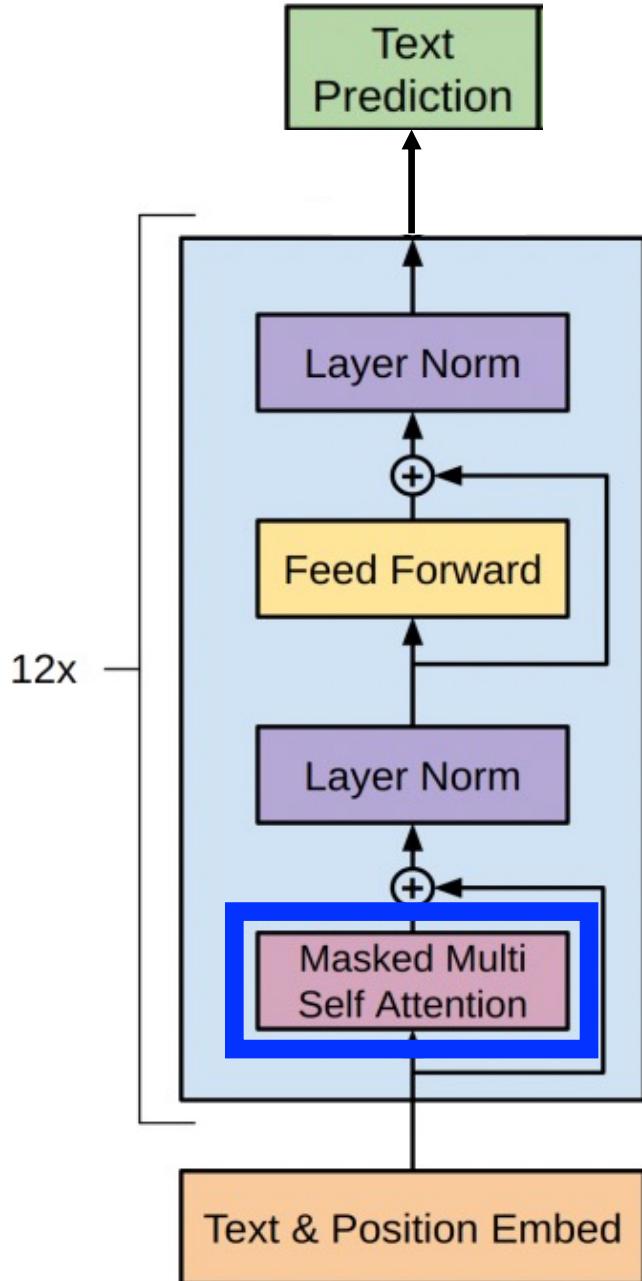


Architecture: Masked Attention

Limit each word's new representation to only reflect earlier words
(mimics inference time when only previous tokens can be seen):

		Query x Key						
		\	1	2	3	4	5	6
Query		1	○	○	○	○	○	○
3		2	●	○	○	○	○	○
5		3	●	●	○	○	○	○
7		4	●	●	●	○	○	○
9		5	●	●	●	●	○	○
11		6	●	●	●	●	●	○

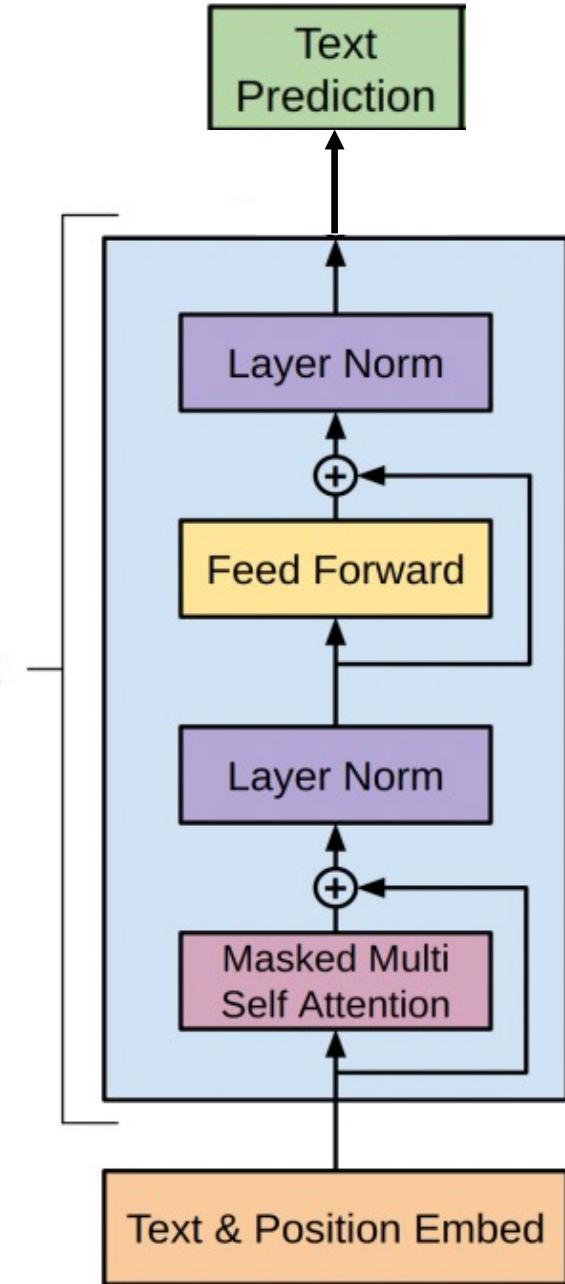
Masked out scores are represented as negative infinity so the softmax result (i.e., attention weight) returns 0



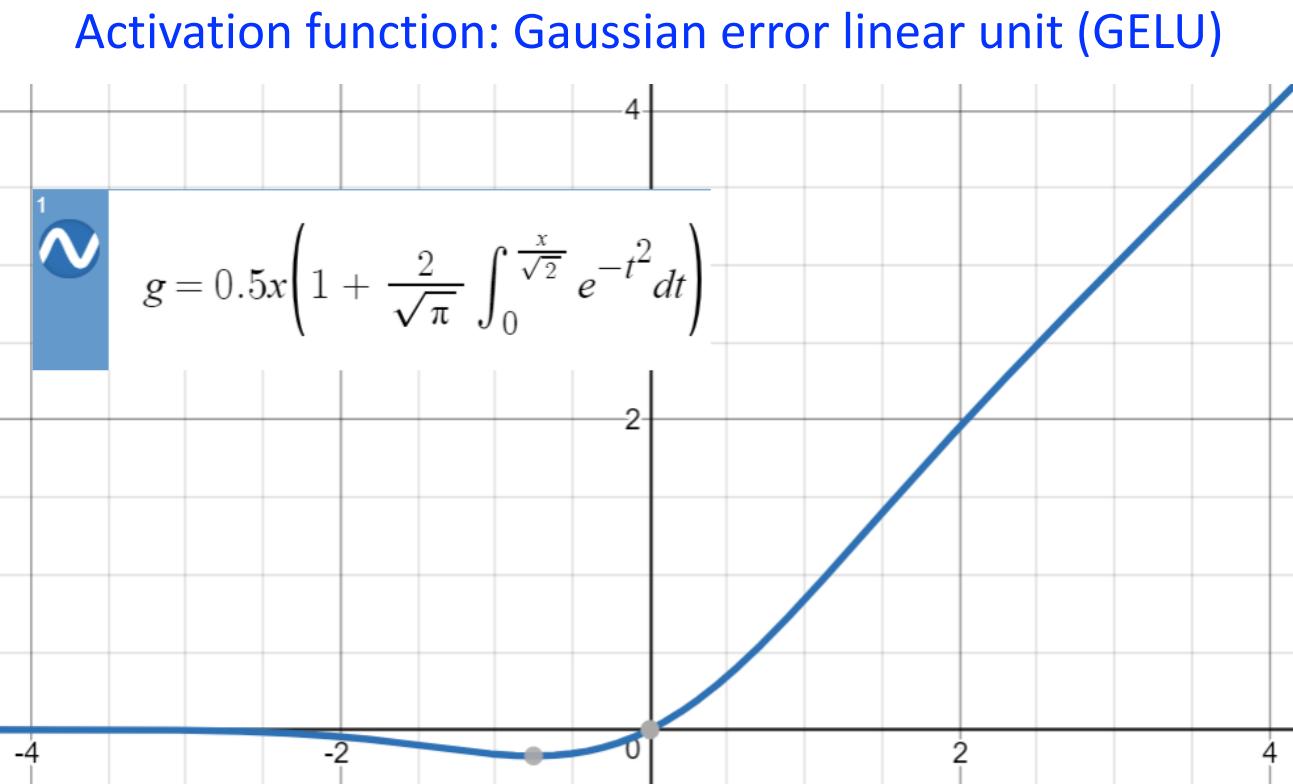
<https://stackoverflow.com/questions/64799622/how-is-the-gpts-masked-self-attention-is-utilized-on-fine-tuning-inference>

Training

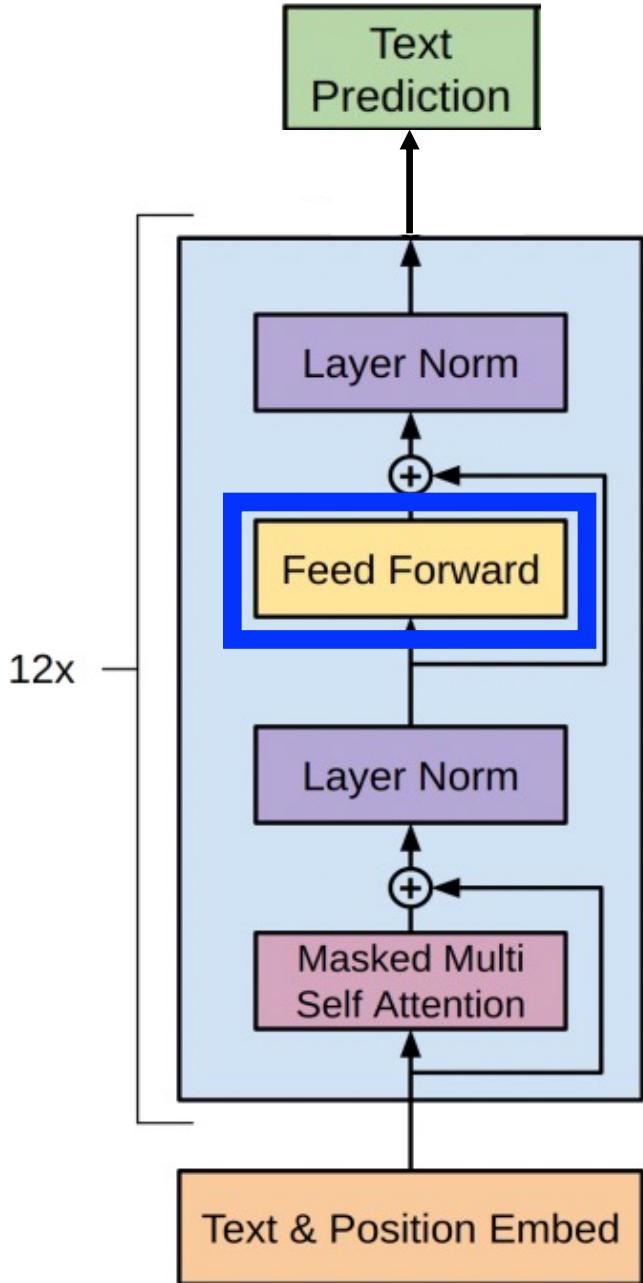
- Dataset: 800M words from BooksCorpus (>7,000 books)
- Optimizer: Adam
- Training duration: 100 epochs



Implementation Details



<https://datascience.stackexchange.com/questions/49522/what-is-gelu-activation>



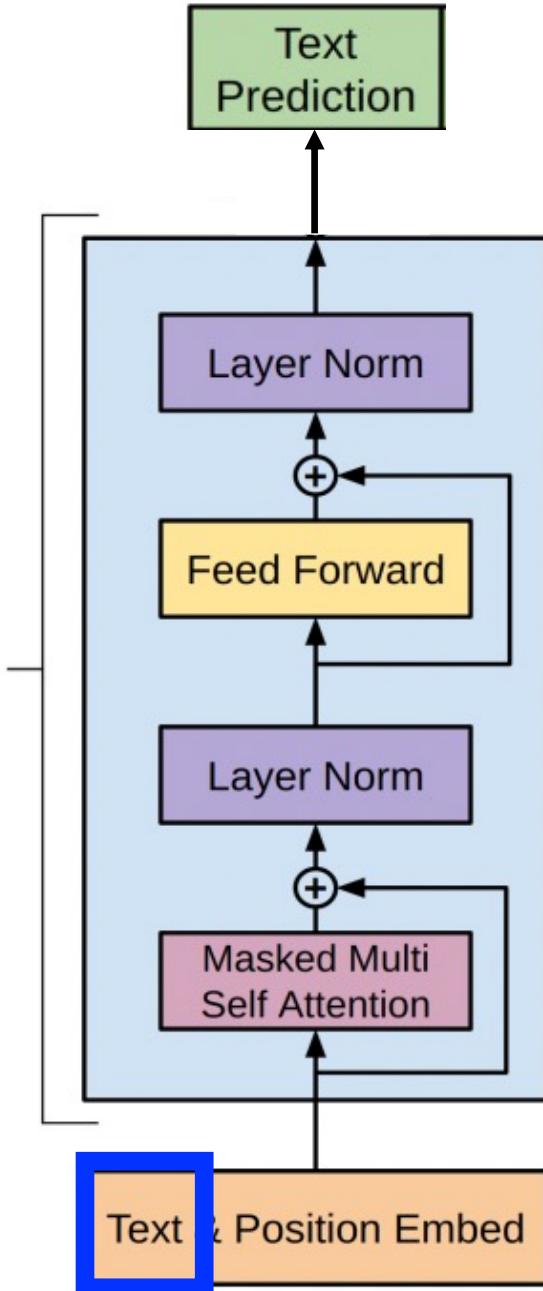
Implementation Details

Avoid out of vocabulary tokens with subword tokenization (byte pair encoding):

1. Identify all tokens in the training data with their frequency
2. Define vocabulary size; e.g., 14
3. Add all characters in the tokenized input to the vocabulary; e.g.,

Character sequence	Cost
C o s t	2
b e s t	2
m e n u	1
m e n	1
c a m e l	1

https://static.packt-cdn.com/downloads/9781838821593_ColorImages.pdf



Implementation Details

Avoid out of vocabulary tokens with subword tokenization (byte pair encoding):

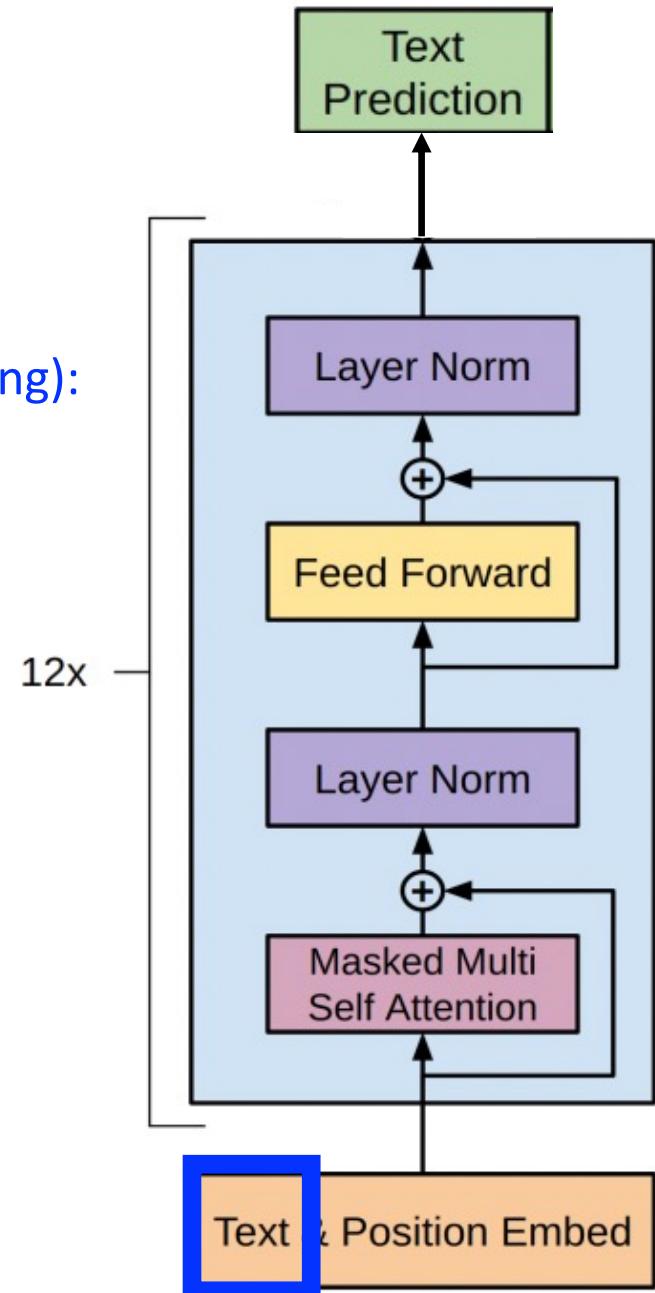
1. Identify all tokens in the training data with their frequency
2. Define vocabulary size; e.g., 14
3. Add all characters in the tokenized input to the vocabulary; e.g.,
4. Until vocabulary is filled, add merged highest frequency symbol pairs

e.g., What are the highest frequency symbol pairs?

Character sequence	Cost
C o s t	2
b e s t	2
m e n u	1
m e n	1
c a m e l	1

Vocabulary

a, b, c, e, l, m, n, o, s, t, u



https://static.packt-cdn.com/downloads/9781838821593_ColorImages.pdf

Implementation Details

Avoid out of vocabulary tokens with subword tokenization (byte pair encoding):

1. Identify all tokens in the training data with their frequency
2. Define vocabulary size; e.g., 14
3. Add all characters in the tokenized input to the vocabulary; e.g.,
4. Until vocabulary is filled, add merged highest frequency symbol pairs

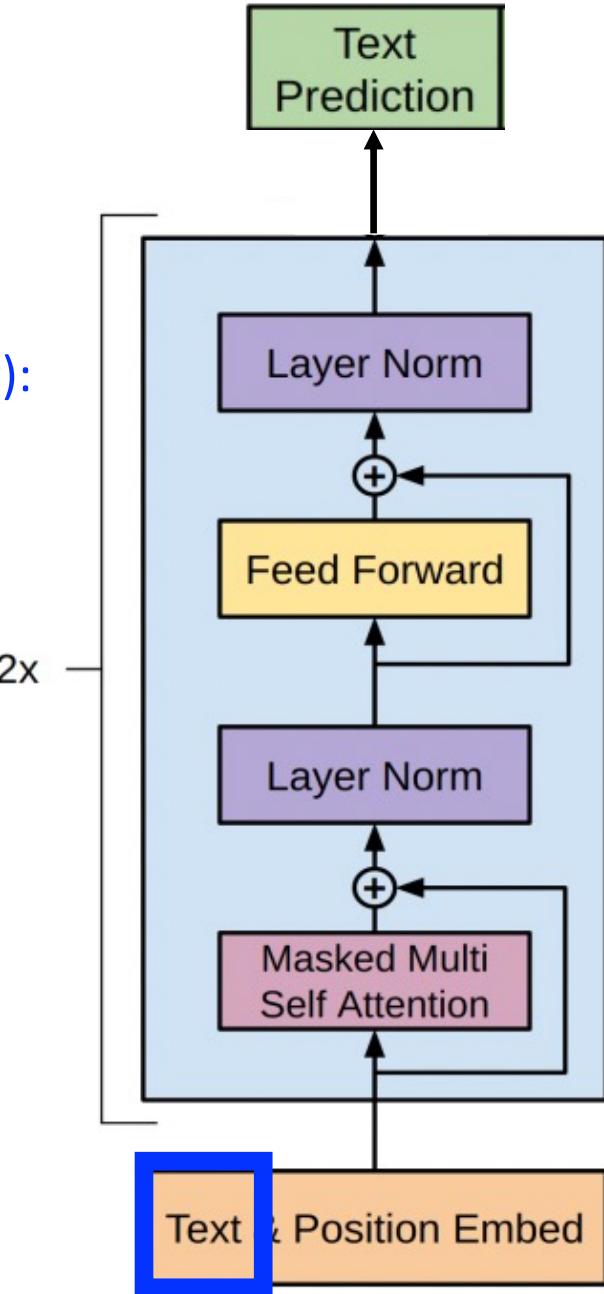
e.g., What are the highest frequency symbol pairs?

Character sequence	Cost
C o s t	2
b e s t	2
m e n u	1
m e n	1
c a m e l	1

Vocabulary

a, b, c, e, l, m, n, o, s, t, u, st

https://static.packt-cdn.com/downloads/9781838821593_ColorImages.pdf



Implementation Details

Avoid out of vocabulary tokens with subword tokenization (byte pair encoding):

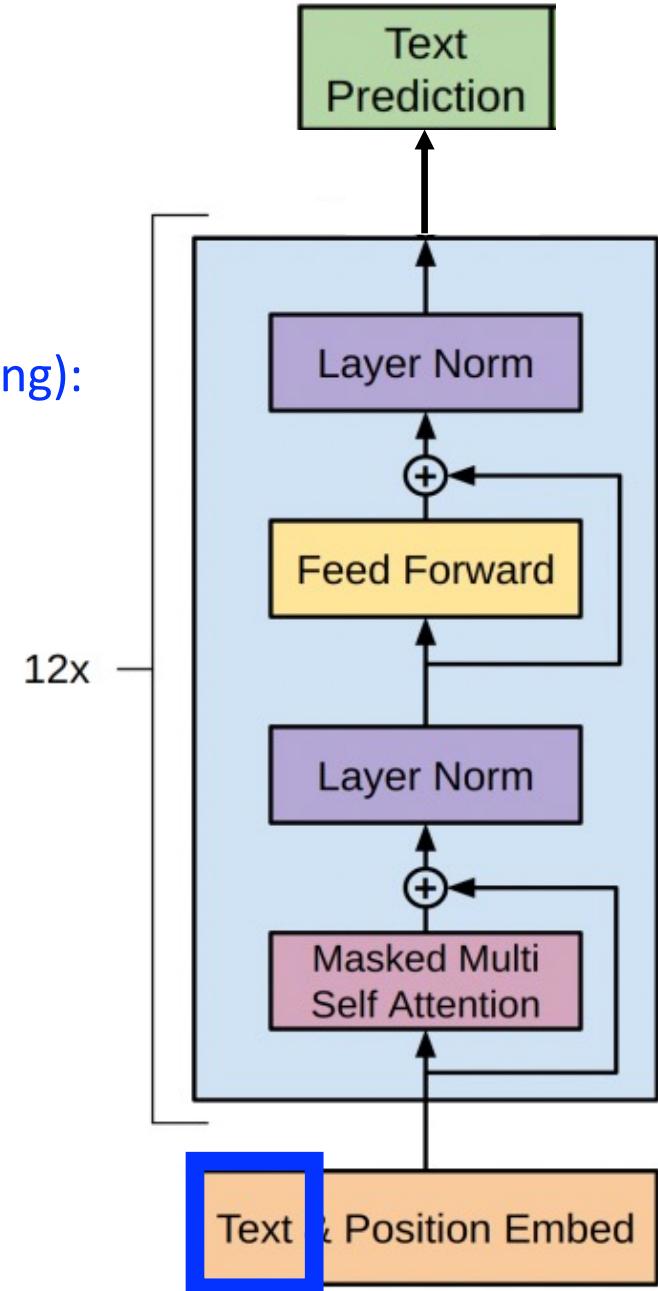
1. Identify all tokens in the training data with their frequency
2. Define vocabulary size; e.g., 14
3. Add all characters in the tokenized input to the vocabulary; e.g.,
4. Until vocabulary is filled, add merged highest frequency symbol pairs

e.g., What are the highest frequency symbol pairs?

Character sequence	Cost
C o s t	2
b e s t	2
men u	1
me n	1
c a m e l	1

Vocabulary

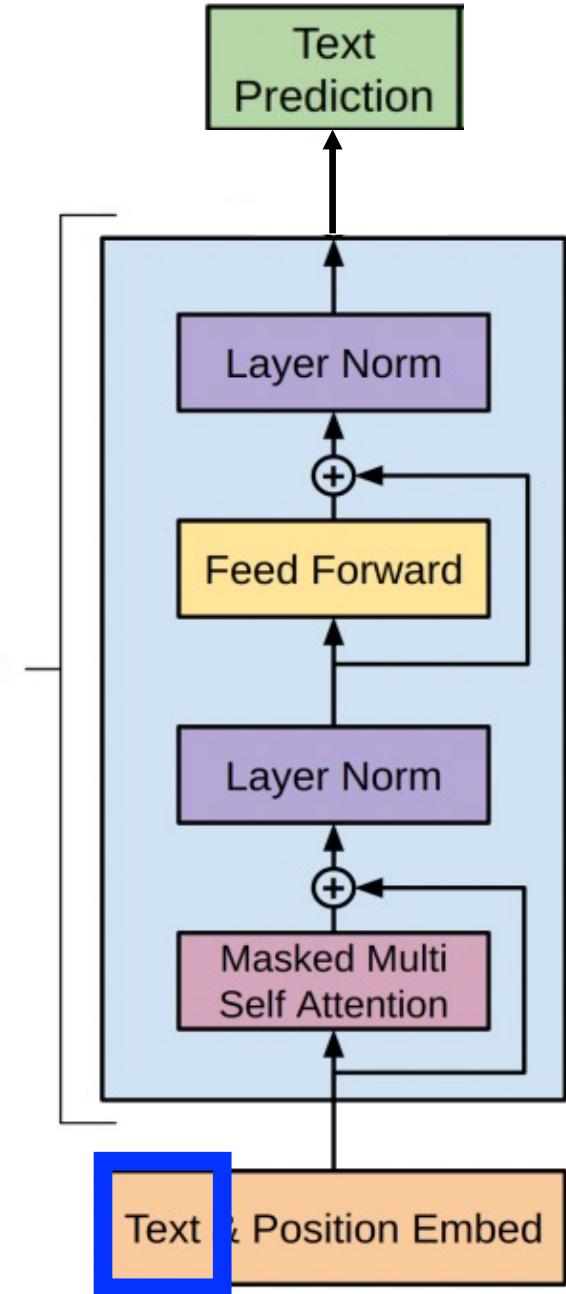
a, b, c, e, l, m, n, o, s, t, u, st, me, men



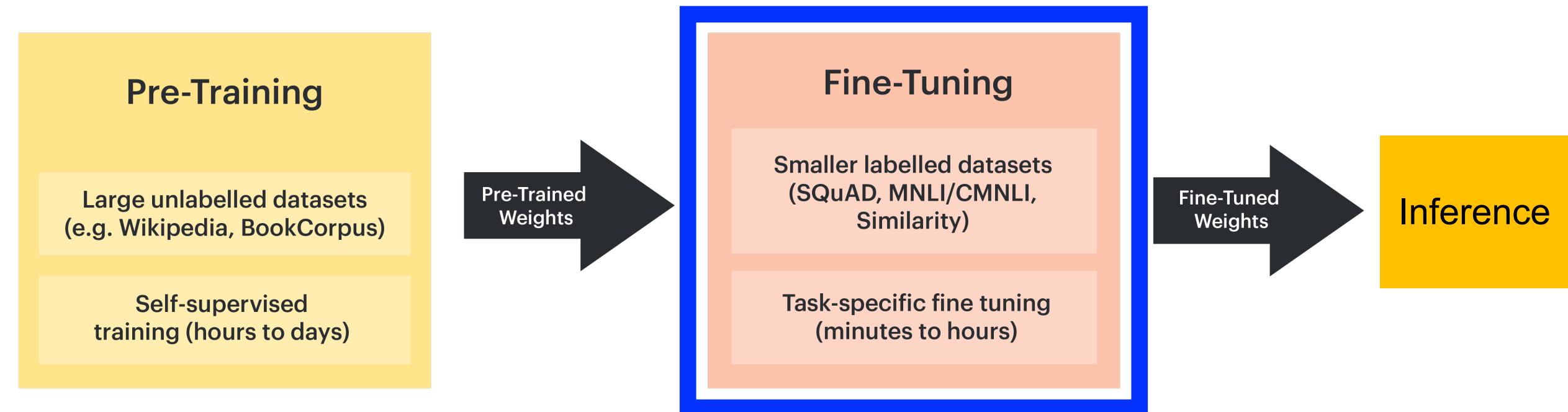
https://static.packt-cdn.com/downloads/9781838821593_ColorImages.pdf

Implementation Details

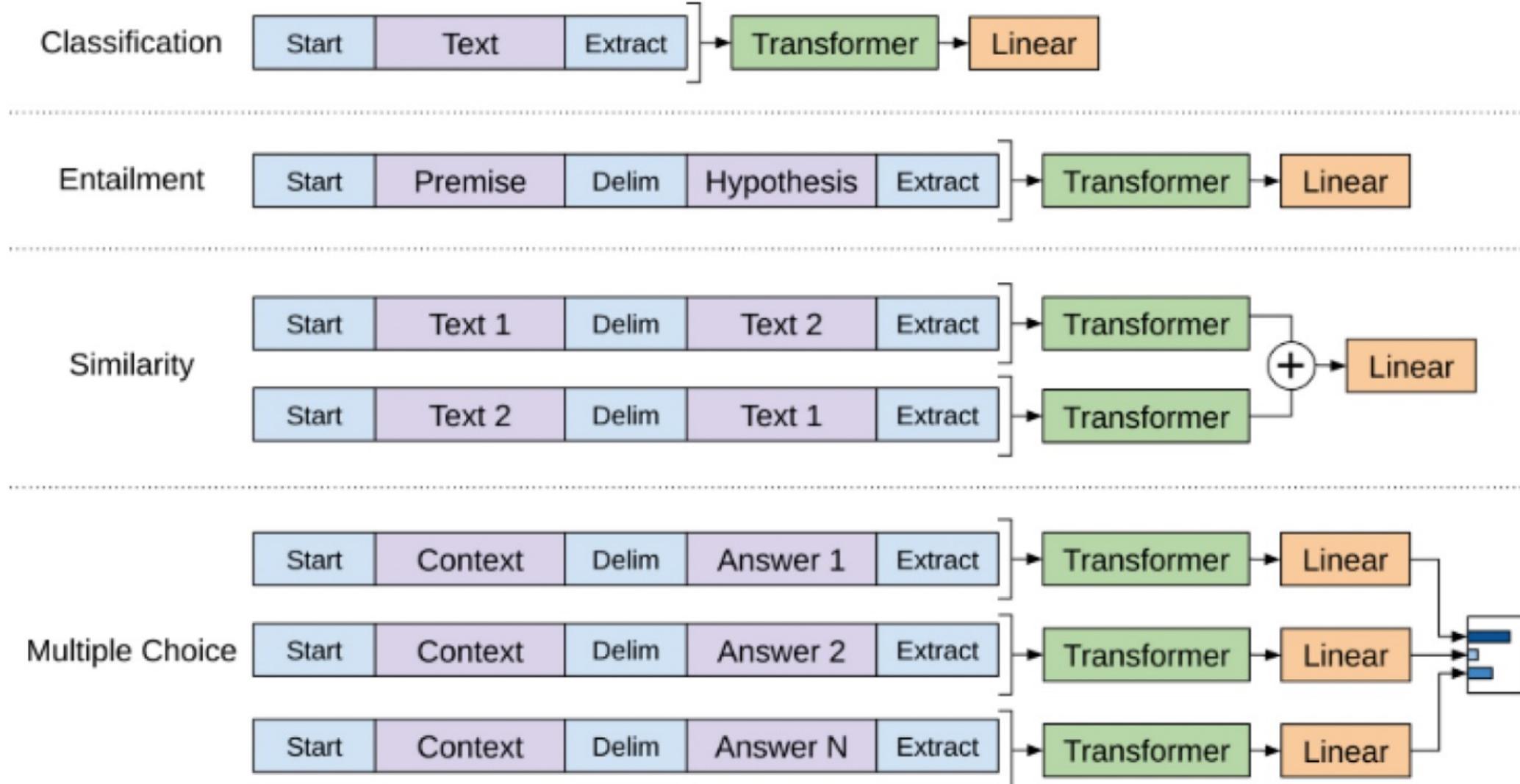
Avoid out of vocabulary tokens with subword tokenization (byte pair encoding):
- 40,000 merges used



GPT: Generative Pre-Training



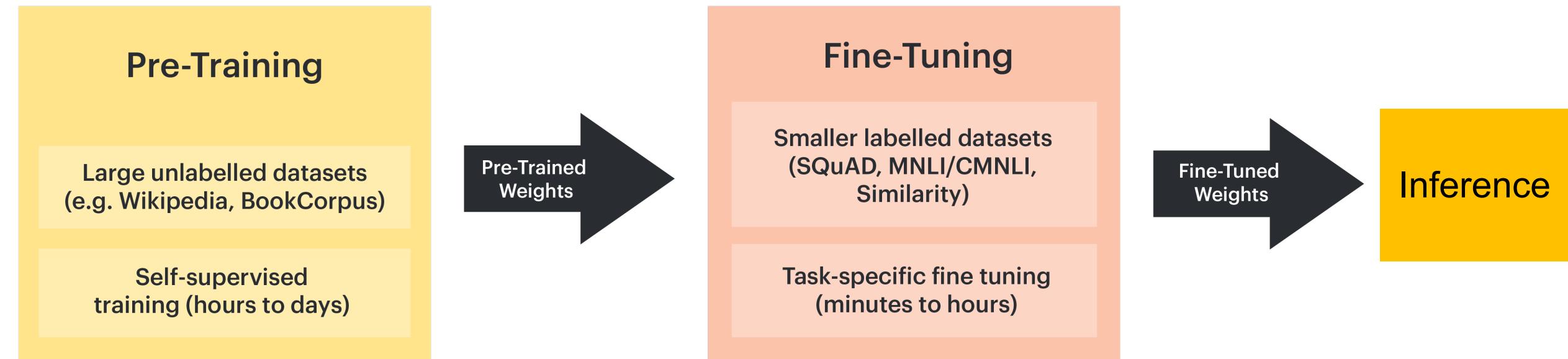
Fine-Tuning to Target Task (Softmax Output); i.e.,



Experimental Findings

Achieved the best performance on 9 NLP dataset challenges

GPT: Generative Pre-Training



Today's Topics

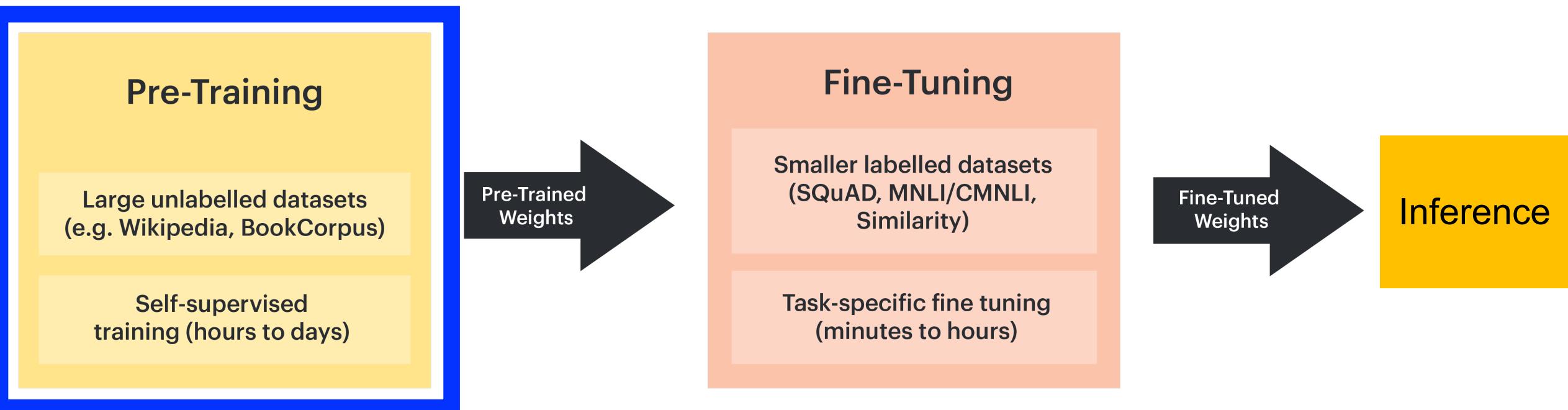
- Explosion of transformers
- GPT
- BERT
- ViT
- Limitations of transformer models

Motivation: Choose a Pretraining Task That Is Not Unidirectional

GPT's prediction of the next word given previous ones is unidirectional (left-to-right)

1. Background music from a _____
2. Many people danced around the _____
3. I practiced for many years to learn how to play the _____

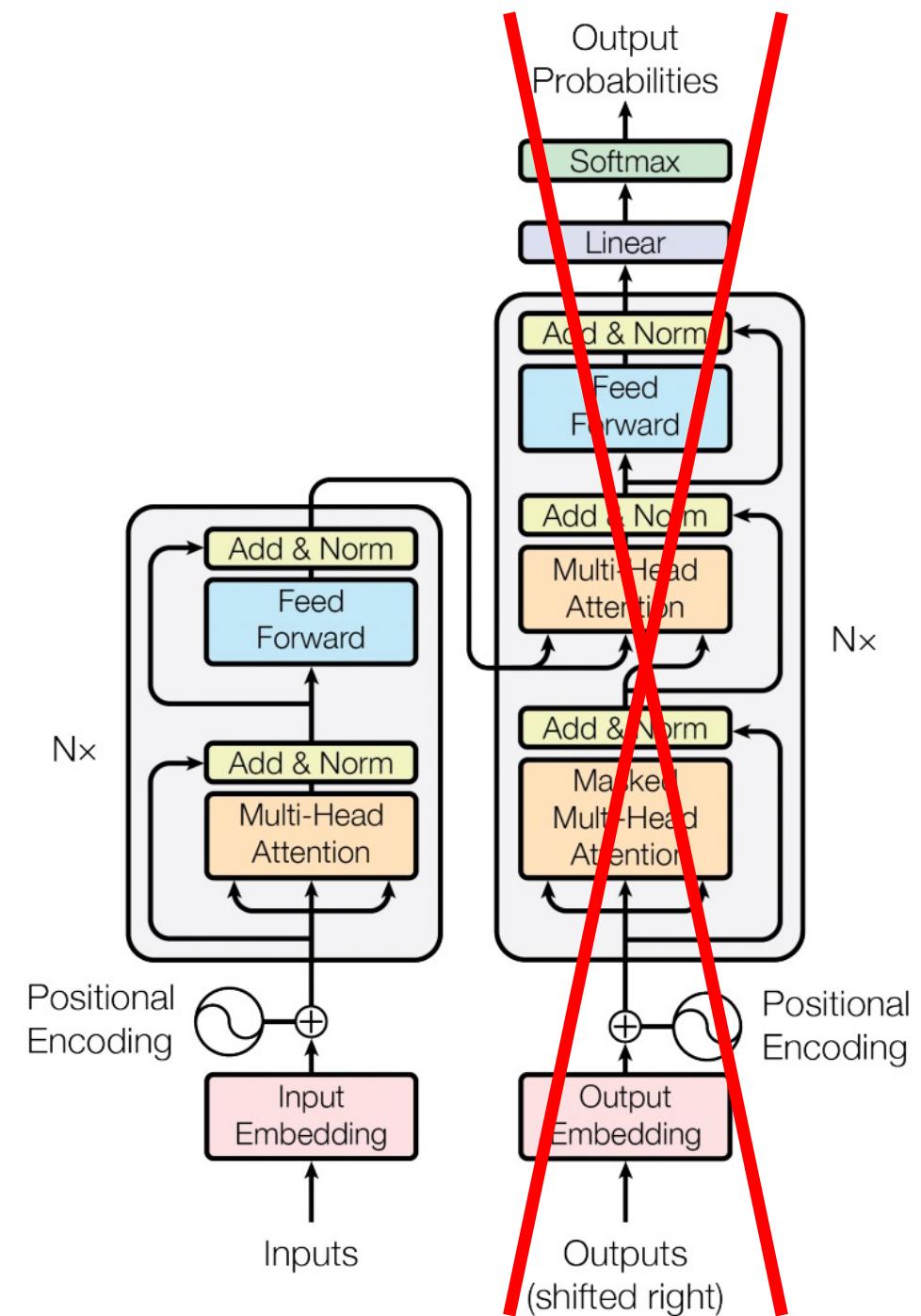
BERT: Bidirectional Encoder Representation from Transformer



Two Tasks

1. Predict masked token (key contribution)
2. Predict if one sentence follows a second sentence (augments understanding of how sentences relate)

Architecture: Encoder from Pioneering Transformer

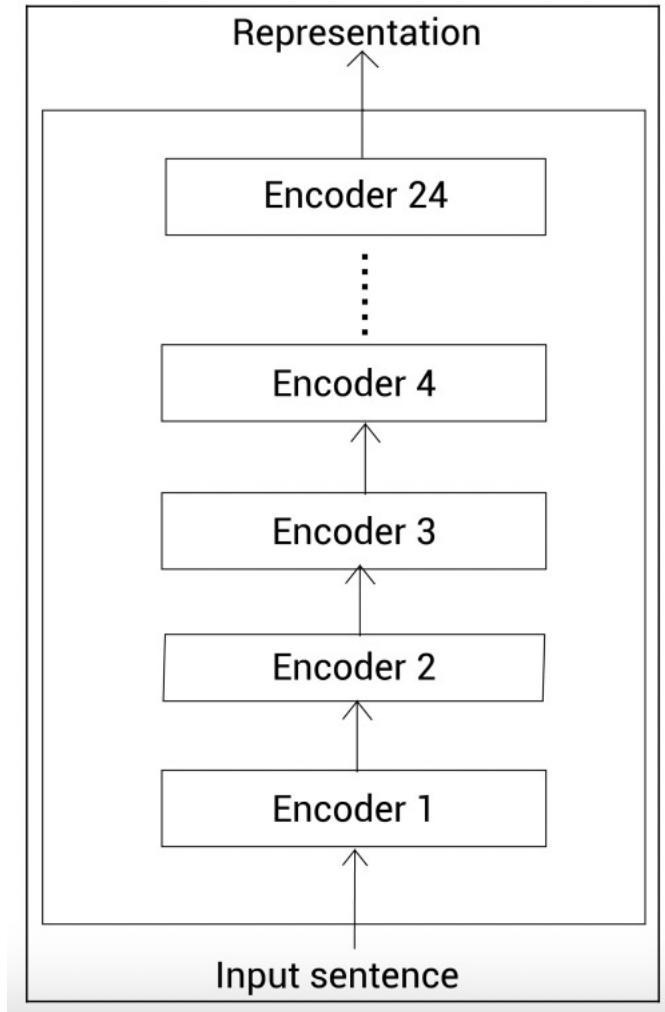


Architecture: Variants

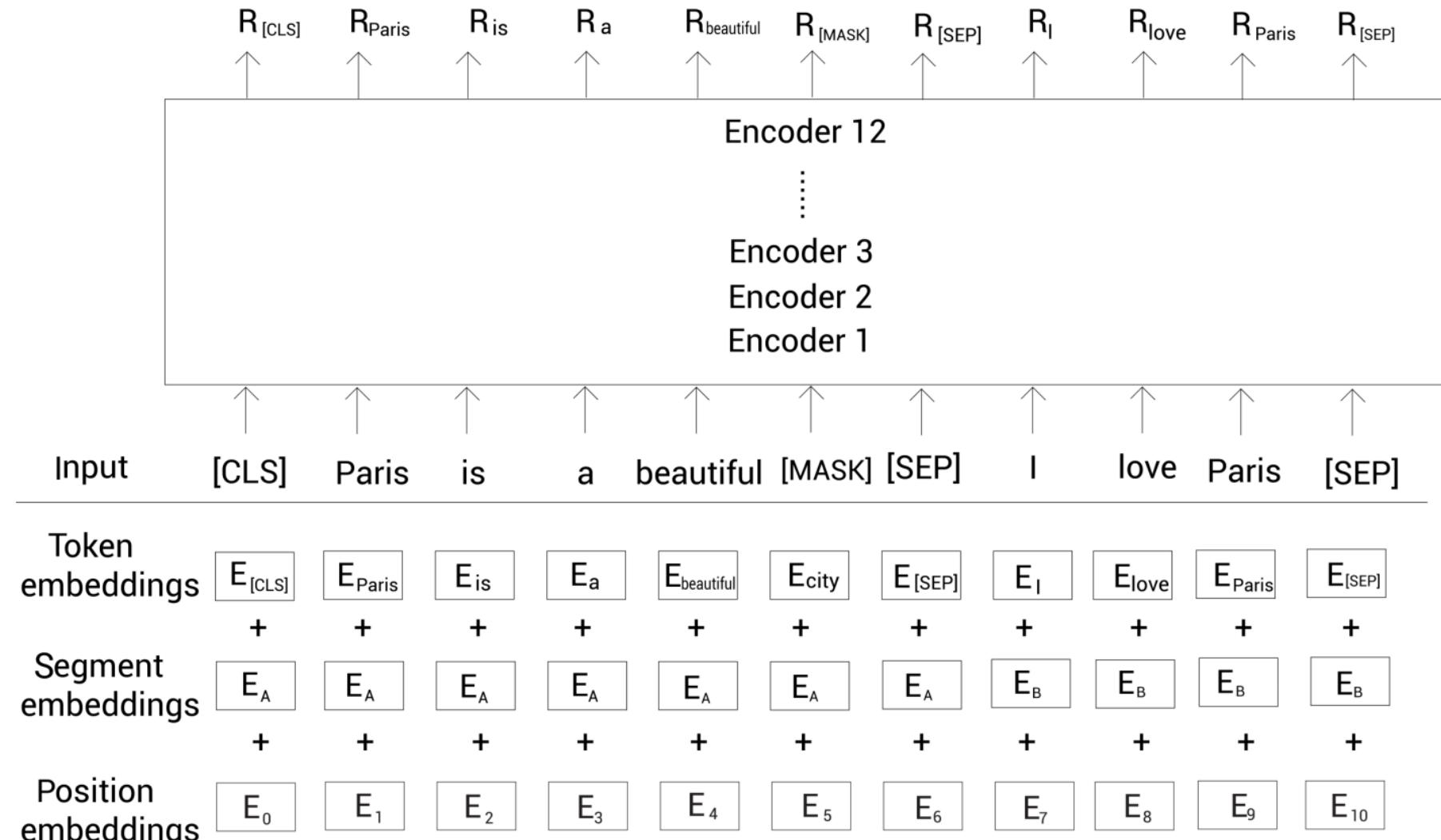
- L = number of stacked encoders
- H = number of hidden units in feedforward layer

	H=128	H=256	H=512	H=768
L=2	2/128(BERT-tiny)	2/256	2/512	2/768
L=4	4/128	4/256(BERT-mini)	4/512 (BERT-small)	4/768
L=6	6/128	6/256	6/512	6/768
L=8	8/128	8/256	8/512 (BERT-medium)	8/768
L=10	10/128	10/256	10/512	10/768
L=12	12/128	12/256	12/512	12/768(BERT-base)

BERT-large (H = 1024)

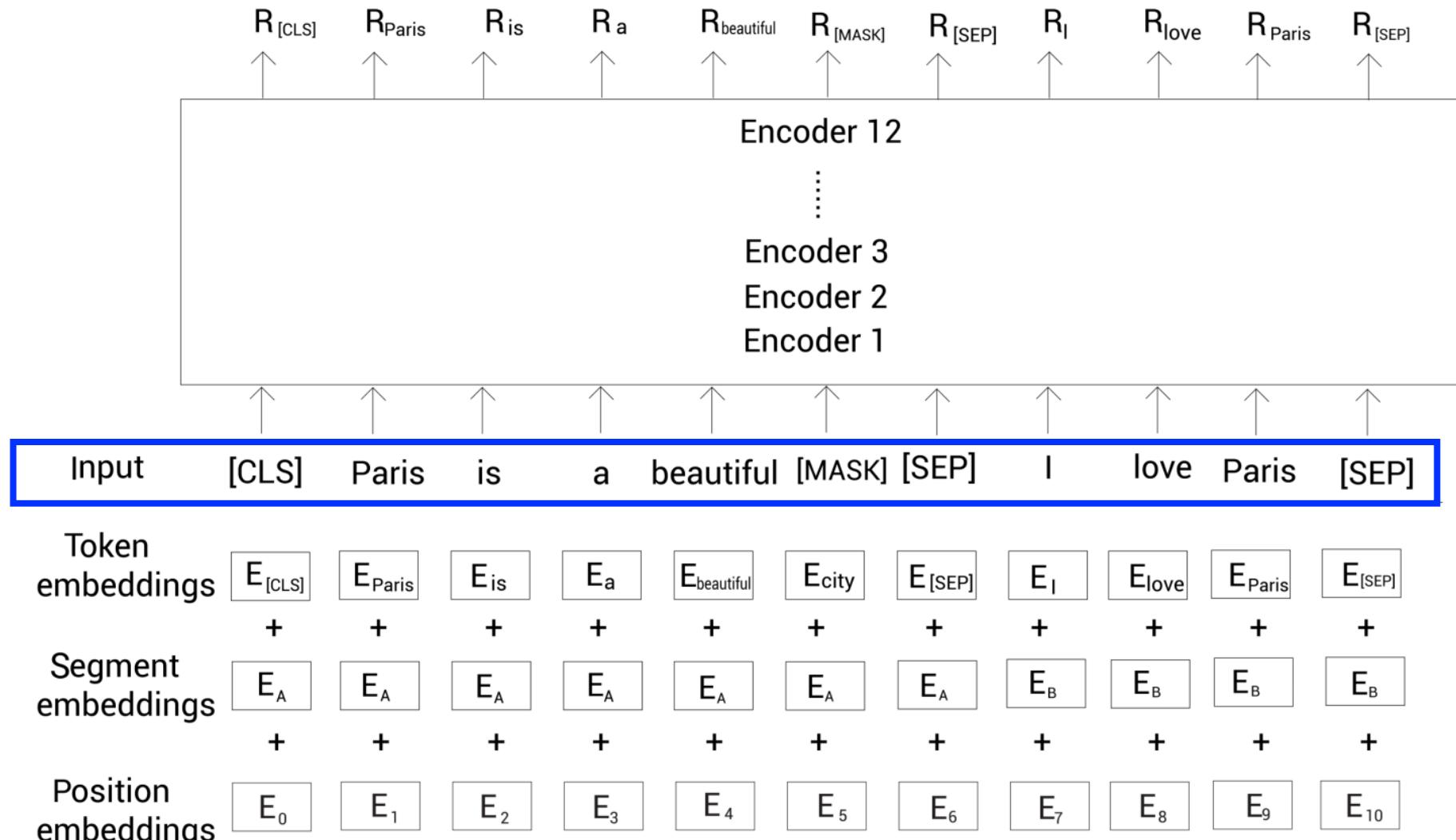


Architecture: BERT-Base (Matches Size of GPT)

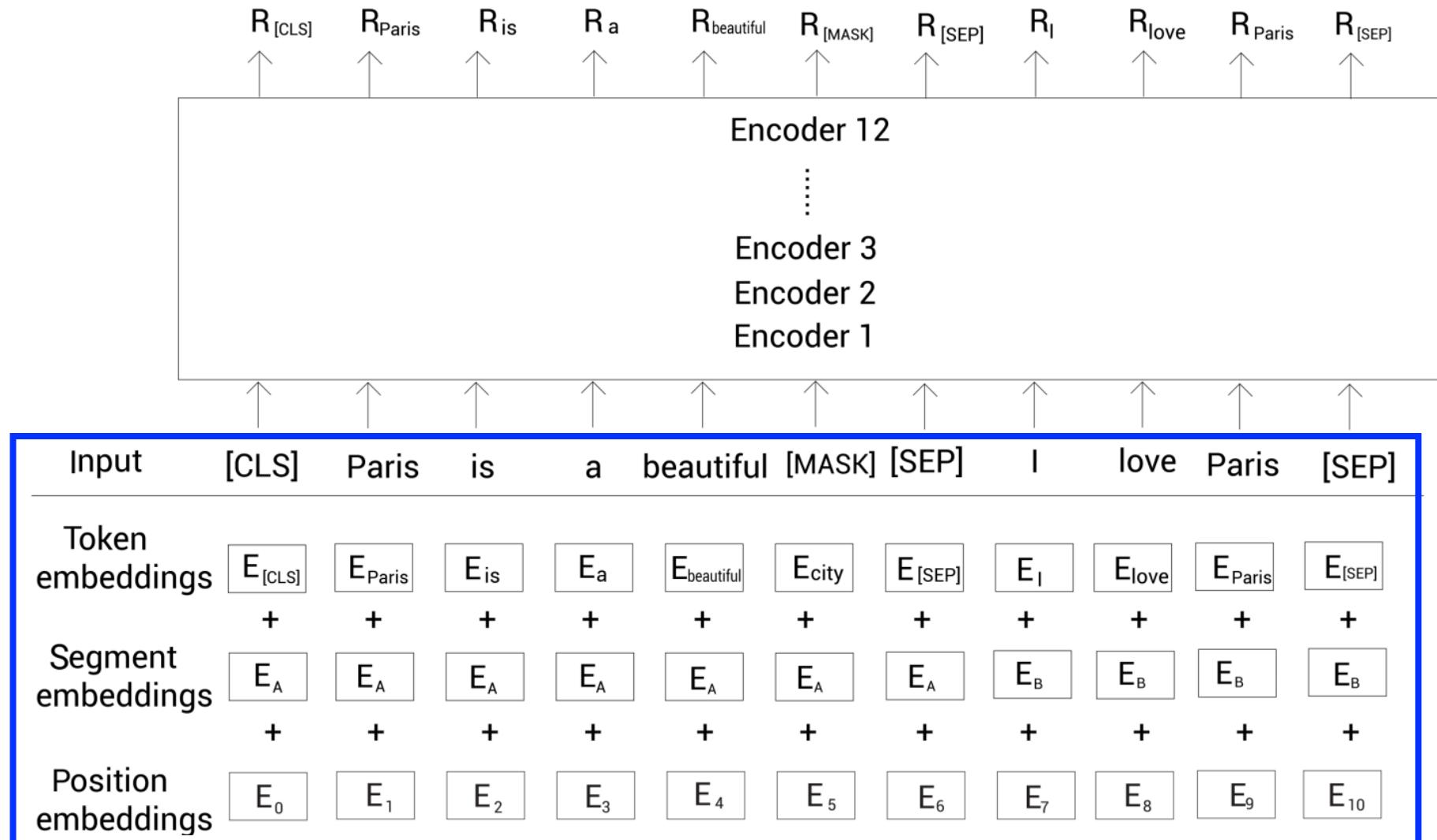


Architecture: Input

Two input sentences with [CLS] at the start and [SEP] between sentences



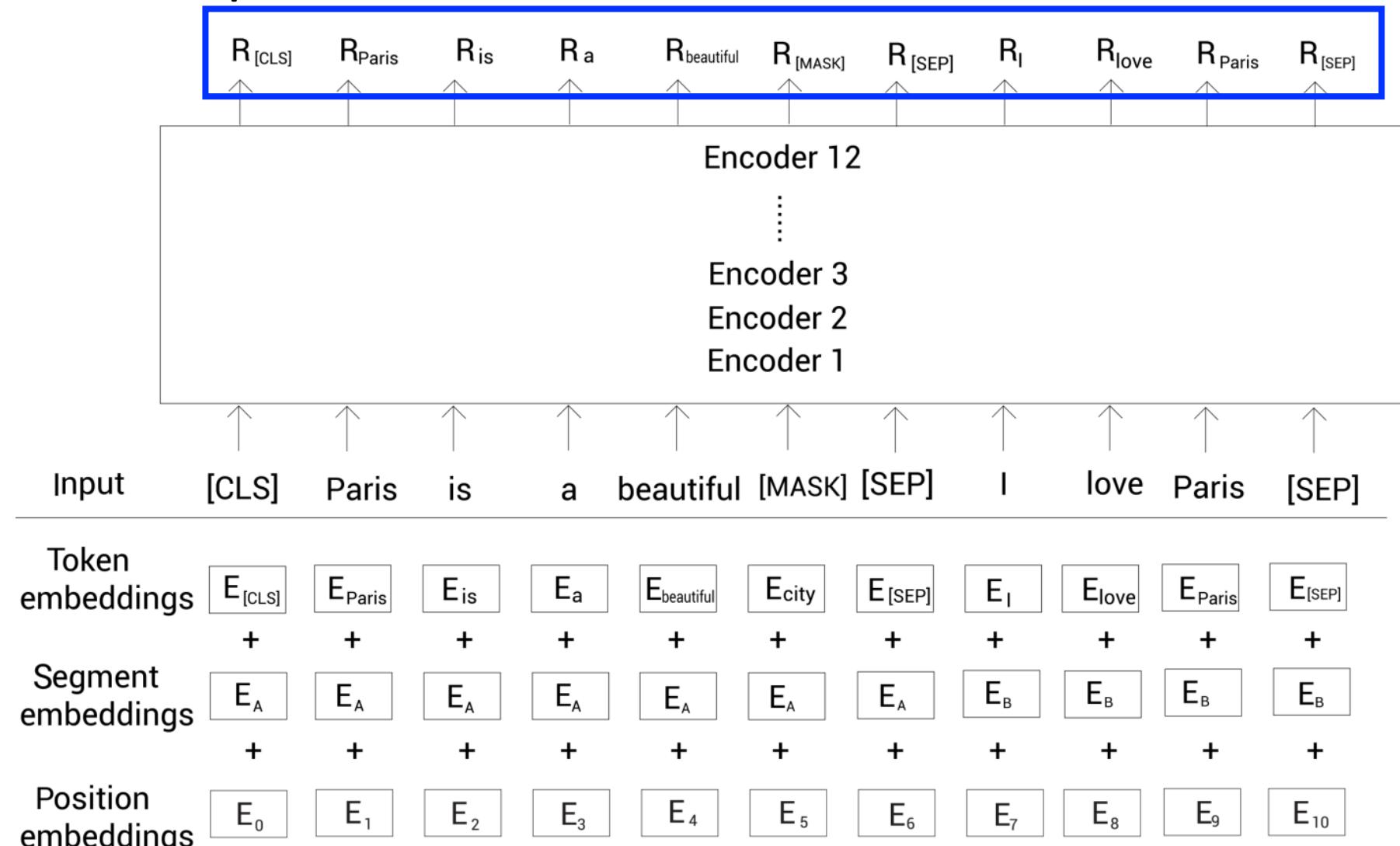
Architecture: Input



Input is addition of a segment embedding to the token and position embeddings (helps differentiate which tokens belong to which sentence)

Architecture: Output

New representation of each input that accounts for context

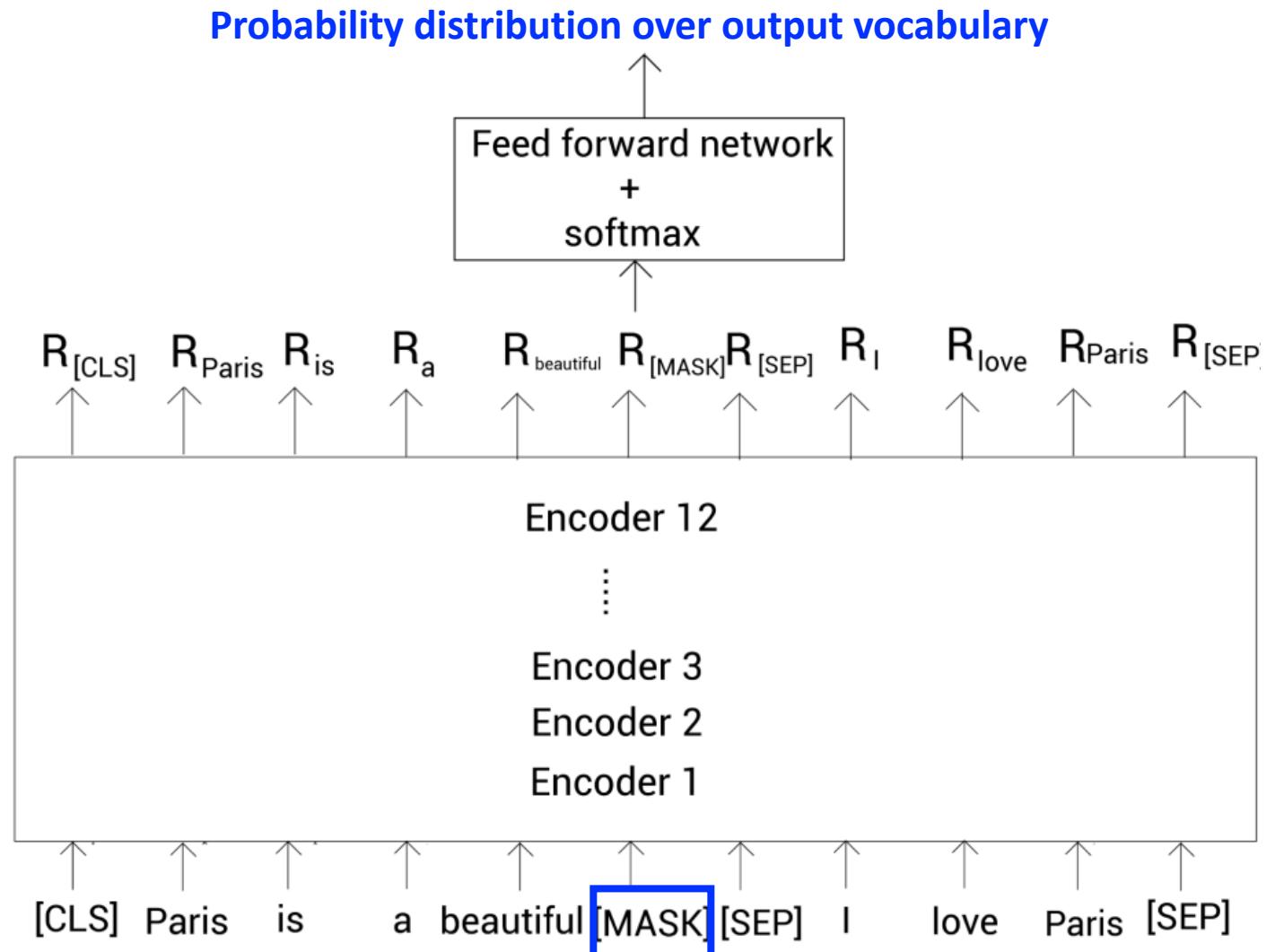


Architecture: Predicting Masked Token Task

15% of random tokens from sequence **masked**

- 80% use [MASK]
- 10% use a random token
- 10% are unchanged

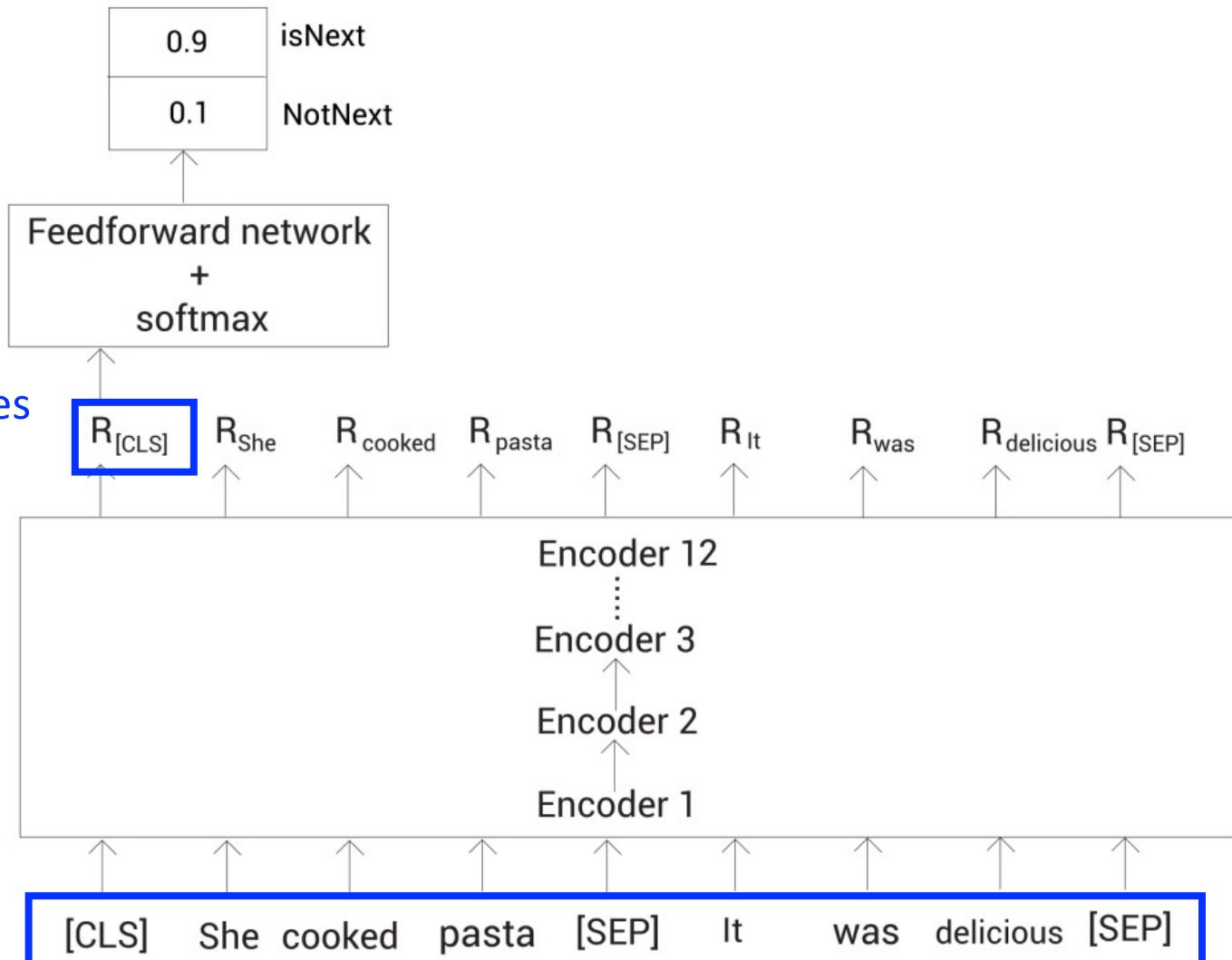
Multiple masking options encourage the model to pay attention to each token separately



Architecture: Predict if Next Sentence Task

Predict with token representation that aggregates representation of all tokens in both sentences

50% of 2nd sentences are the original next sentence and the rest are random

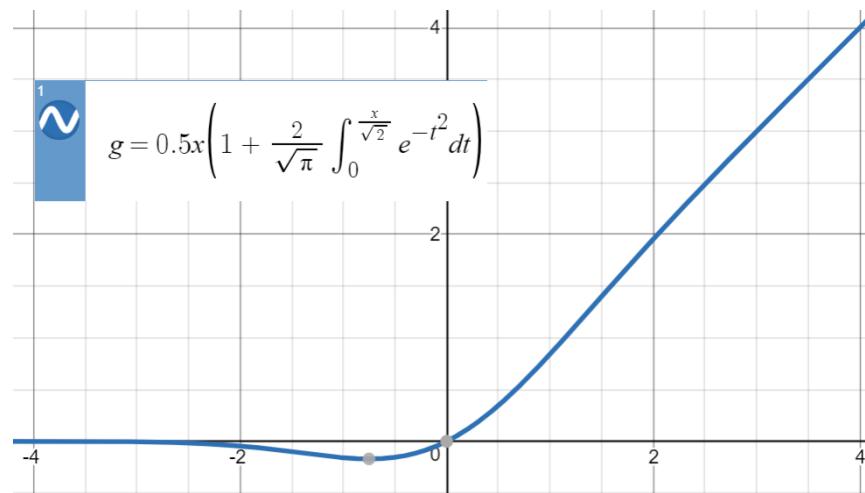


Training

- Dataset: 2,500M words in Wikipedia + 800M words in BooksCorpus used for GPT
- Optimizer: Adam
- Training loss: sums over losses from predicting masked words and if next sentence

Implementation Details: Mimics GPT

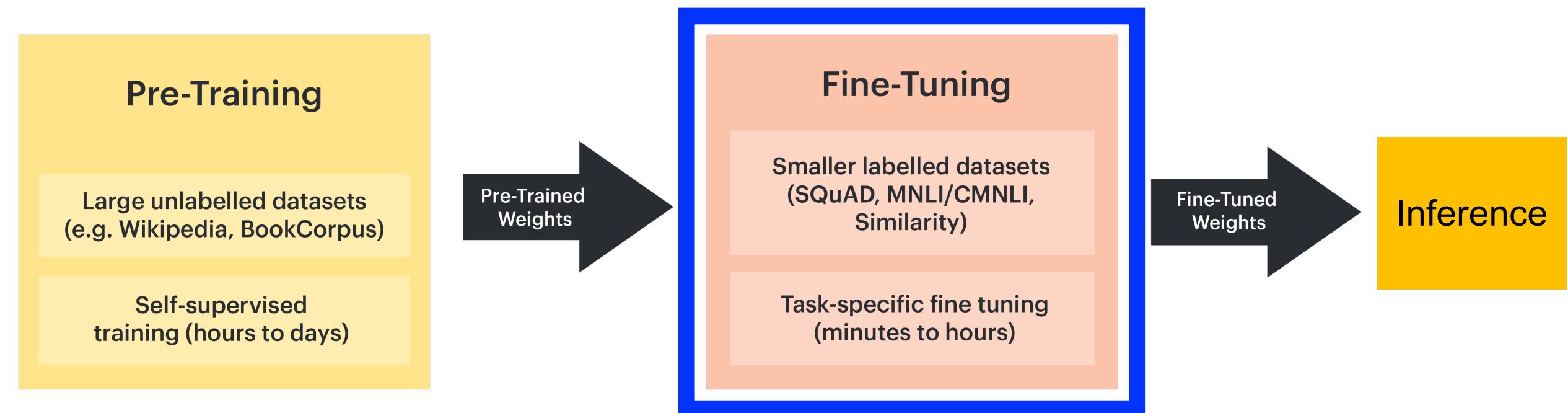
- Gaussian error linear unit (GELU) used as activation function in feedforward layers



[https://datascience.stackexchange.com/
questions/49522/what-is-gelu-activation](https://datascience.stackexchange.com/questions/49522/what-is-gelu-activation)

- Avoids out of vocabulary tokens by using subword tokenization, with a different variant called WordPiece Tokenization

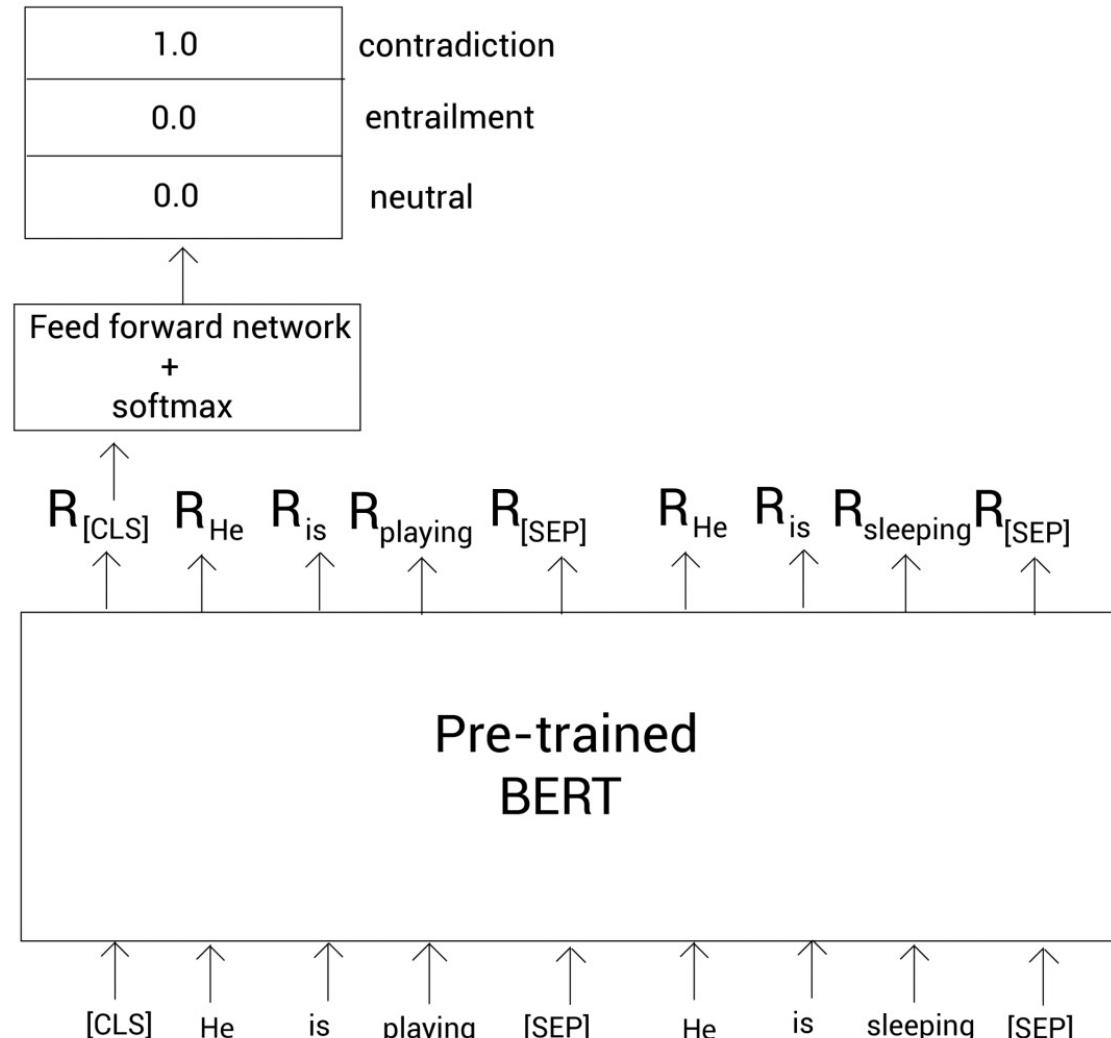
BERT: Bidirectional Encoder Representation from Transformer



Fine-Tuning for Natural Language Inference

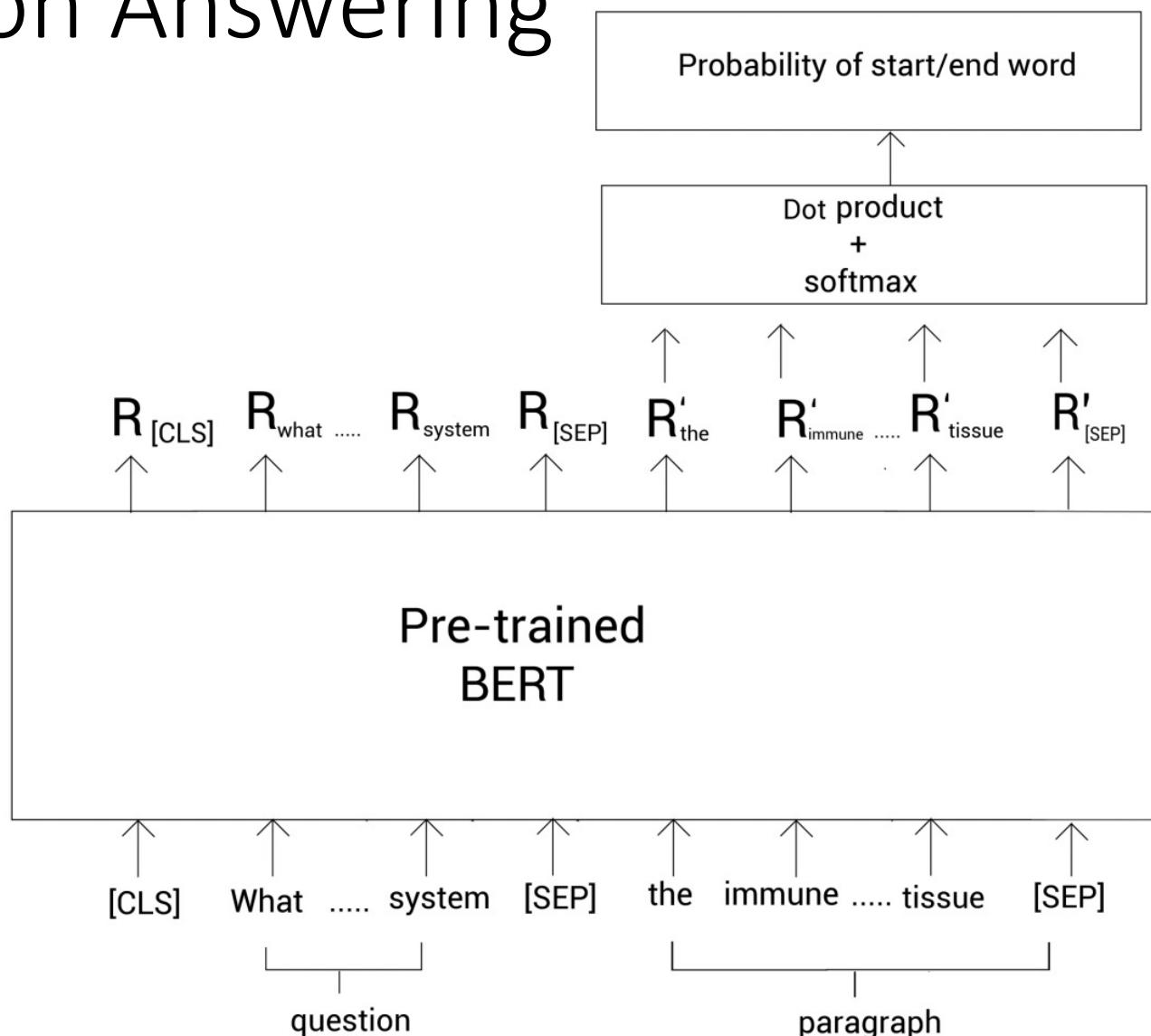
Premise	Hypothesis	Label
He is playing	He is sleeping	Contradiction
A soccer game with multiple males playing	Some men are playing sport	Entailment
An older and a younger man smiling	Two men are smiling at the dogs playing on the floor	Neutral

Fine-Tuning for Natural Language Inference



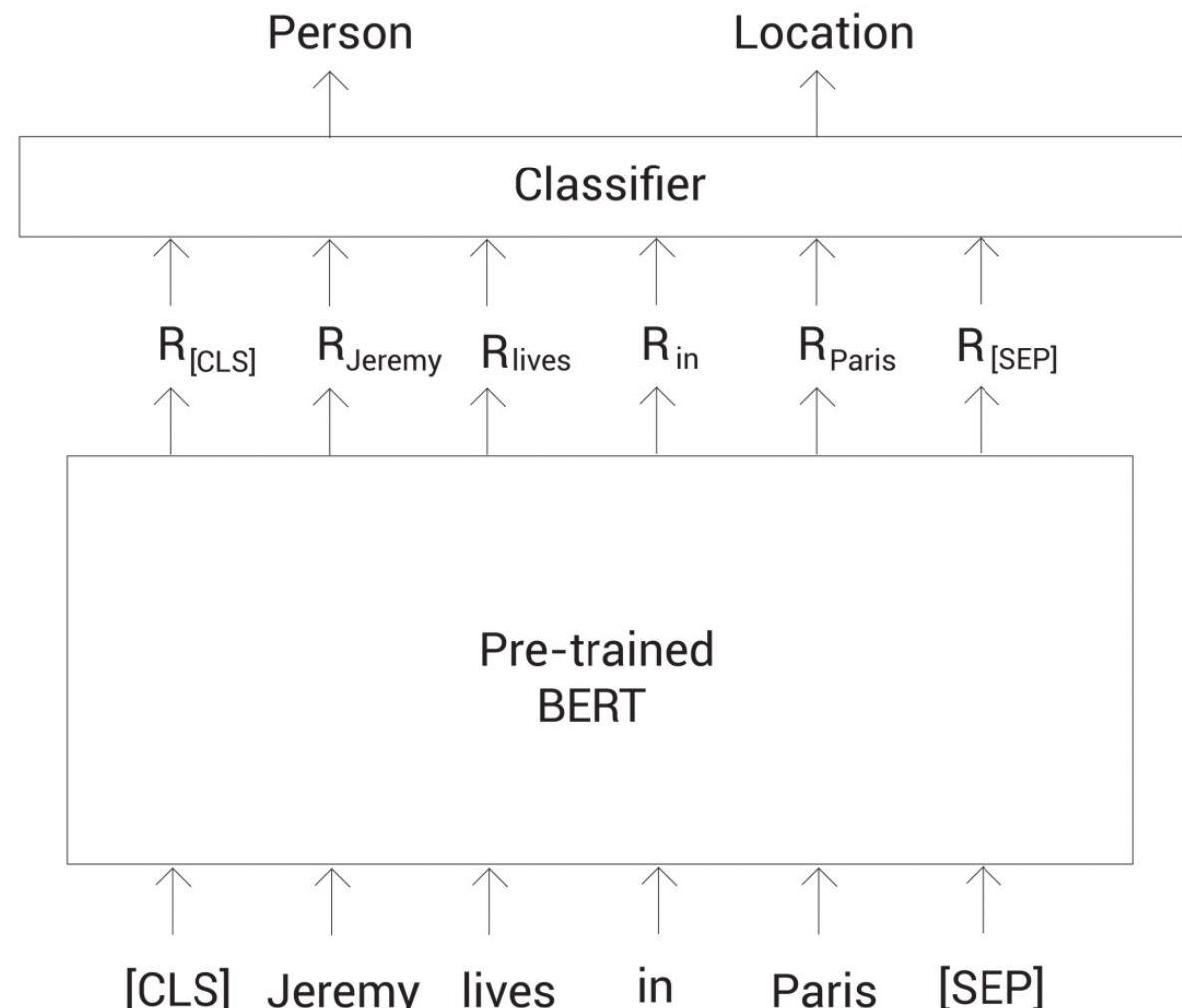
Fine-Tuning for Question Answering

To find indices of the start and end words in the paragraph, two vector representations are learned that lead to the approximate softmax output when computing the dot product with each token



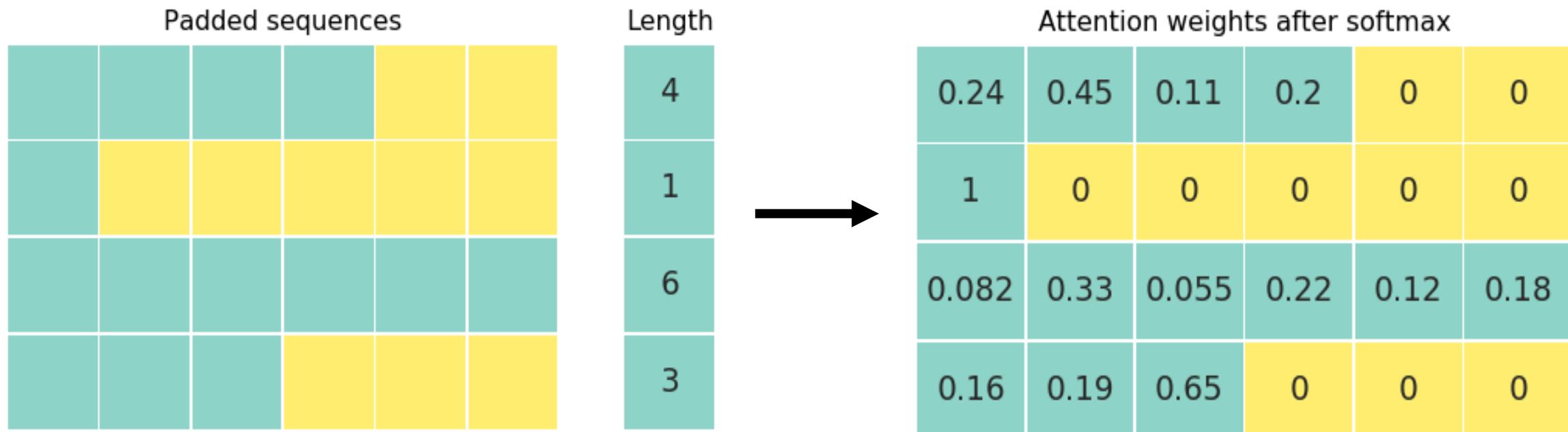
Fine-Tuning for Named Entity Recognition

Each token's new representation
is passed to a classifier



Implementation Detail

- Padding supports the use of variable input length
 - Uses attention vector of 1s and 0s, with the latter at indices of [PAD] tokens



Experimental Findings

Achieved the best performance on 11 NLP dataset challenges

Experimental Findings

Tasks	Dev Set				
	MNLI-m (Acc)	QNLI (Acc)	MRPC (Acc)	SST-2 (Acc)	SQuAD (F1)
BERT _{BASE}	84.4	88.4	86.7	92.7	88.5
No NSP	83.9	84.9	86.5	92.6	87.9
LTR & No NSP	82.1	84.3	77.5	92.1	77.8
+ BiLSTM	82.1	84.1	75.7	91.6	84.9

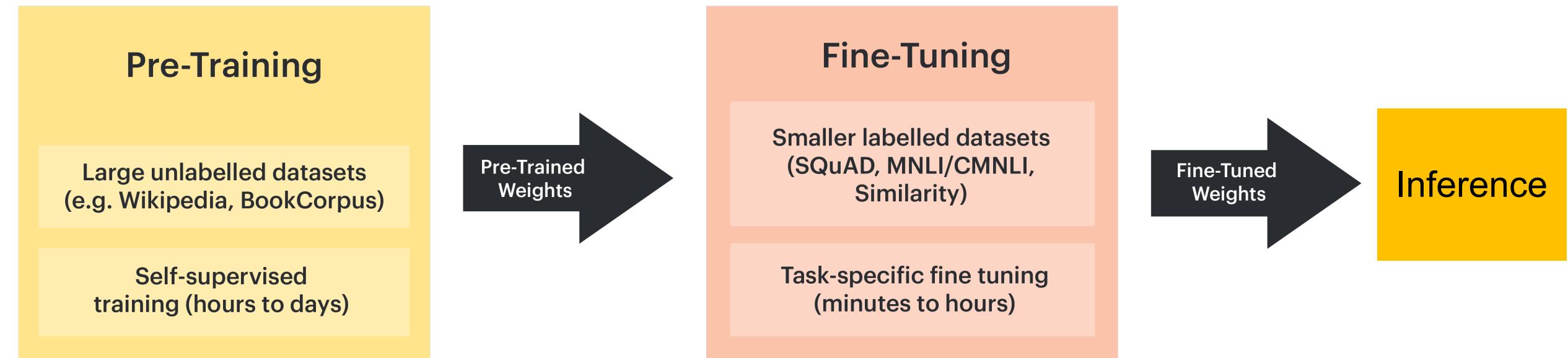
Next sentence prediction (NSP) supports slight improvements

Experimental Findings

Tasks	Dev Set				
	MNLI-m (Acc)	QNLI (Acc)	MRPC (Acc)	SST-2 (Acc)	SQuAD (F1)
BERT _{BASE}	84.4	88.4	86.7	92.7	88.5
No NSP	83.9	84.9	86.5	92.6	87.9
LTR & No NSP	82.1	84.3	77.5	92.1	77.8
+ BiLSTM	82.1	84.1	75.7	91.6	84.9

We observe a performance boost when using bidirectional pretraining instead of unidirectional pretraining (LTR)

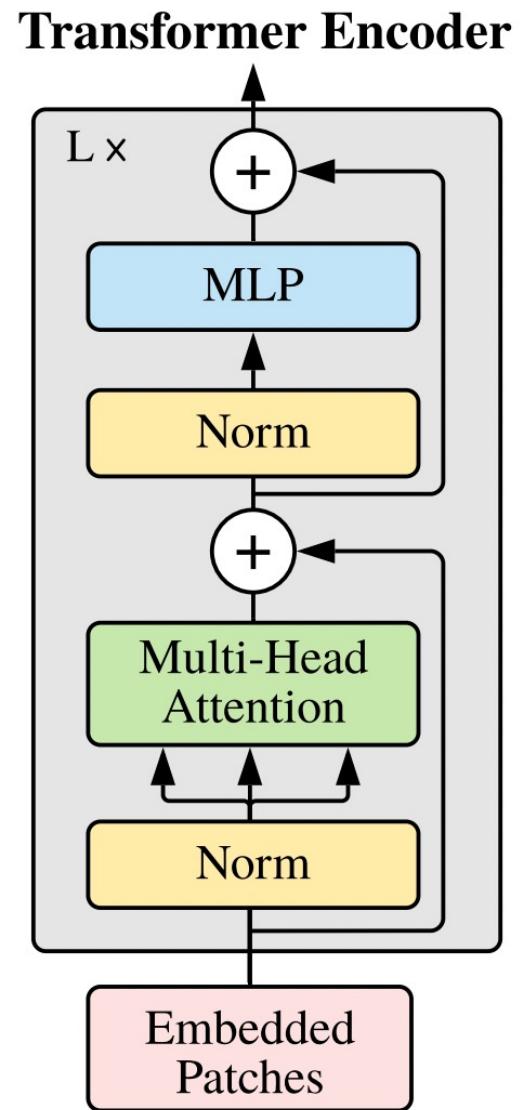
BERT: Bidirectional Encoder Representation from Transformer



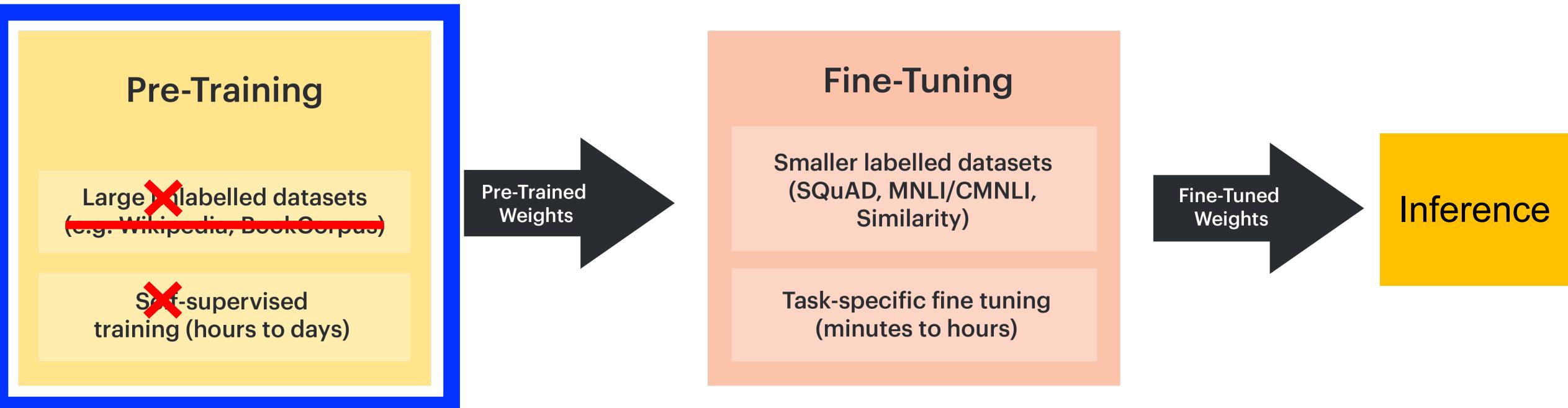
Today's Topics

- Explosion of transformers
- GPT
- BERT
- ViT
- Limitations of transformer models

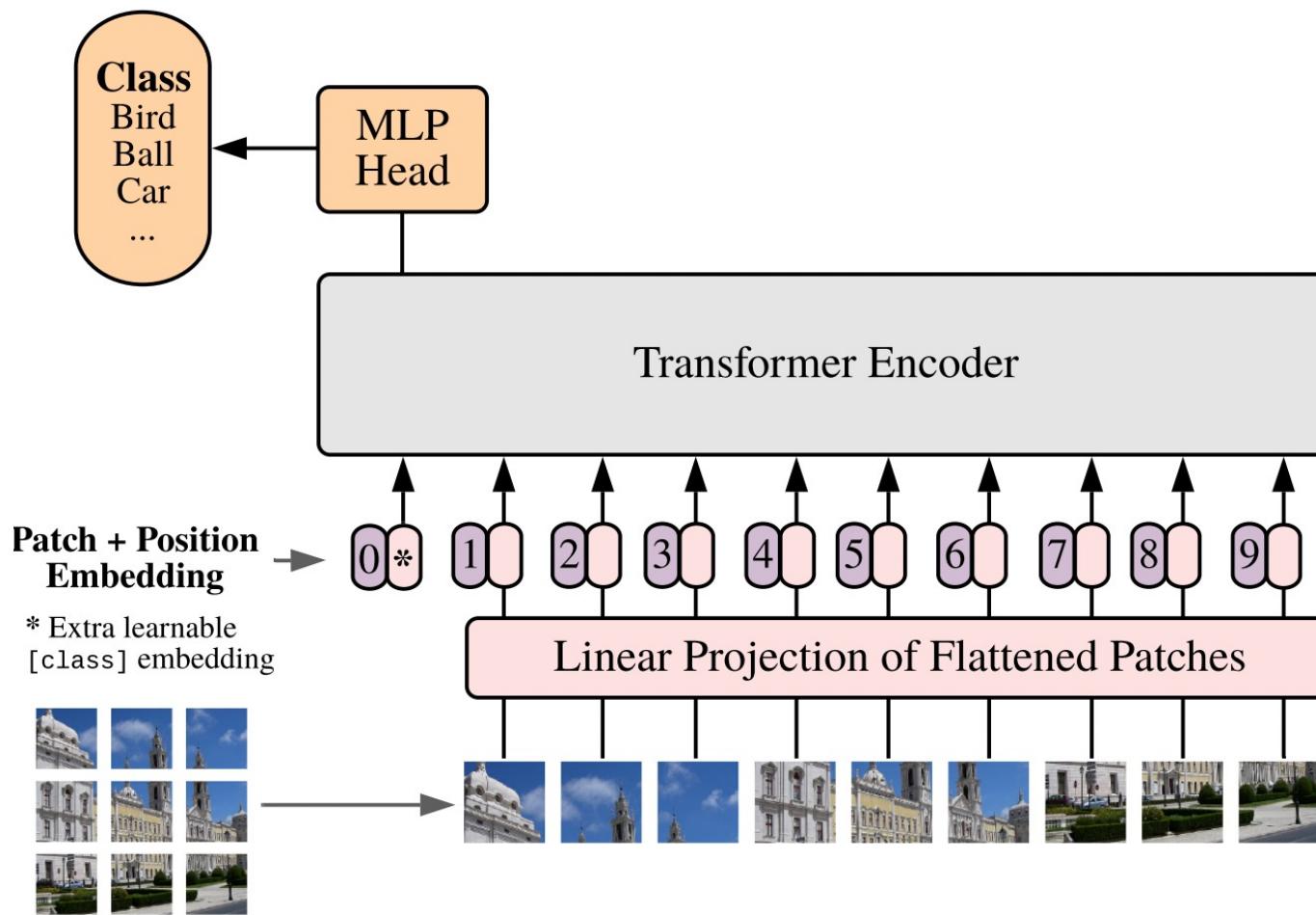
Motivation: Transformers for Image Classification (Repurpose BERT)



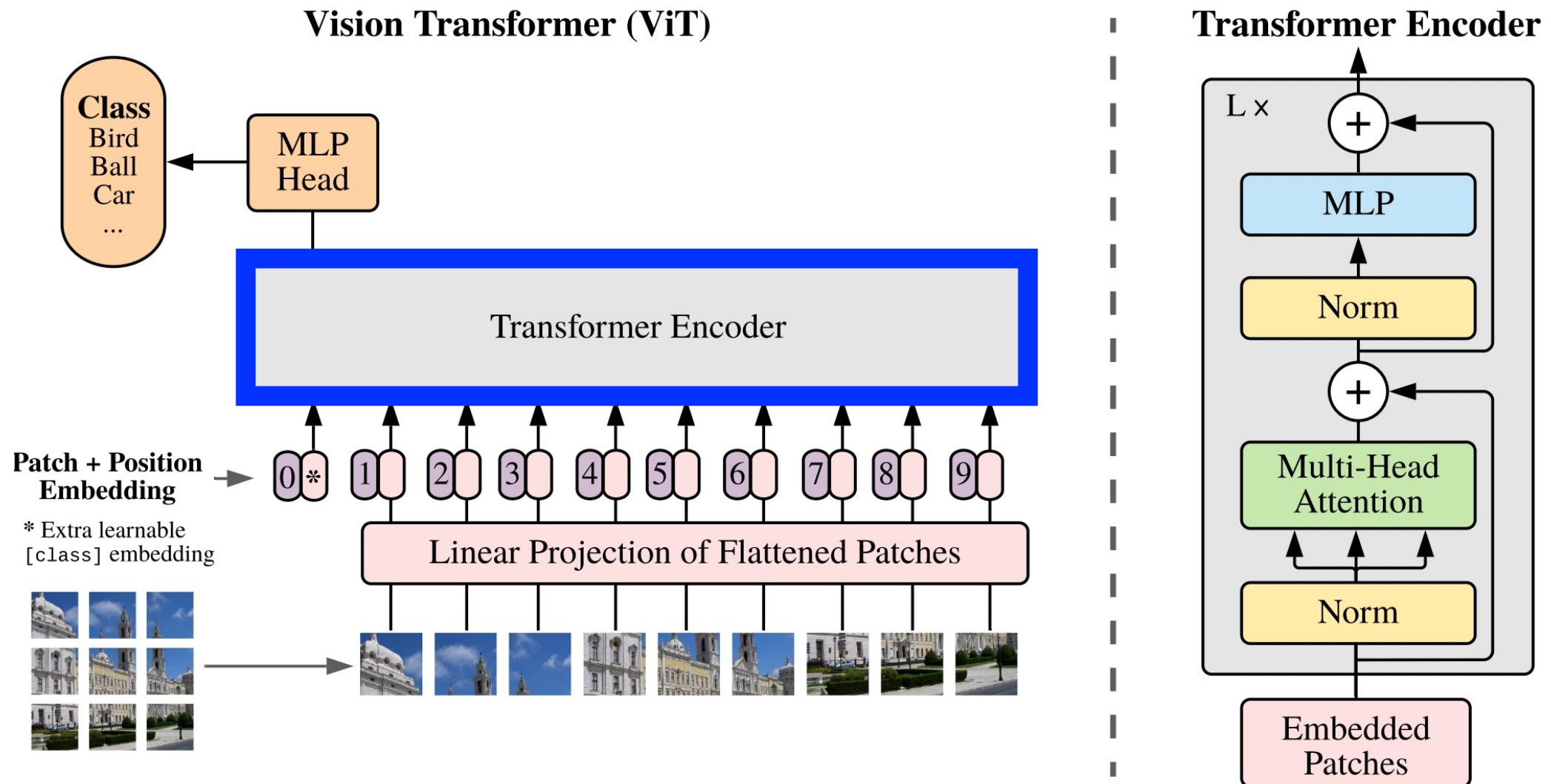
ViT: Vision Transformer



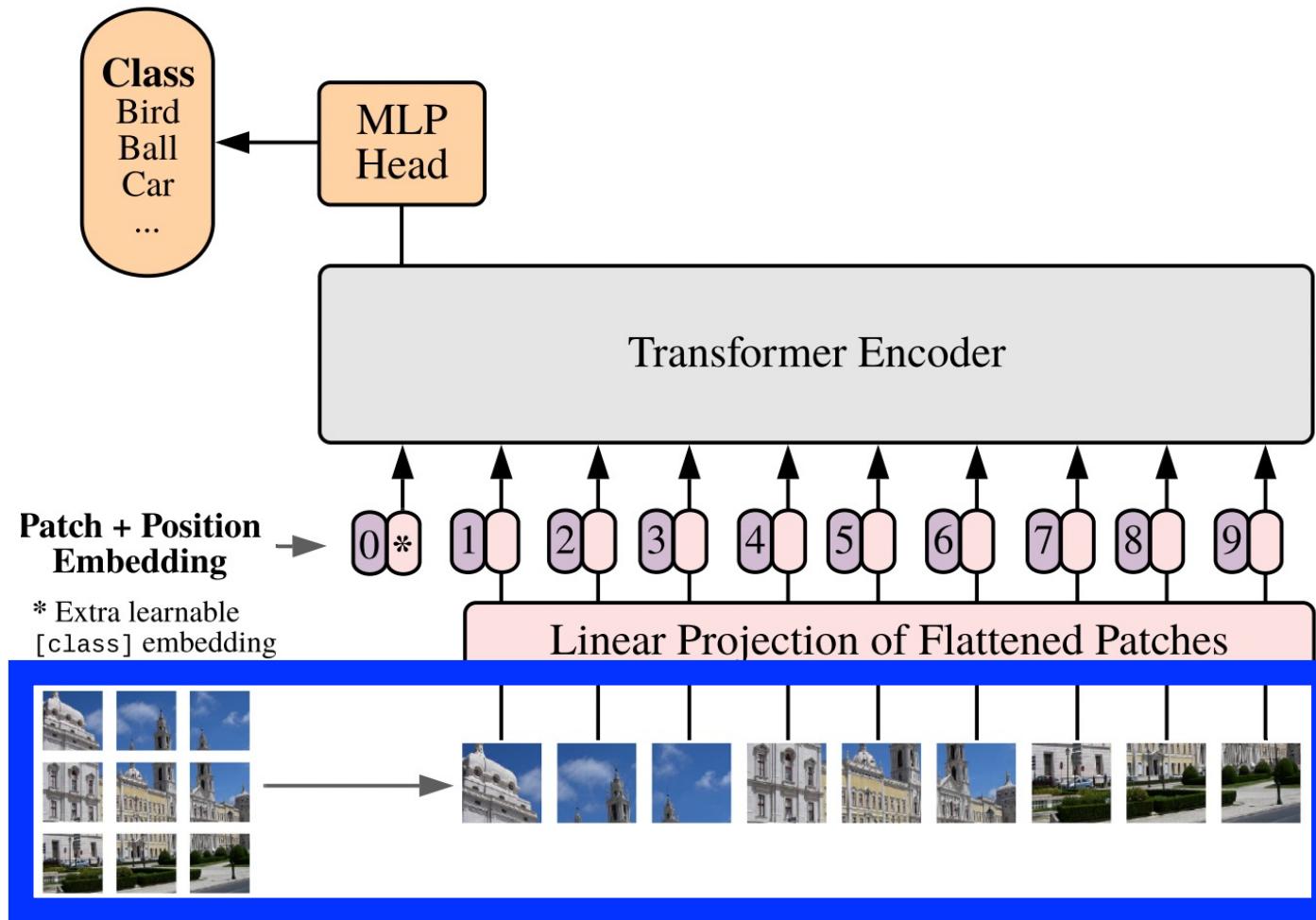
Architecture



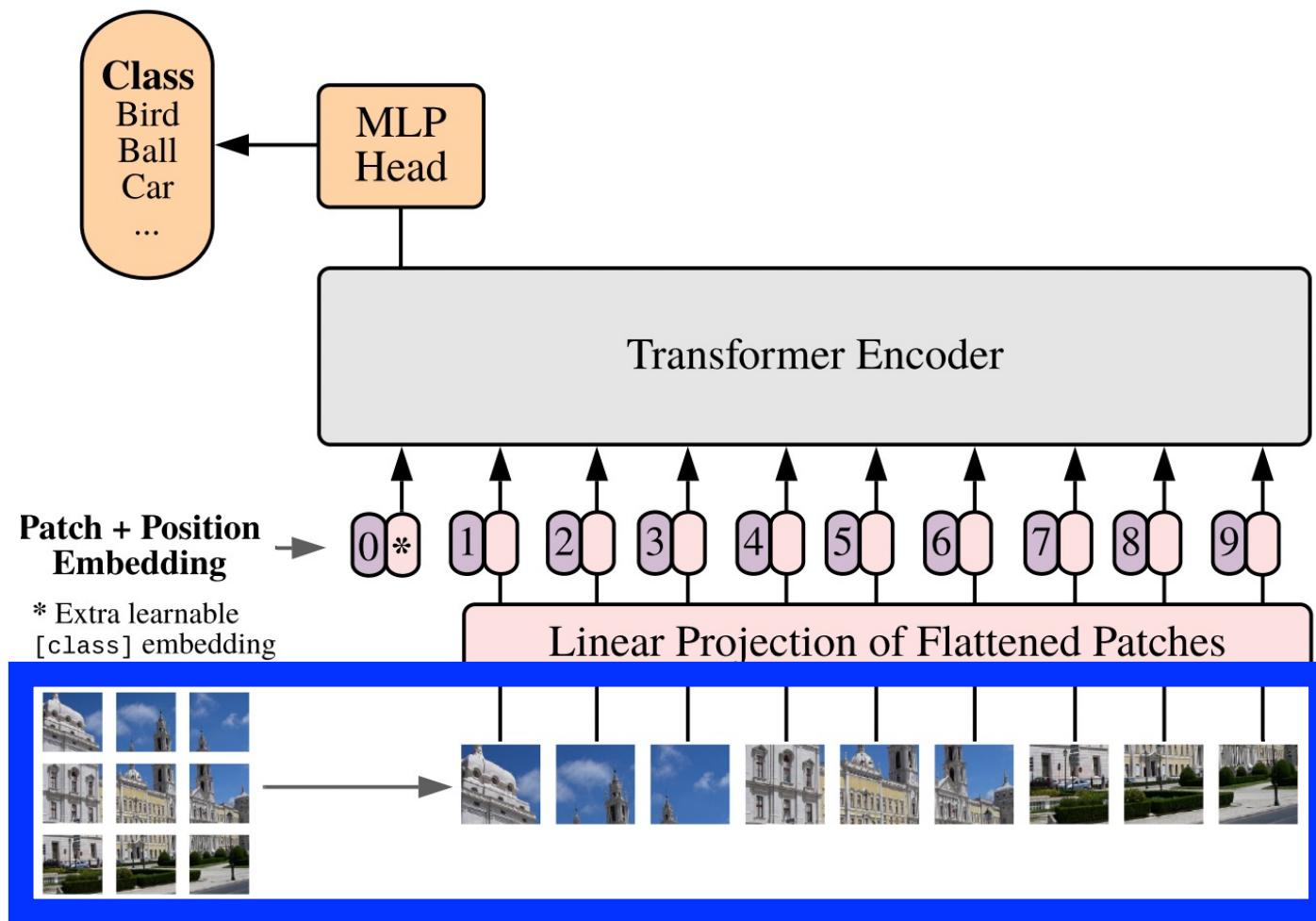
Architecture: BERT



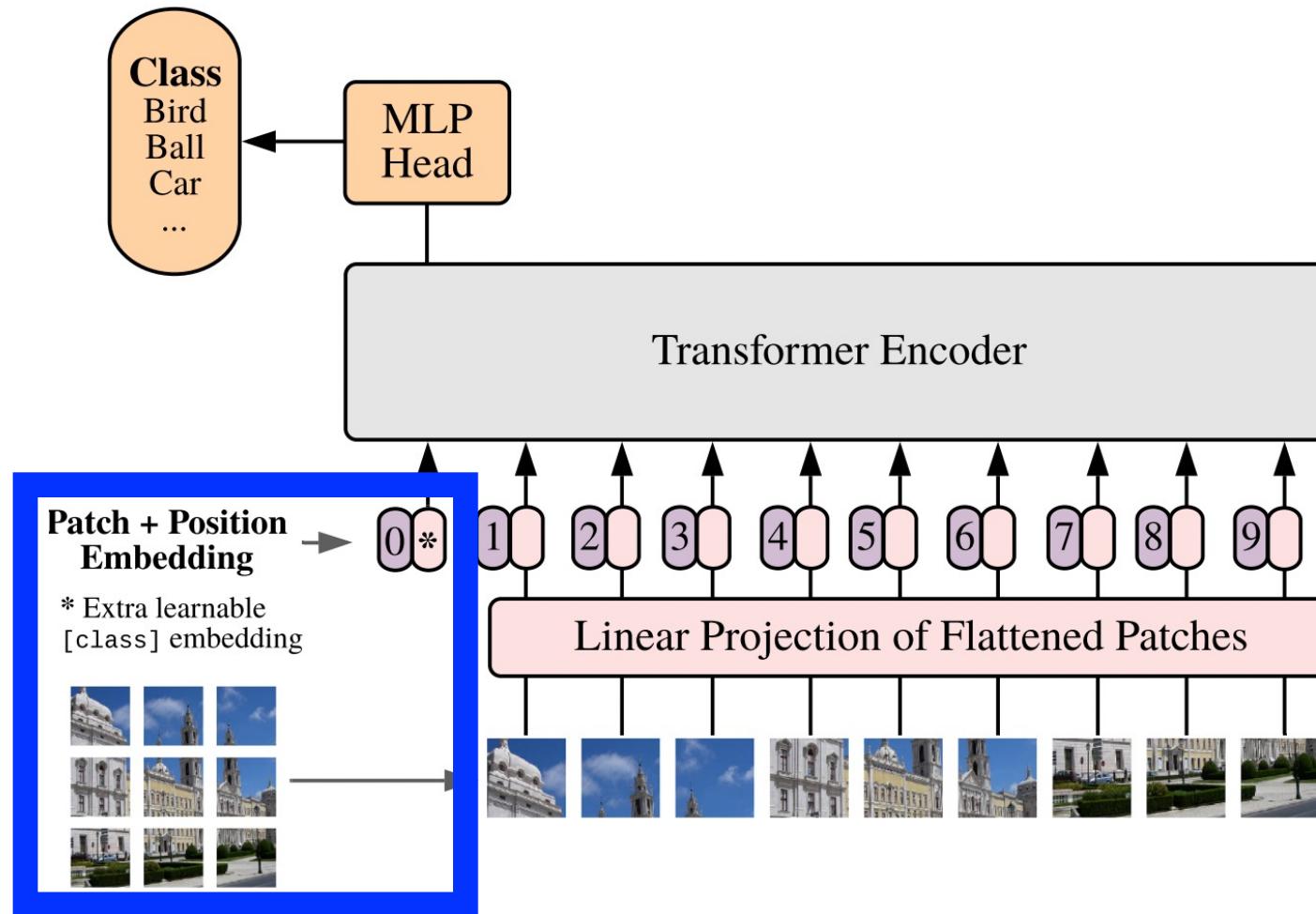
Architecture: Input



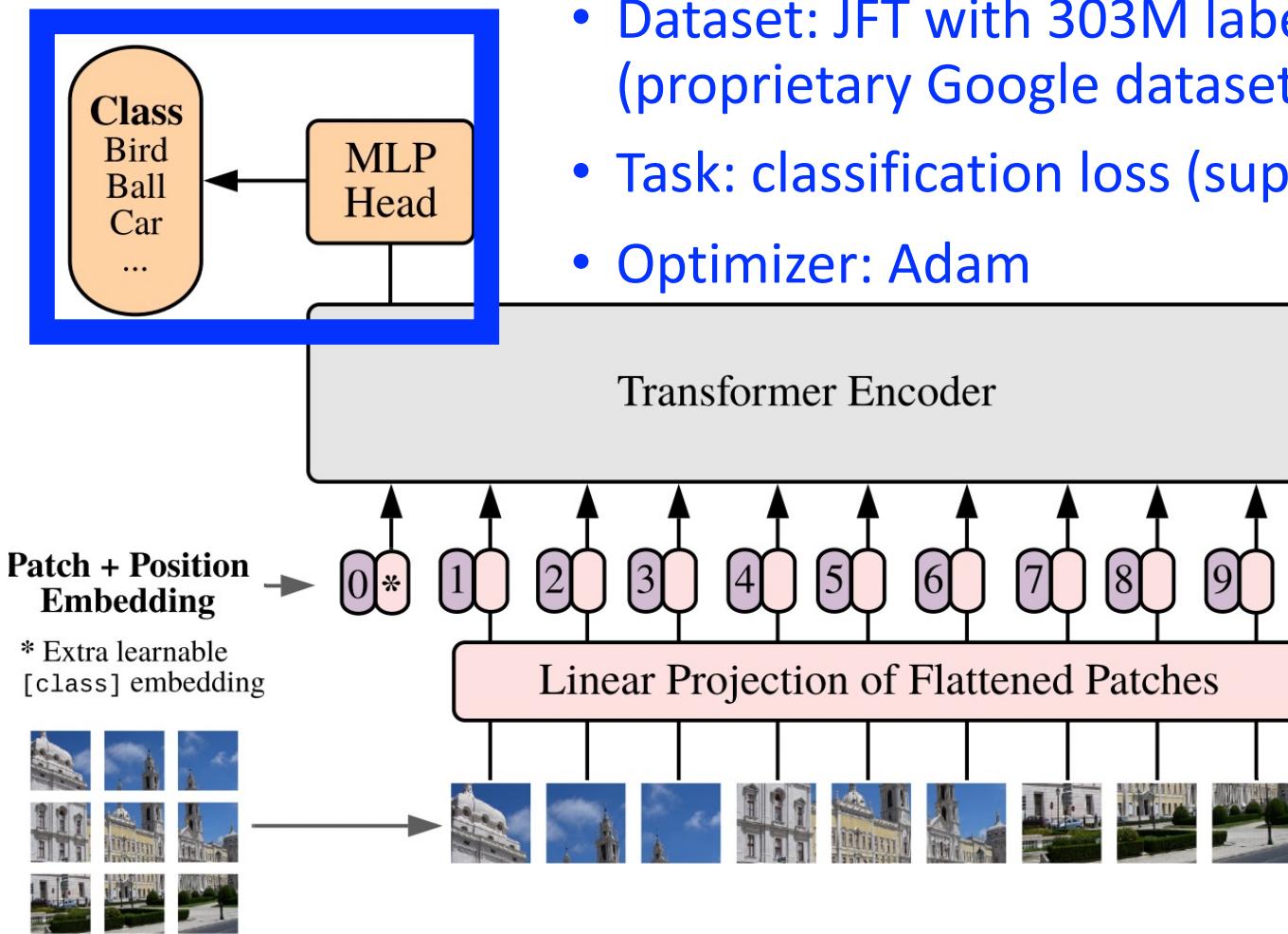
Architecture: Input



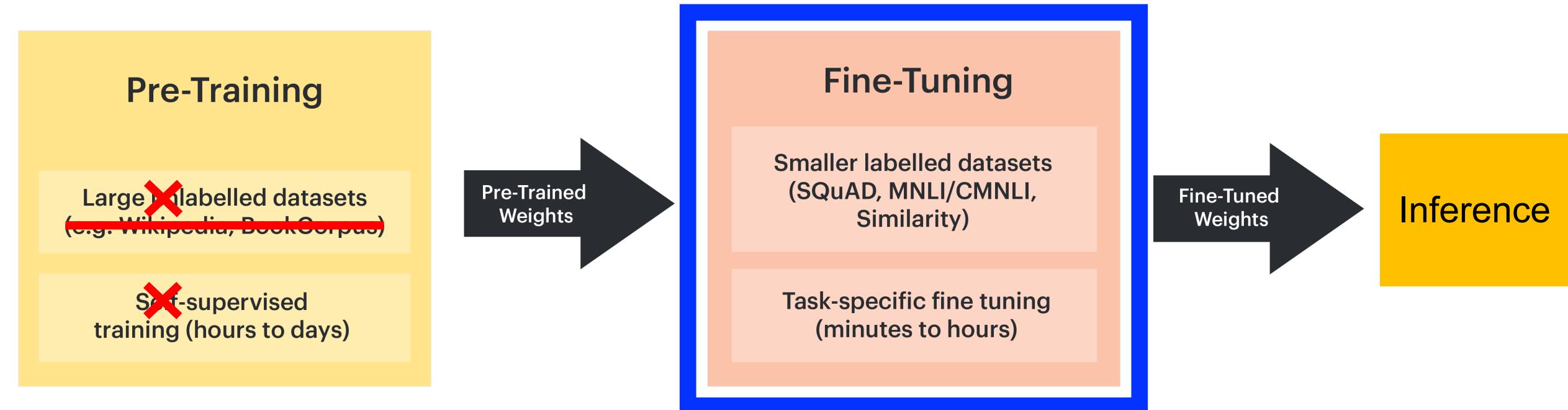
Architecture: Input



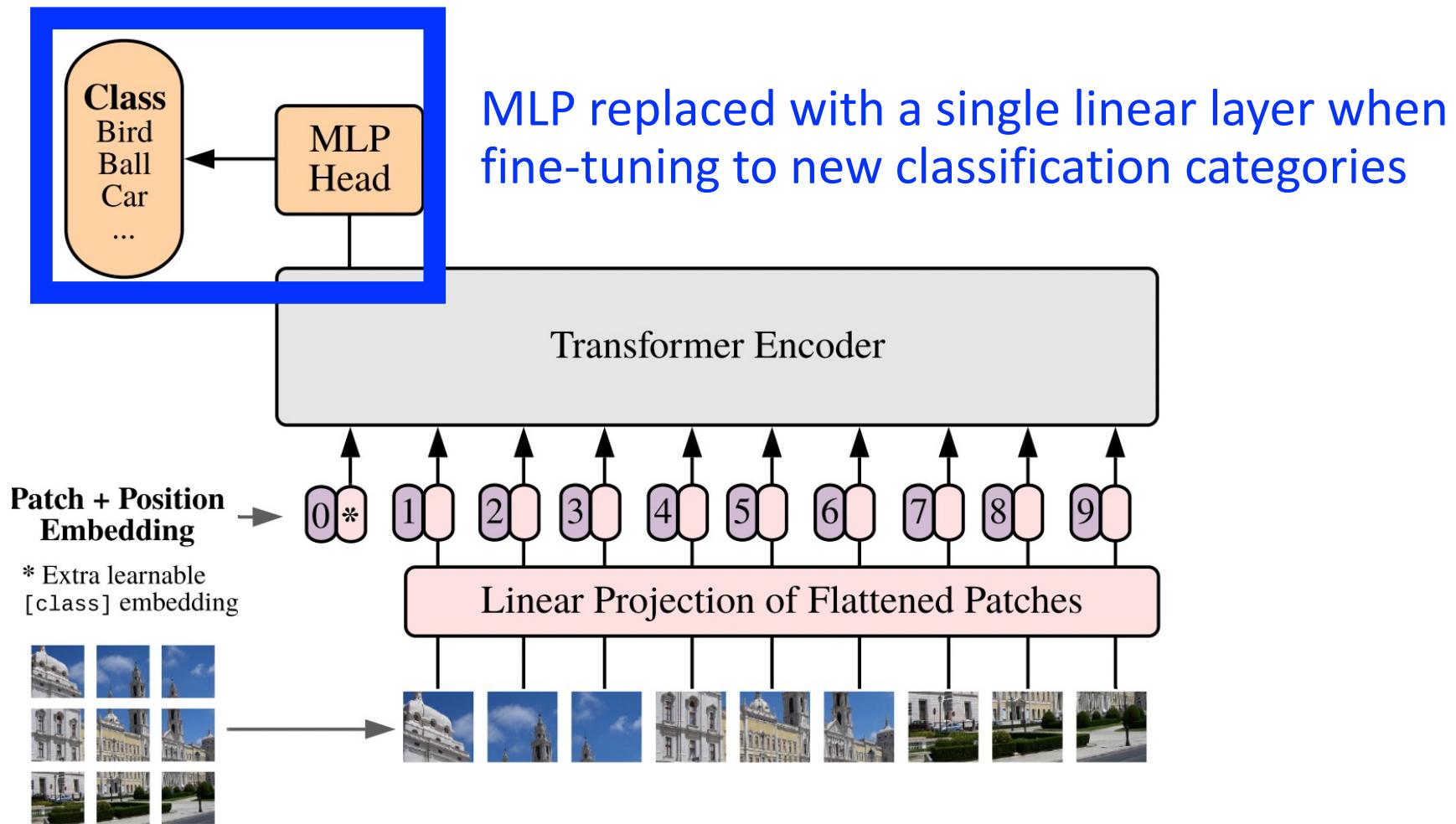
Architecture: Training



ViT: Vision Transformer



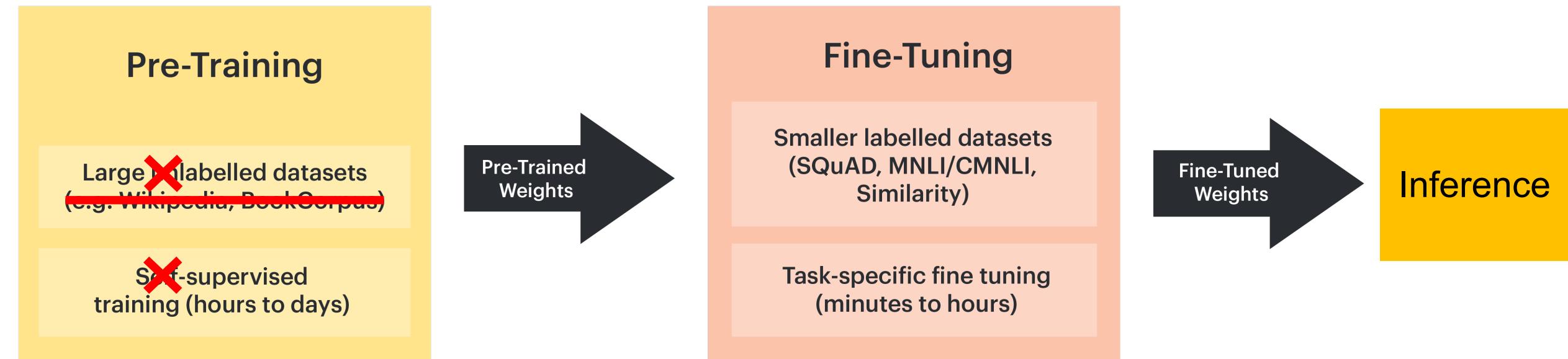
Fine-Tuning for Other Image Classification Tasks



Experimental Findings

Achieved strong results on five image classification datasets

ViT: Vision Transformer



Today's Topics

- Explosion of transformers
- GPT
- BERT
- ViT
- Limitations of transformer models

On the Dangers of Stochastic Parrots: Can Language Models Be Too Big?

Emily M. Bender*

ebender@uw.edu

University of Washington

Seattle, WA, USA

Angelina McMillan-Major

aymm@uw.edu

University of Washington

Seattle, WA, USA

Timnit Gebru*

timnit@blackinai.org

Black in AI

Palo Alto, CA, USA

Shmargaret Shmitchell

shmargaret.shmitchell@gmail.com

The Aether

Context: original Transformer paper
and BERT published by Google



<https://www.wired.com/story/google-timnit-gebru-ai-what-really-happened/>

Transformers' Financial Cost; e.g., To Train BERT, How Much Do You Think it Costed in US Dollars?

THE COST OF TRAINING NLP MODELS A CONCISE OVERVIEW

Or Sharir
AI21 Labs
ors@ai21.com

Barak Peleg
AI21 Labs
barakp@ai21.com

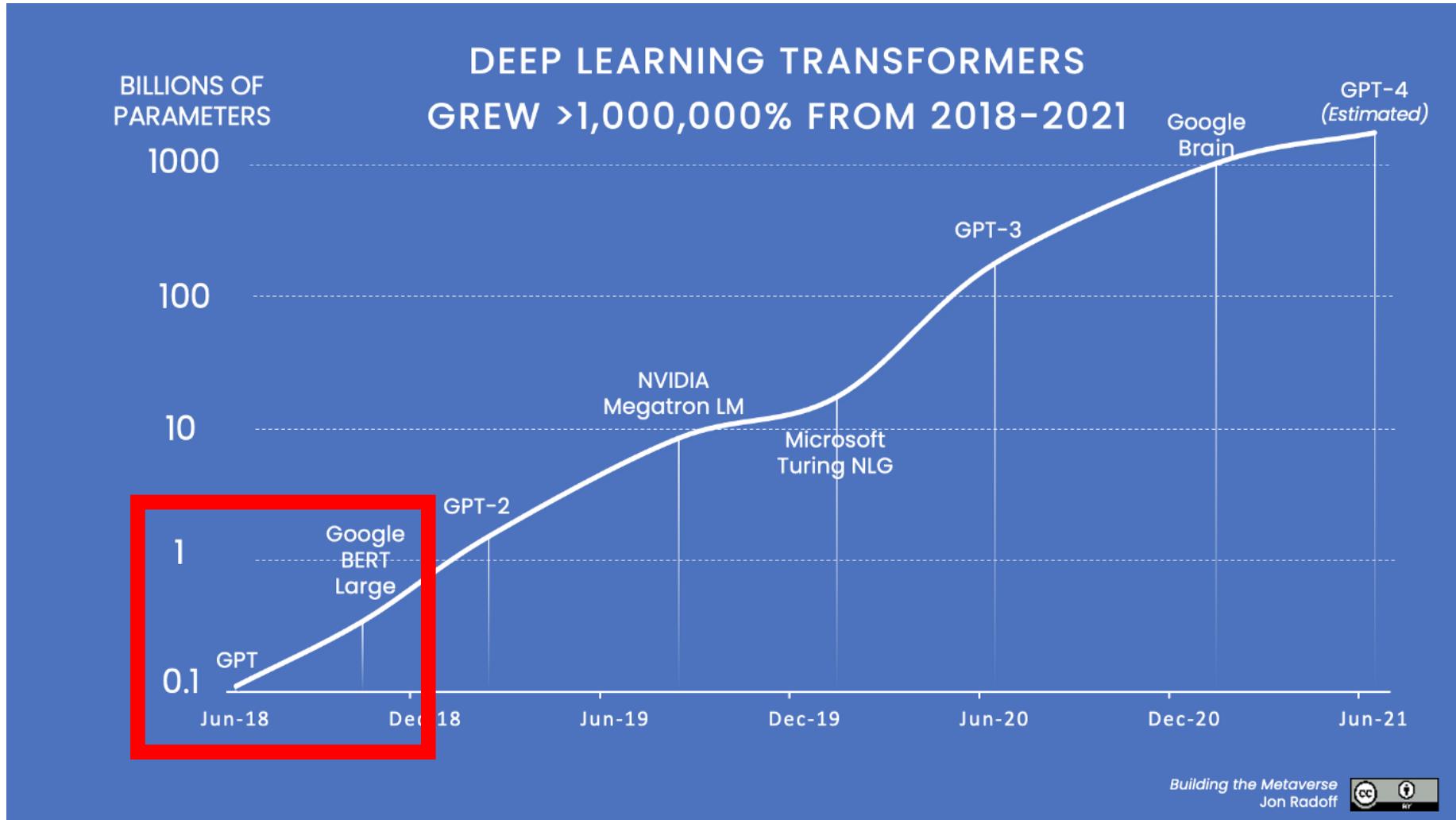
Yoav Shoham
AI21 Labs
yoavs@ai21.com

- \$2.5k - \$50k (110 million parameter model)
- \$10k - \$200k (340 million parameter model)
- \$80k - \$1.6m (1.5 billion parameter model)

Transformers' Environmental Cost

- Does training a BERT base model require as much energy as:
 - Microwaving food for 7 minutes
 - Heating a home for a day
 - Driving 100 miles
 - A trans-American flight

Transformers: Huge and Growing in Size



Transformers' Societal Cost; e.g., BERT

- Influence of training data: 2,500M words in Wikipedia + 800M words in BooksCorpus
 - Who does and who does not contribute to such data repositories?
 - e.g., “recent surveys of Wikipedians find that only 8.8–15% are women or girls”
 - e.g., “Internet access itself is not evenly distributed, resulting in Internet data overrepresenting younger users and those from developed countries”
 - What kind of biases might be found in such data repositories?
 - e.g., “BERT associates phrases referencing persons with disabilities with more negative sentiment words, and that gun violence, homelessness, and drug addiction are overrepresented in texts discussing mental illness”
 - Given that “unsupervised pre-training is an integral part of many language understanding systems” (BERT paper: Devlin et al. arXiv 2018), how do we do this responsibly?

Today's Topics

- Explosion of transformers
- GPT
- BERT
- ViT
- Limitations of transformer models

The End