Popular Transformers

Danna Gurari
University of Colorado Boulder
Spring 2024

https://dannagurari.colorado.edu/course/neural-networks-and-deep-learning-spring-2024/

Review

- Last lecture:
  - Transformer overview
  - Self-attention
  - Multi-head attention
  - Common transformer ingredients
  - Pioneering transformer: machine translation
- Assignments (Canvas):
  - Lab assignment 3 due earlier today
  - Problem set 4 (last one!) due in one week
  - Final project after spring break (find partner, including at 5-6pm zoom meeting)
- Questions?

Today’s Topics

- Explosion of transformers
- GPT
- BERT
- ViT
- Programming tutorial

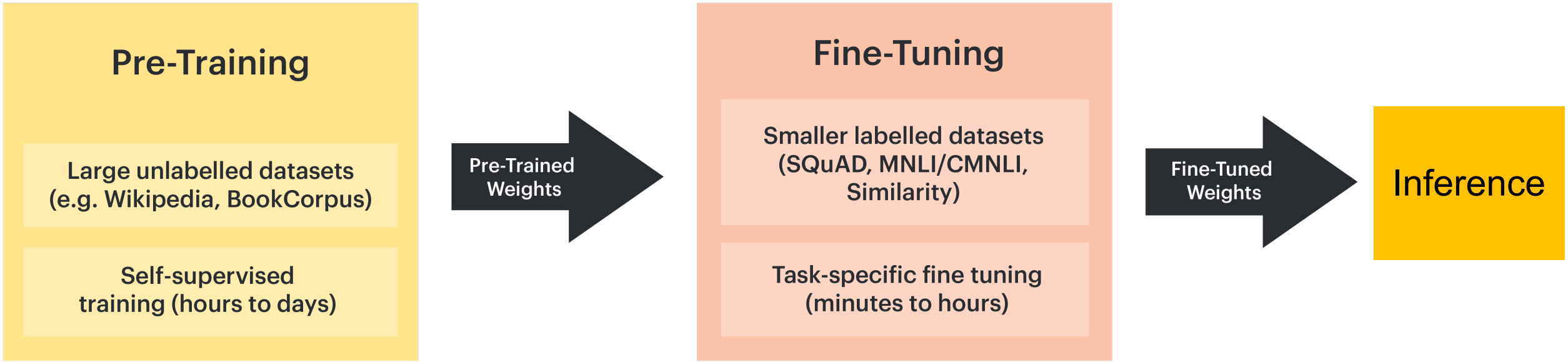
Today’s Topics

- Explosion of transformers
- GPT
- BERT
- ViT
- Programming tutorial

Explosion of Transformers in Society

https://www.pinterest.com/pin/521784306804400819/

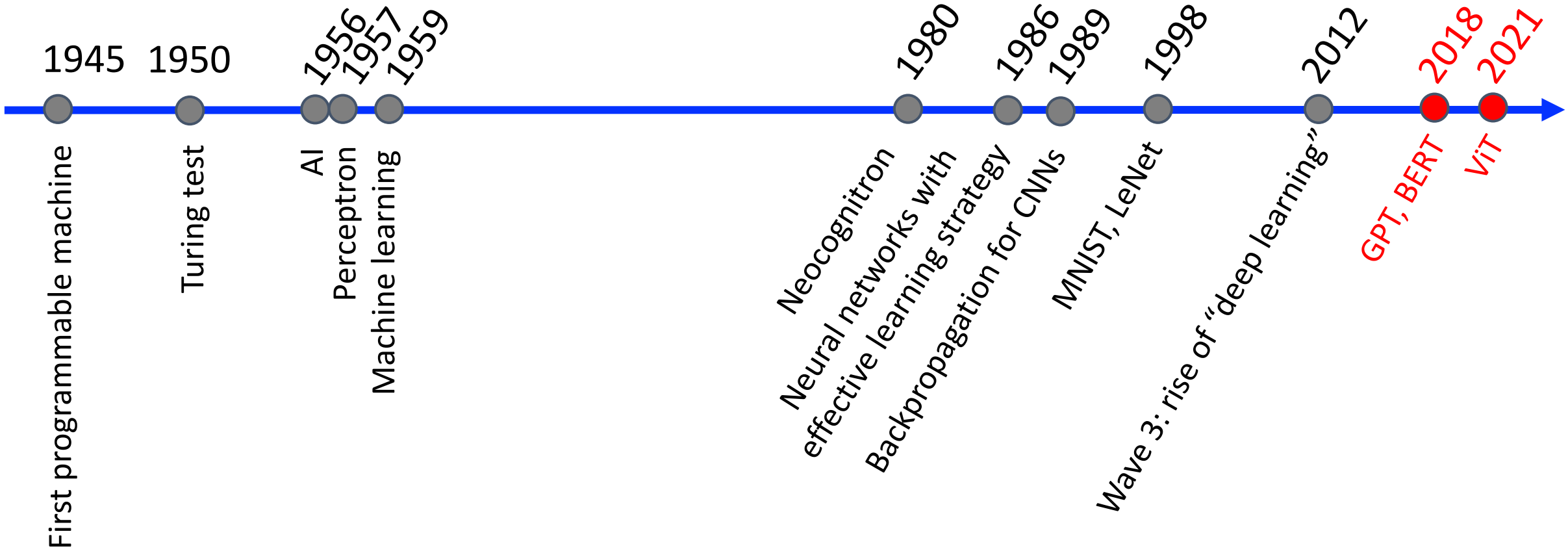
Today's Focus: Methods that Perform Pretraining and then Fine-tuning



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")

https://docs.graphcore.ai/projects/bert-training/en/latest/bert.html

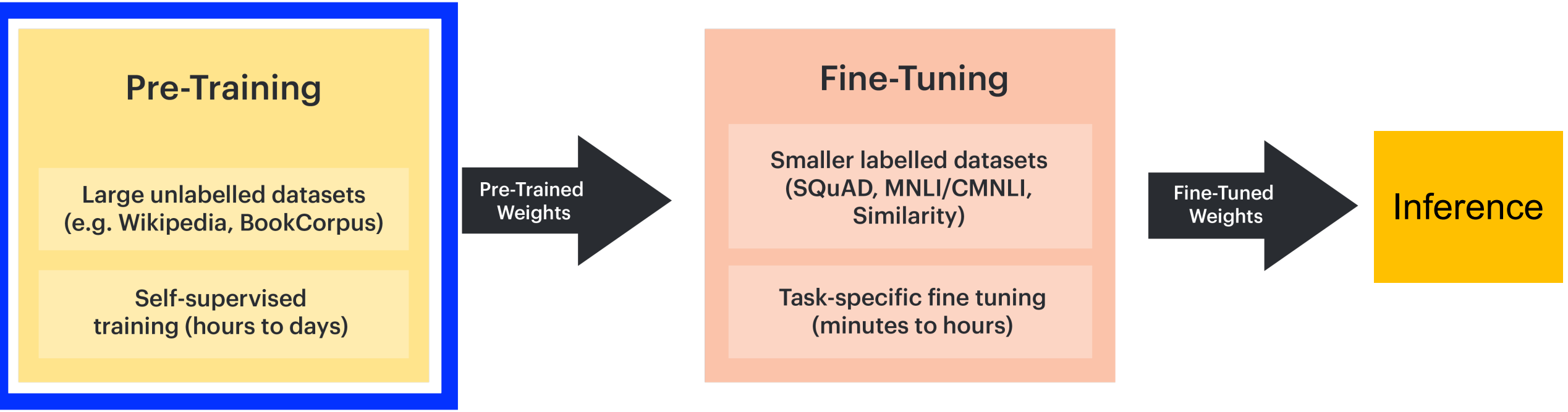
Today's Focus: Historical Context



Today’s Topics

- Explosion of transformers
- GPT
- BERT
- ViT
- Programming tutorial

GPT: Generative Pre-Training



https://docs.graphcore.ai/projects/bert-training/en/latest/bert.html

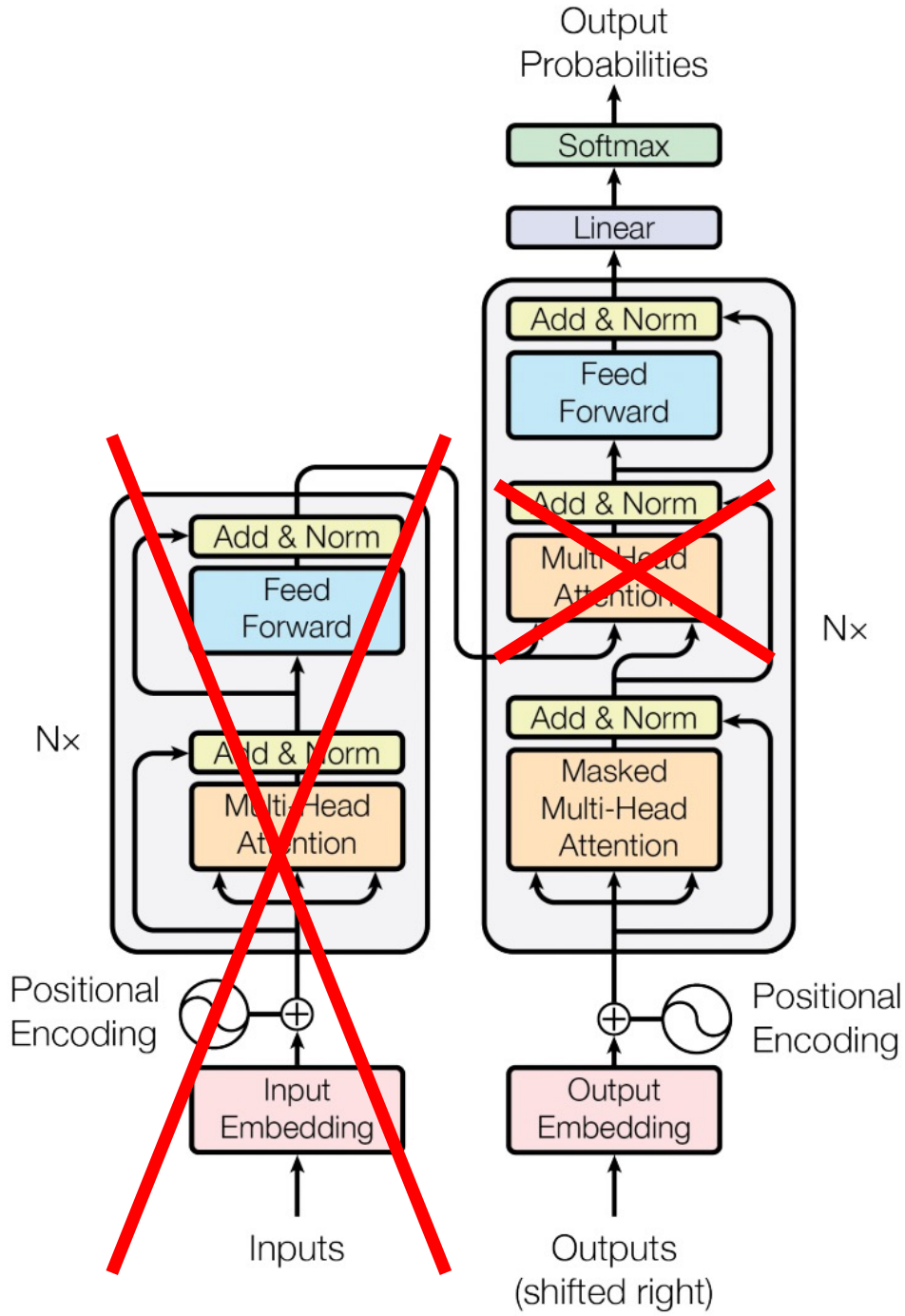
Task: Predict Next Word Given Previous Ones

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 _____

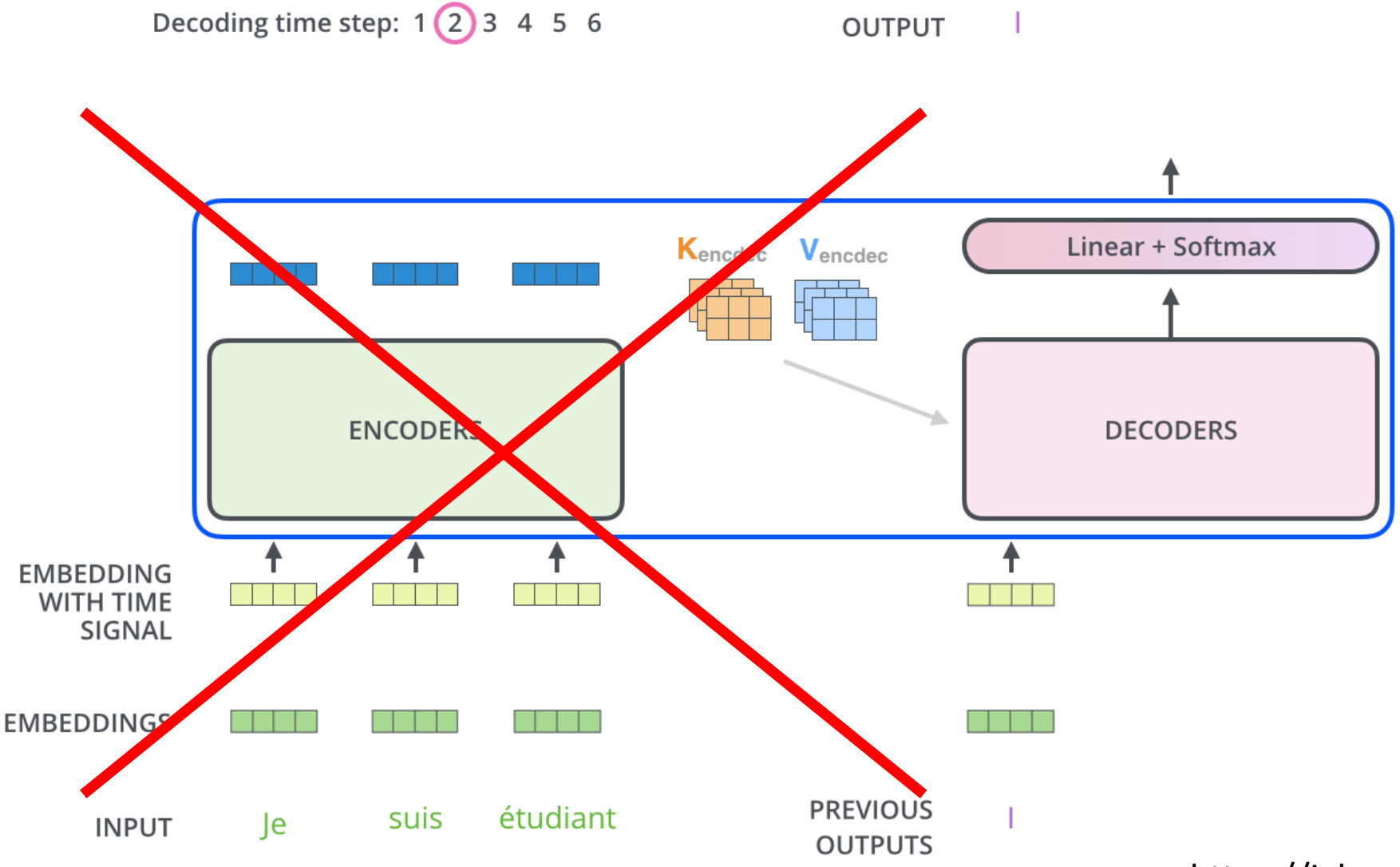
Architecture: Decoder from Pioneering Transformer

Removes encoder and corresponding decoder sub-layer thereby ignoring translation of an input sentence!



Vaswani et al. Attention Is All You Need. Neurips 2017.

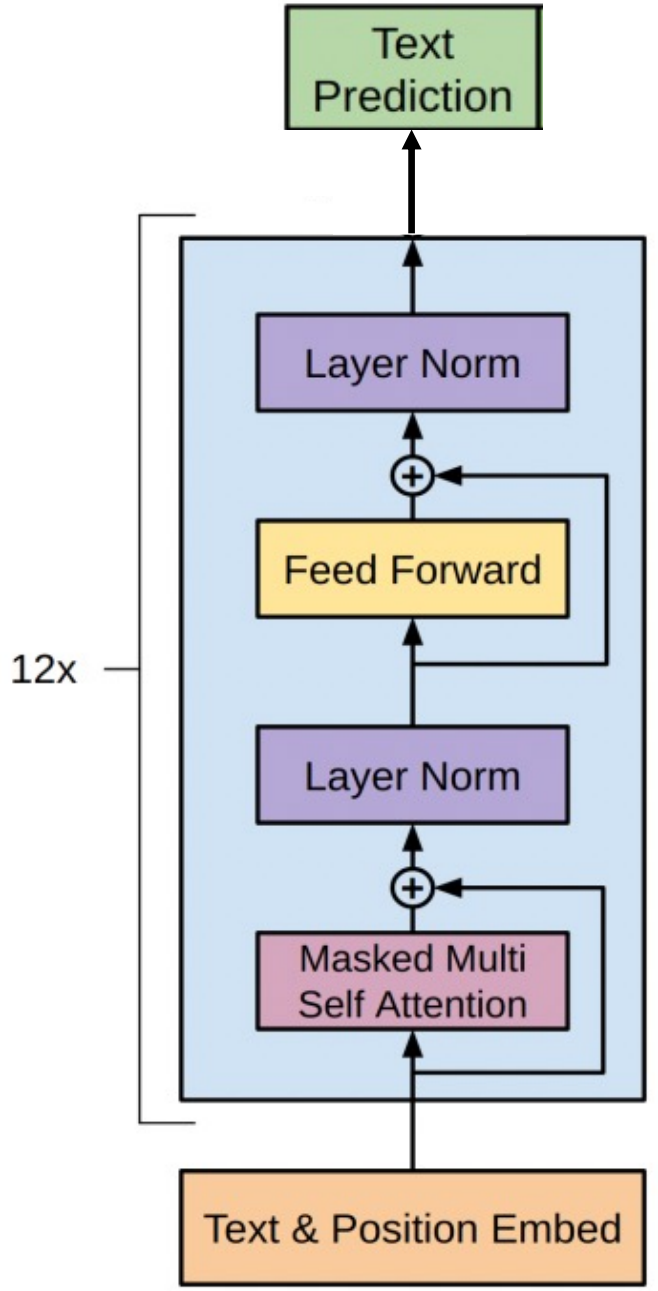
Architecture: Autoregressive Model



"Autoregressive" because previously generated tokens are used to predict subsequent tokens

https://jalammar.github.io/illustrated-transformer/

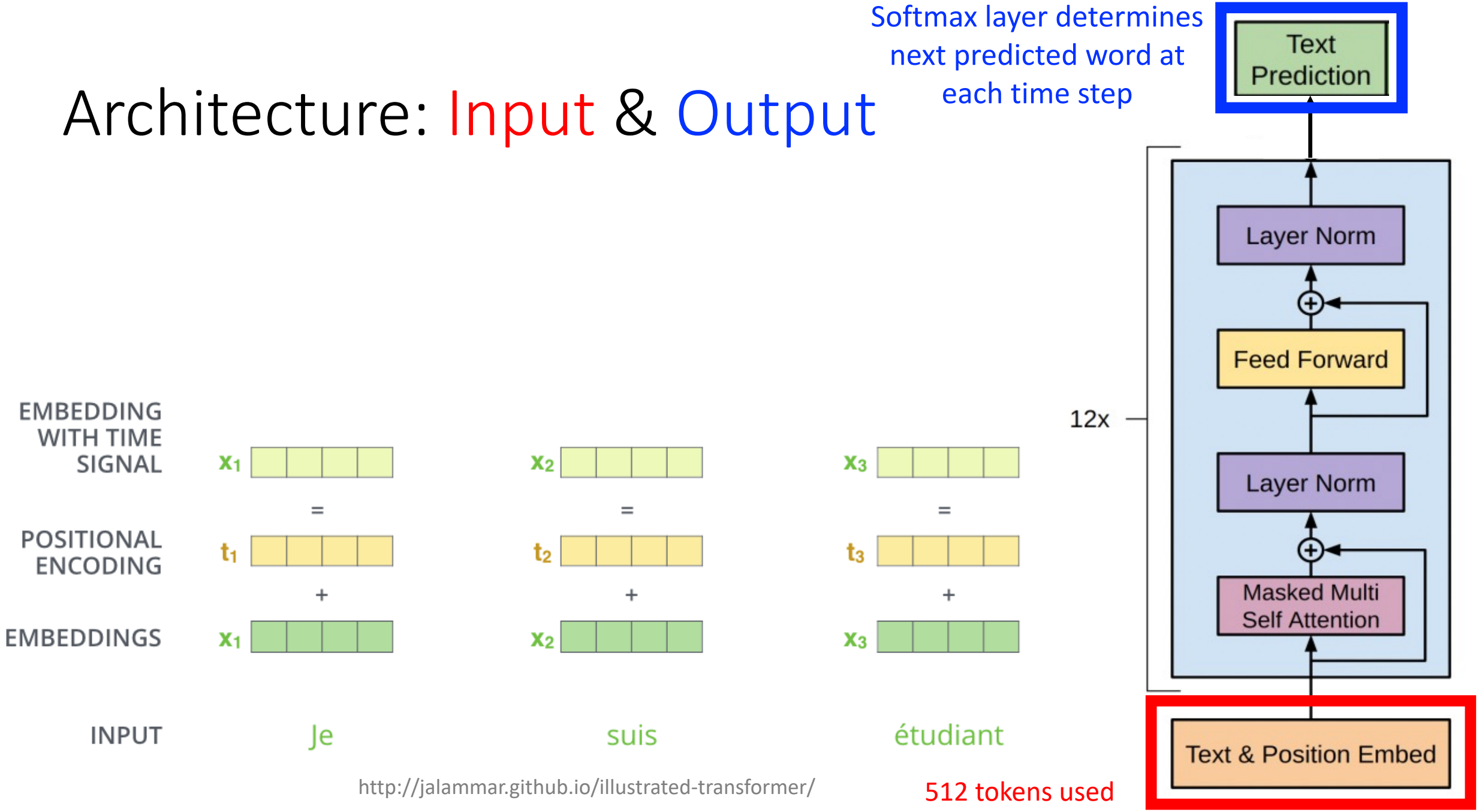
Architecture: Simplified Figure



Radford et al. Improving Language Understanding by Generative Pre-Training. Technical Report 2018.

Architecture: Input & Output

Softmax layer determines next predicted word at each time step



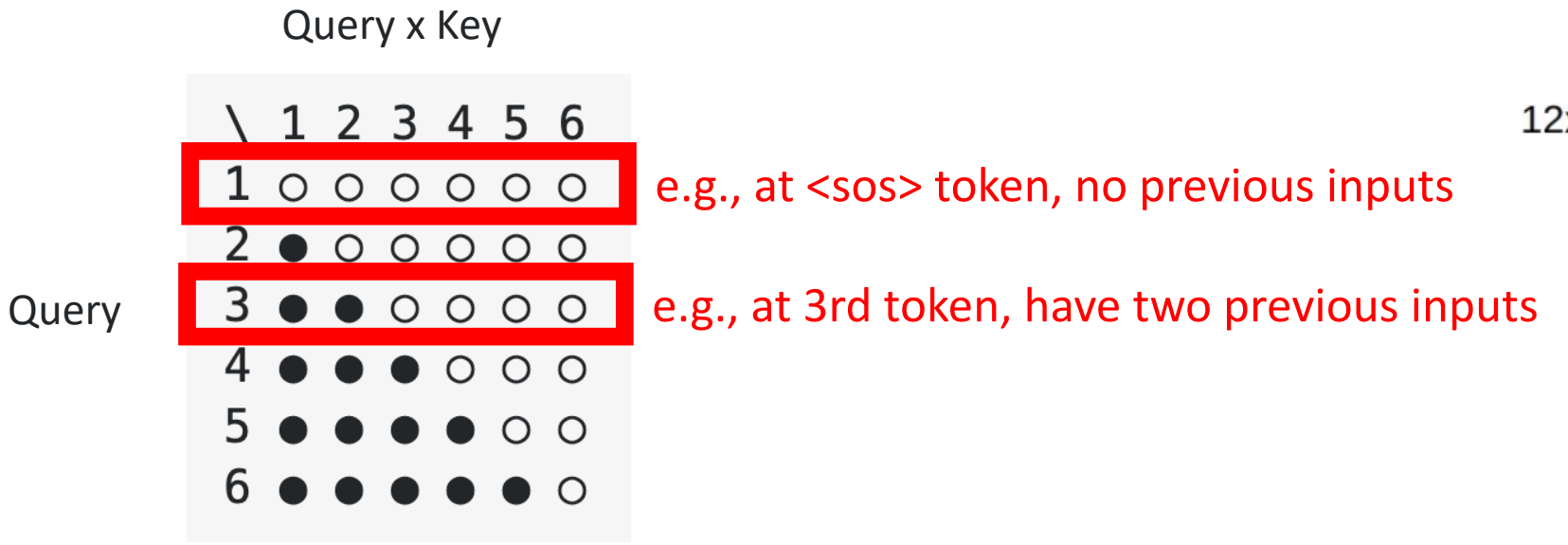
http://jalammar.github.io/illustrated-transformer/

512 tokens used

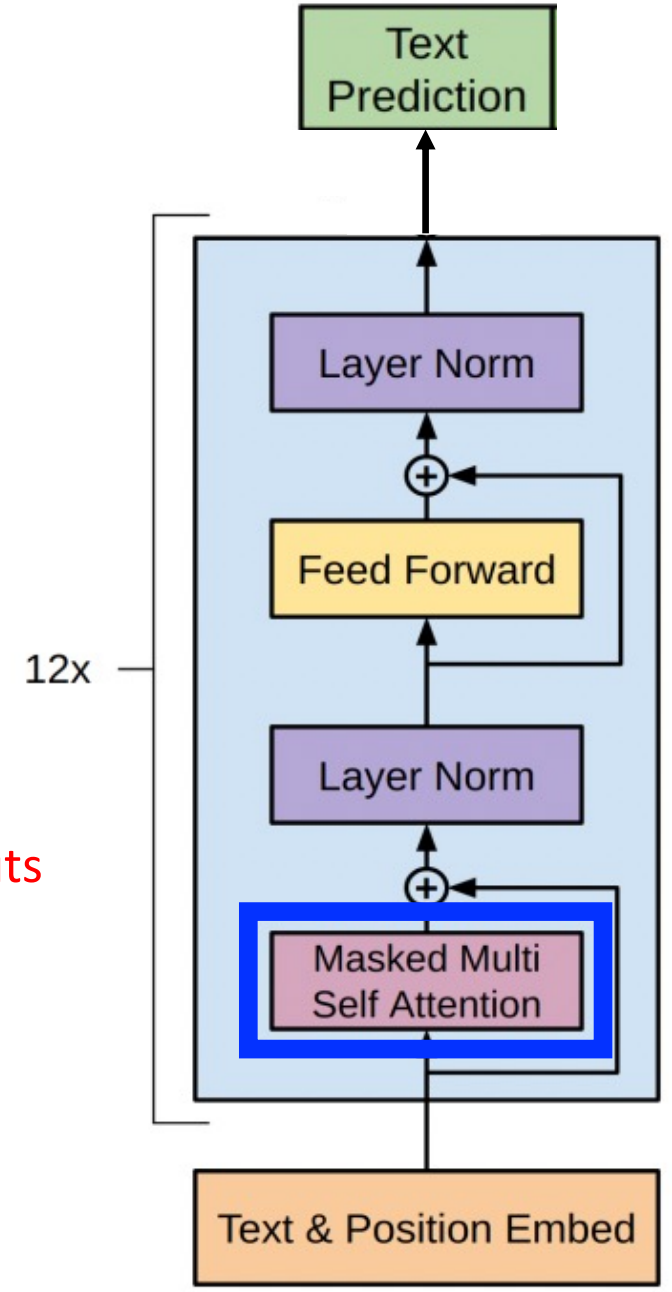
Radford et al. Improving Language Understanding by Generative Pre-Training. Technical Report 2018.

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



Radford et al. Improving Language Understanding by Generative Pre-Training. Technical Report 2018.

Architecture: **Masked** Attention

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

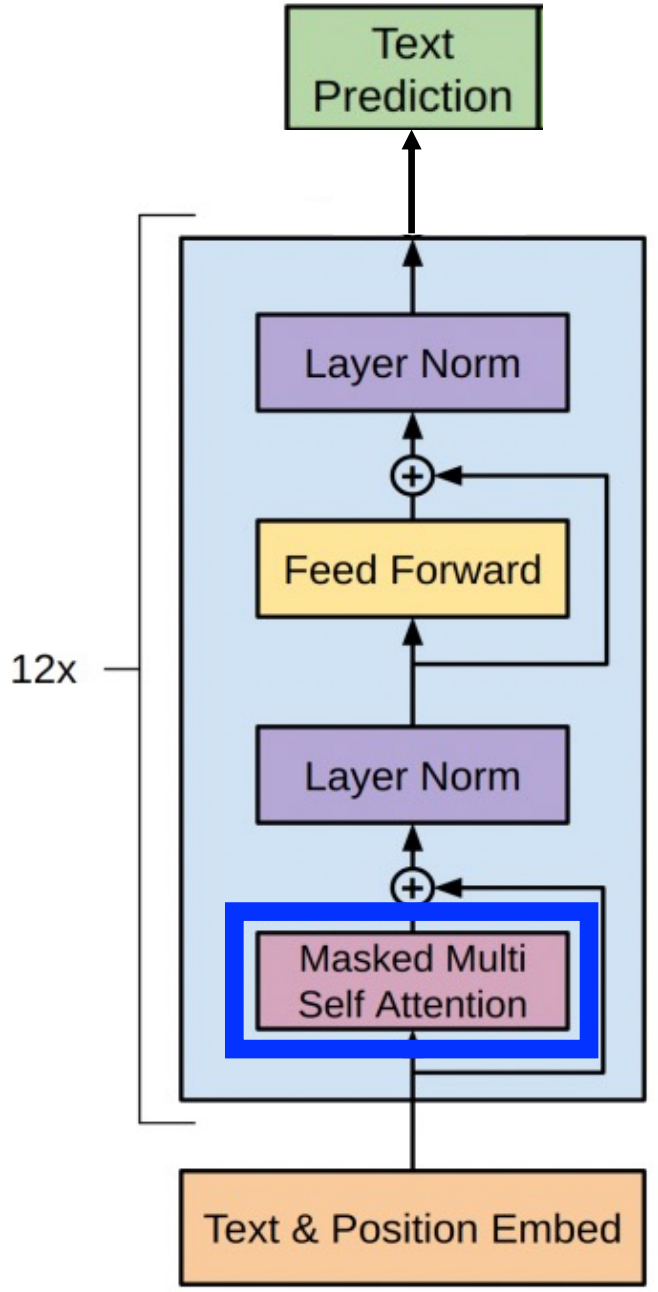
Query x Key

Query

\	1	2	3	4	5	6
1	○	○	○	○	○	○
2	●	○	○	○	○	○
3	●	●	○	○	○	○
4	●	●	●	○	○	○
5	●	●	●	●	○	○
6	●	●	●	●	●	○

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

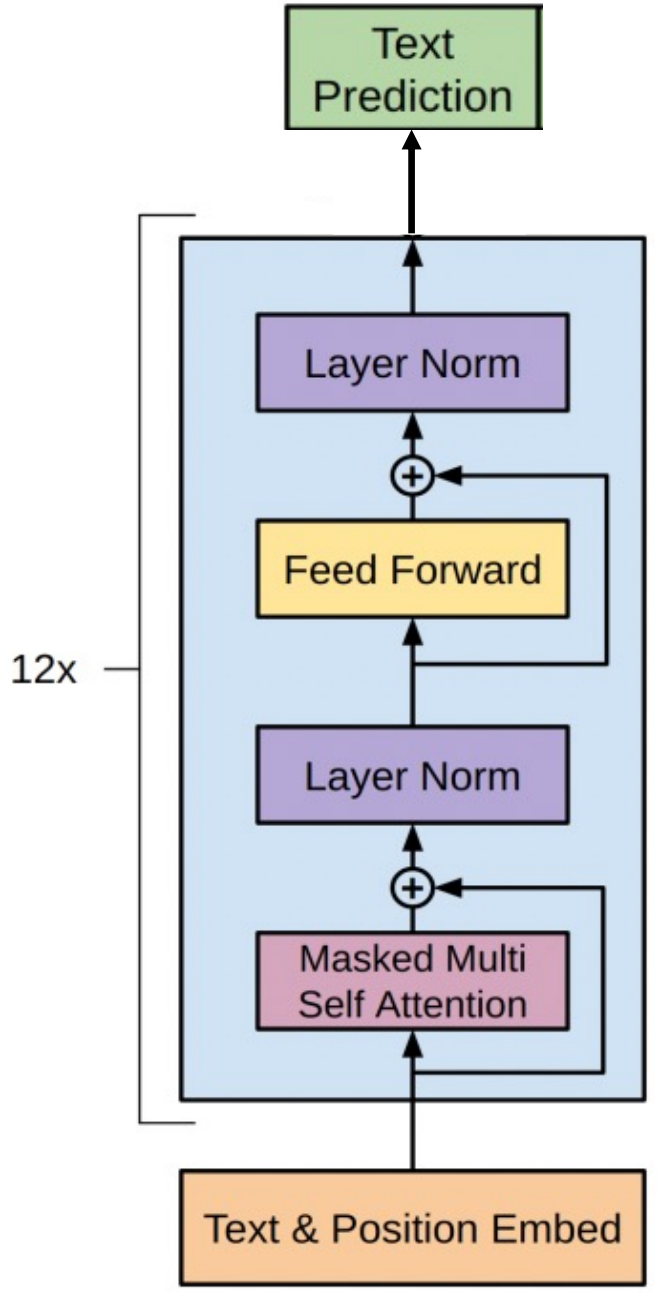
https://stackoverflow.com/questions/64799622/how-is-the-gpts-masked-self-attention-is-utilized-on-fine-tuning-inference



Radford et al. Improving Language Understanding by Generative Pre-Training. Technical Report 2018.

Training

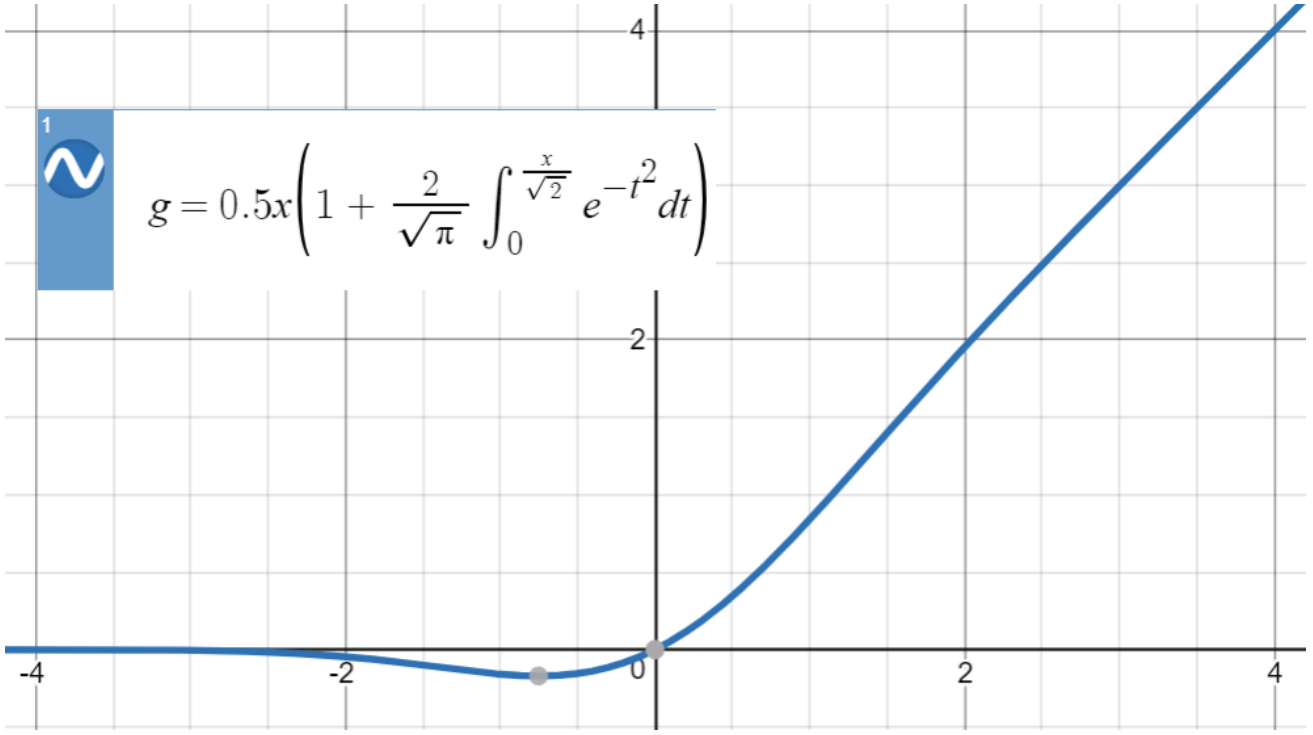
- Dataset: 800M words from BooksCorpus (>7,000 books)
- Optimizer: Adam
- Training duration: 100 epochs



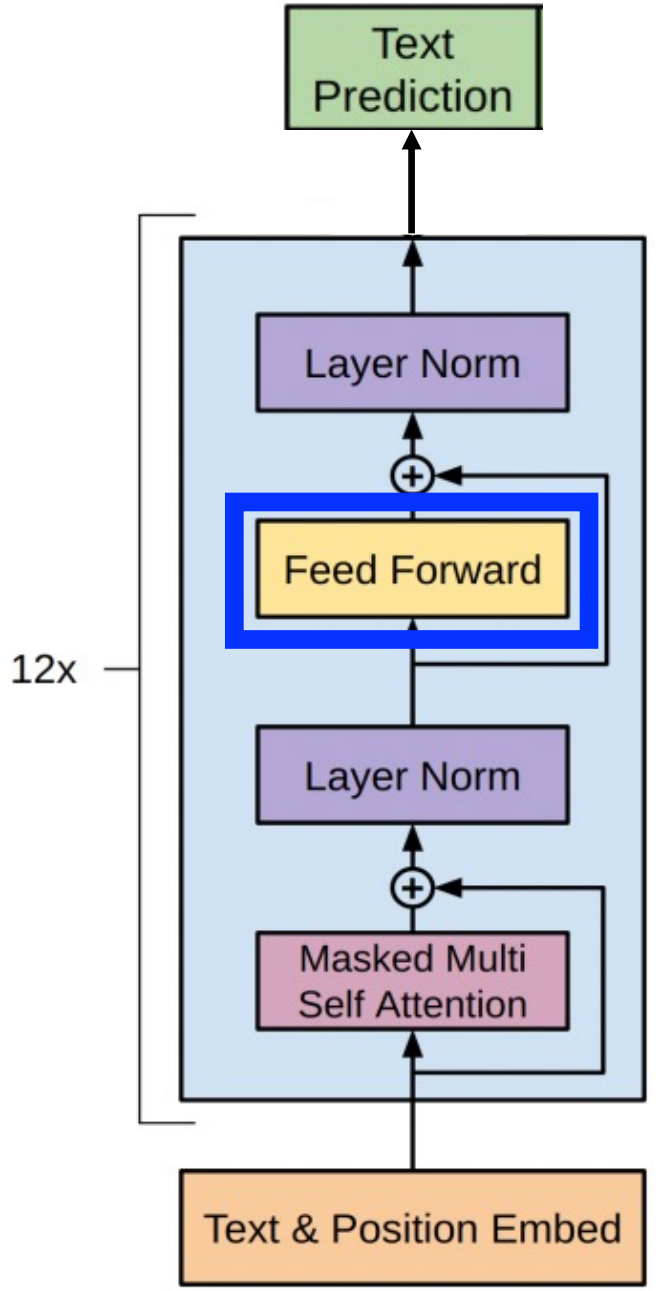
Radford et al. Improving Language Understanding by Generative Pre-Training. Technical Report 2018.

Implementation Details

Activation function: Gaussian error linear unit (GELU)



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



Radford et al. Improving Language Understanding by Generative Pre-Training. Technical Report 2018.

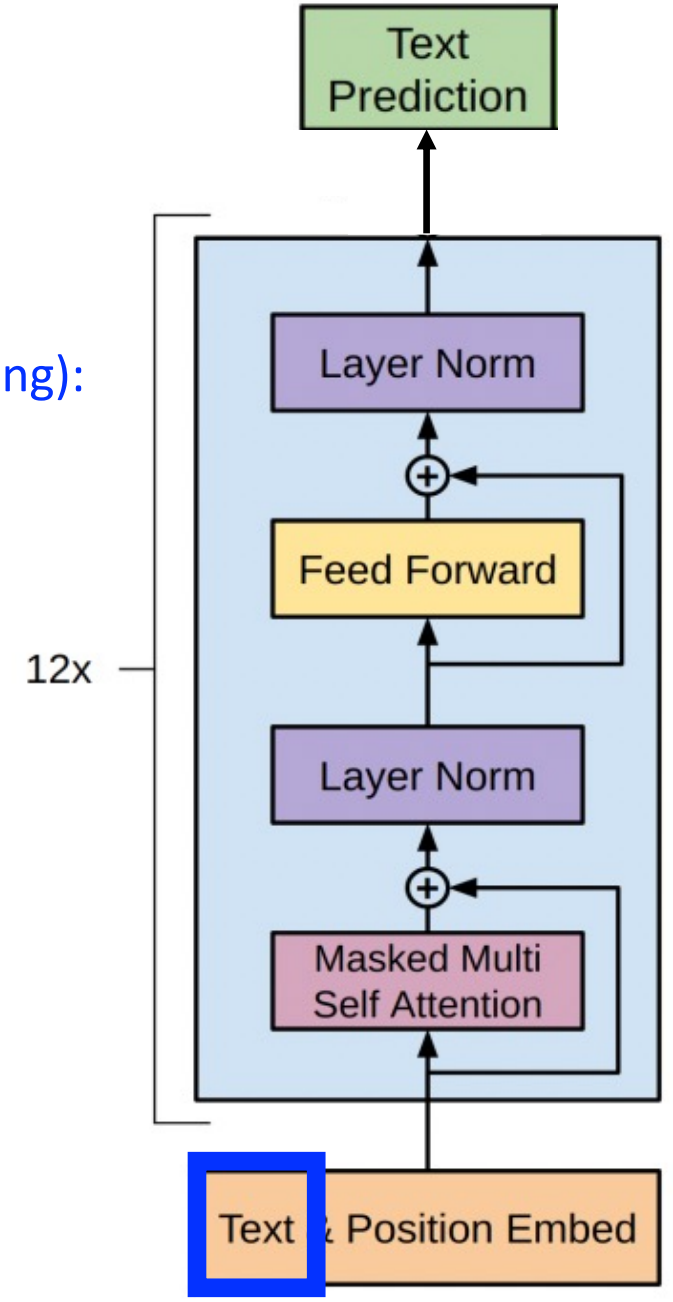
Implementation Details

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



Radford et al. Improving Language Understanding by Generative Pre-Training. Technical Report 2018.

Implementation Details

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

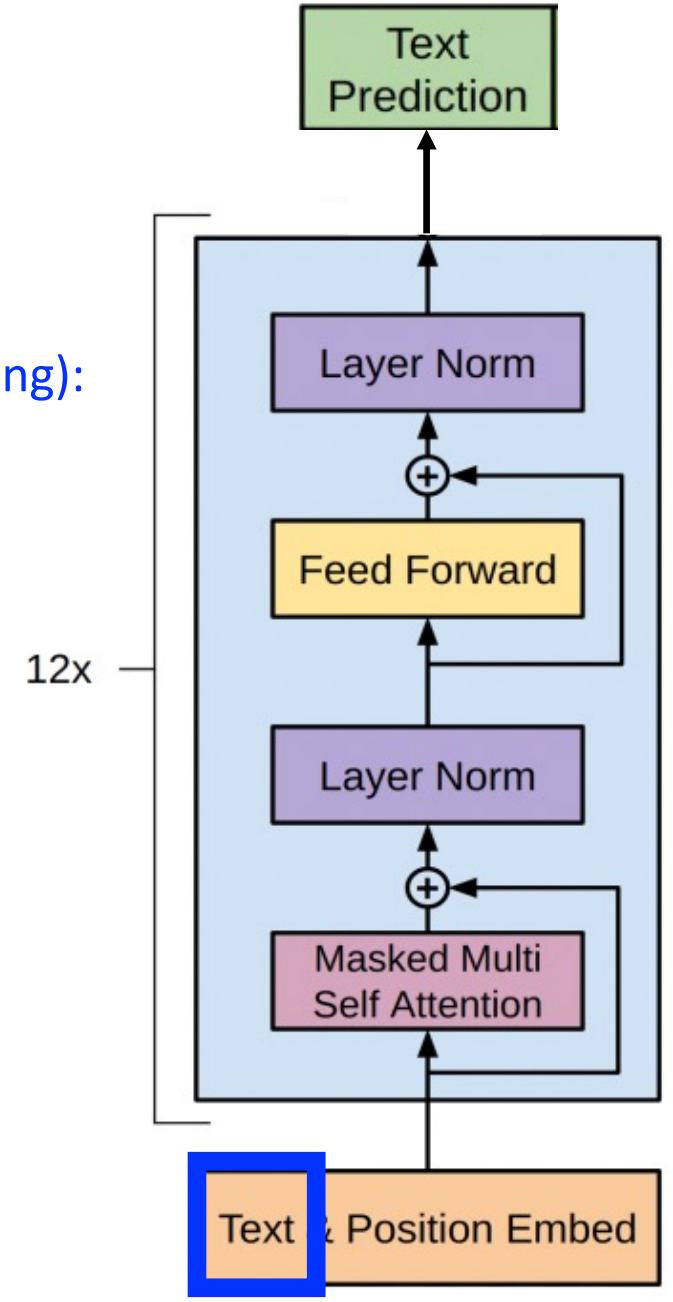
1. Identify all tokens in the training data with their frequency
2. Define vocabulary size; e.g., 14
3. Add all characters in the tokenized input to the vocabulary; e.g.,
4. Until vocabulary is filled, add merged highest frequency symbol pairs

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



Radford et al. Improving Language Understanding by Generative Pre-Training. Technical Report 2018.

Implementation Details

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

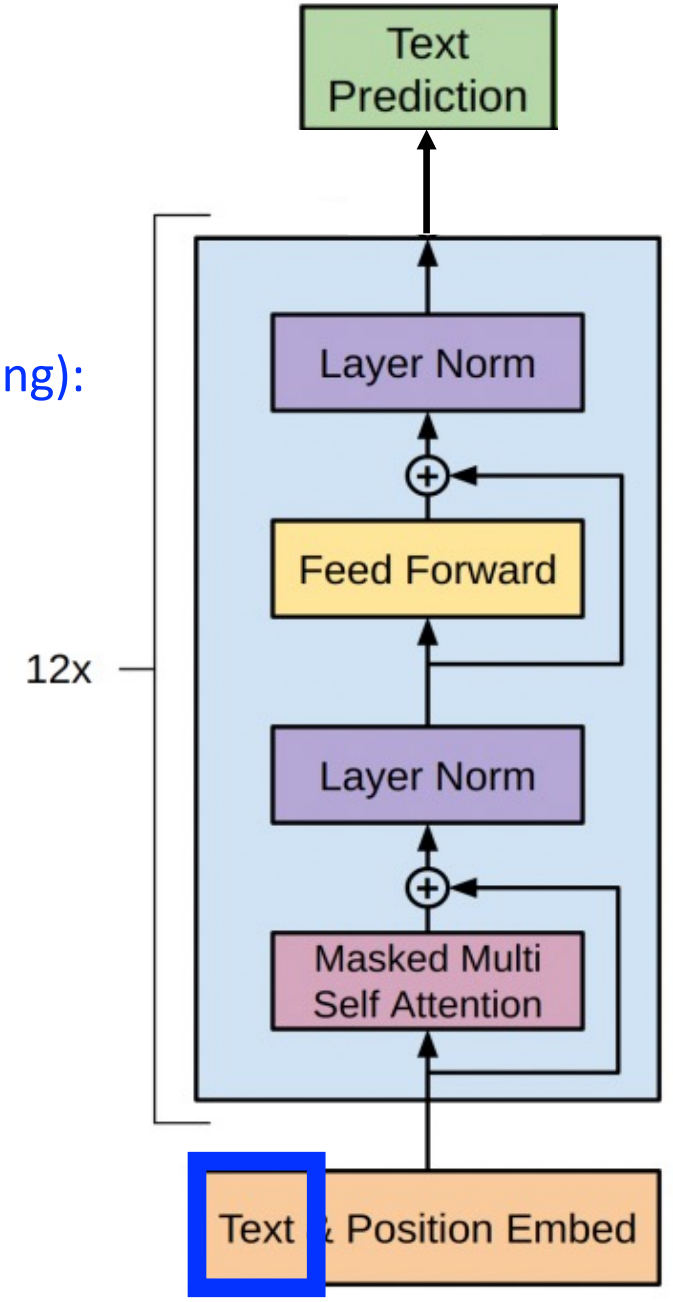
1. Identify all tokens in the training data with their frequency
2. Define vocabulary size; e.g., 14
3. Add all characters in the tokenized input to the vocabulary; e.g.,
4. Until vocabulary is filled, add merged highest frequency symbol pairs

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



Radford et al. Improving Language Understanding by Generative Pre-Training. Technical Report 2018.

Implementation Details

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

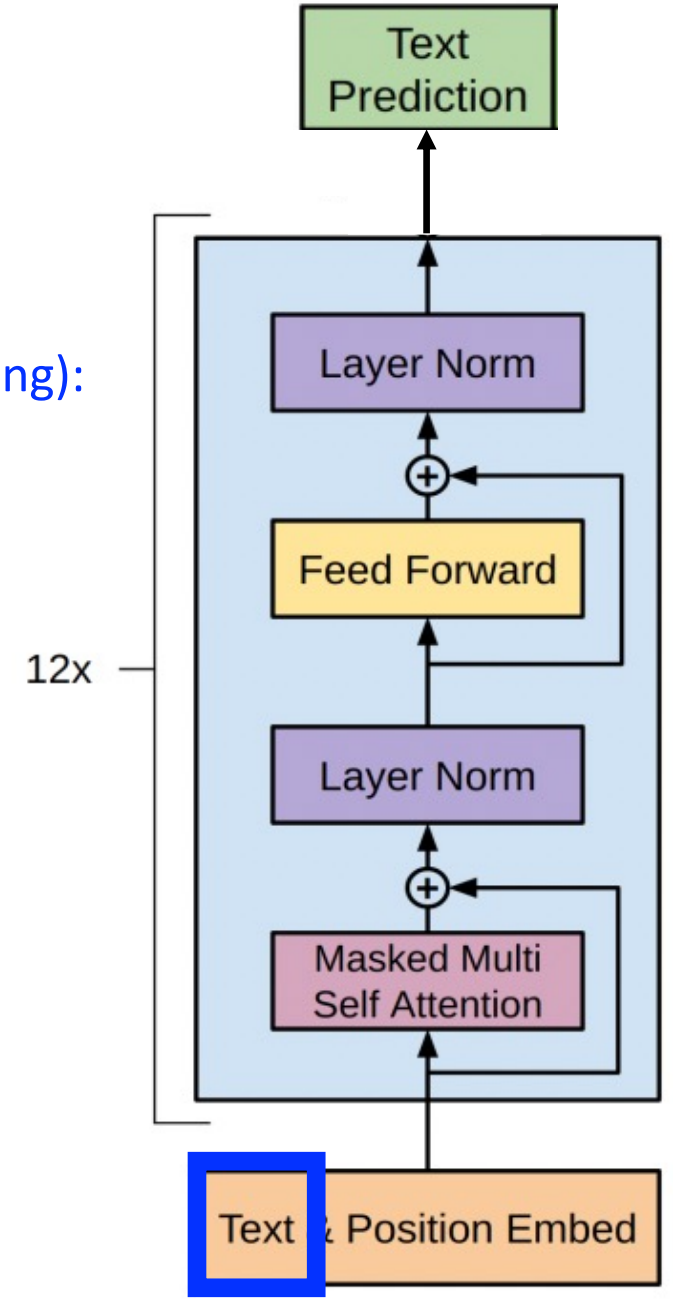
1. Identify all tokens in the training data with their frequency
2. Define vocabulary size; e.g., 14
3. Add all characters in the tokenized input to the vocabulary; e.g.,
4. Until vocabulary is filled, add merged highest frequency symbol pairs

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, me, **men**

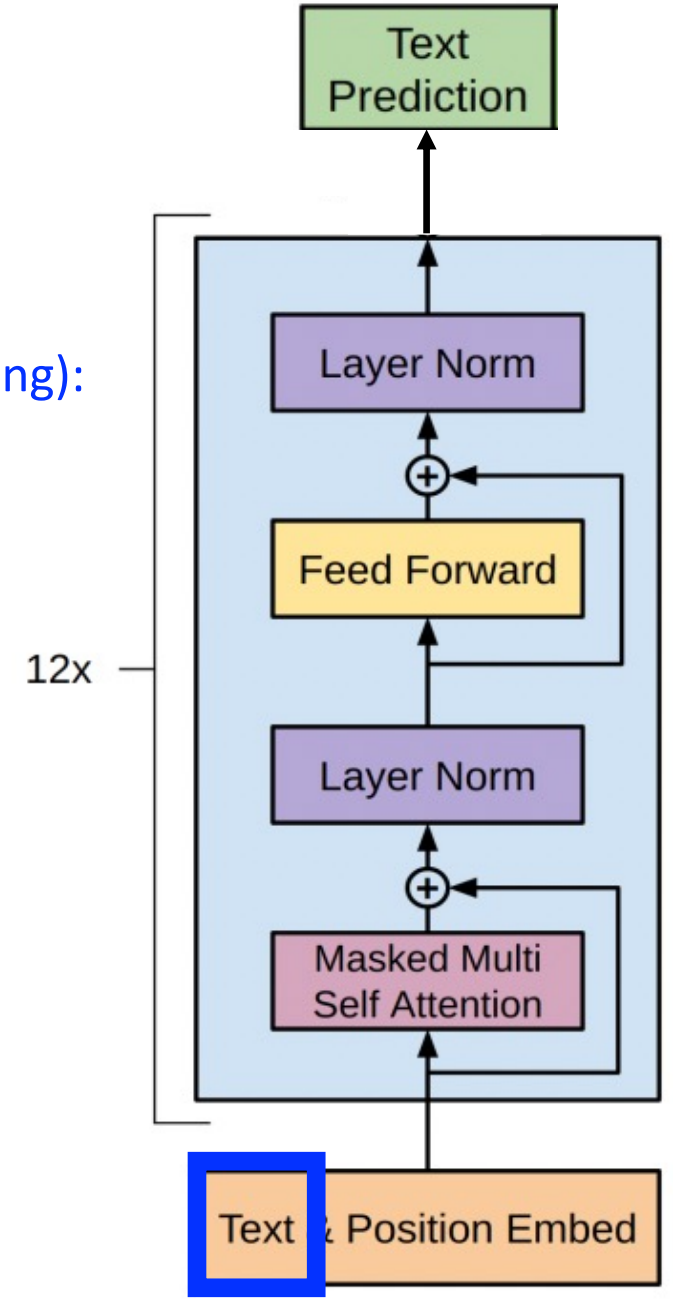
https://static.packt-cdn.com/downloads/9781838821593_ColorImages.pdf



Radford et al. Improving Language Understanding by Generative Pre-Training. Technical Report 2018.

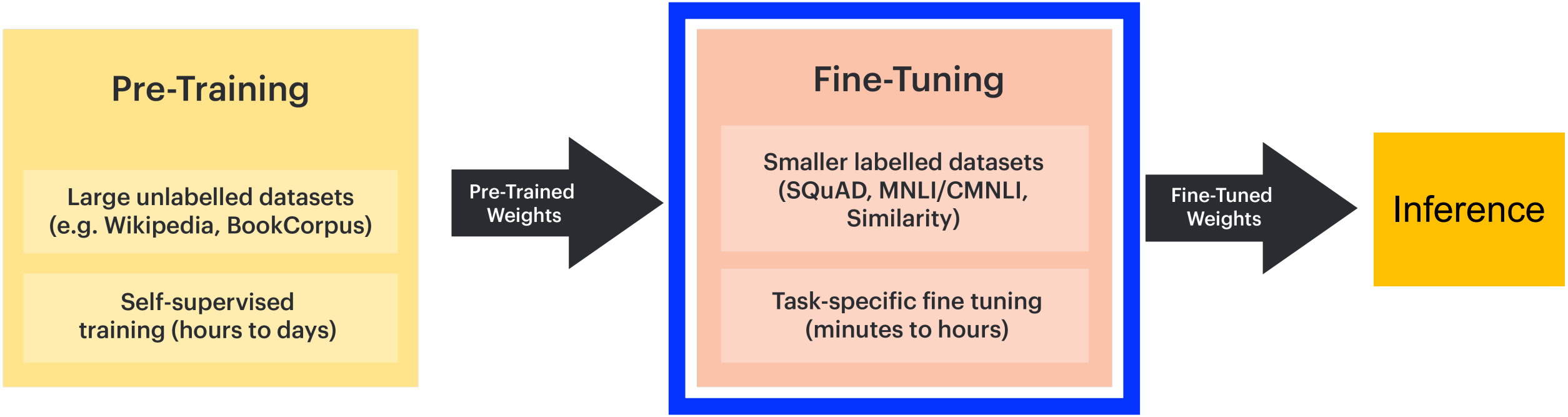
Implementation Details

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



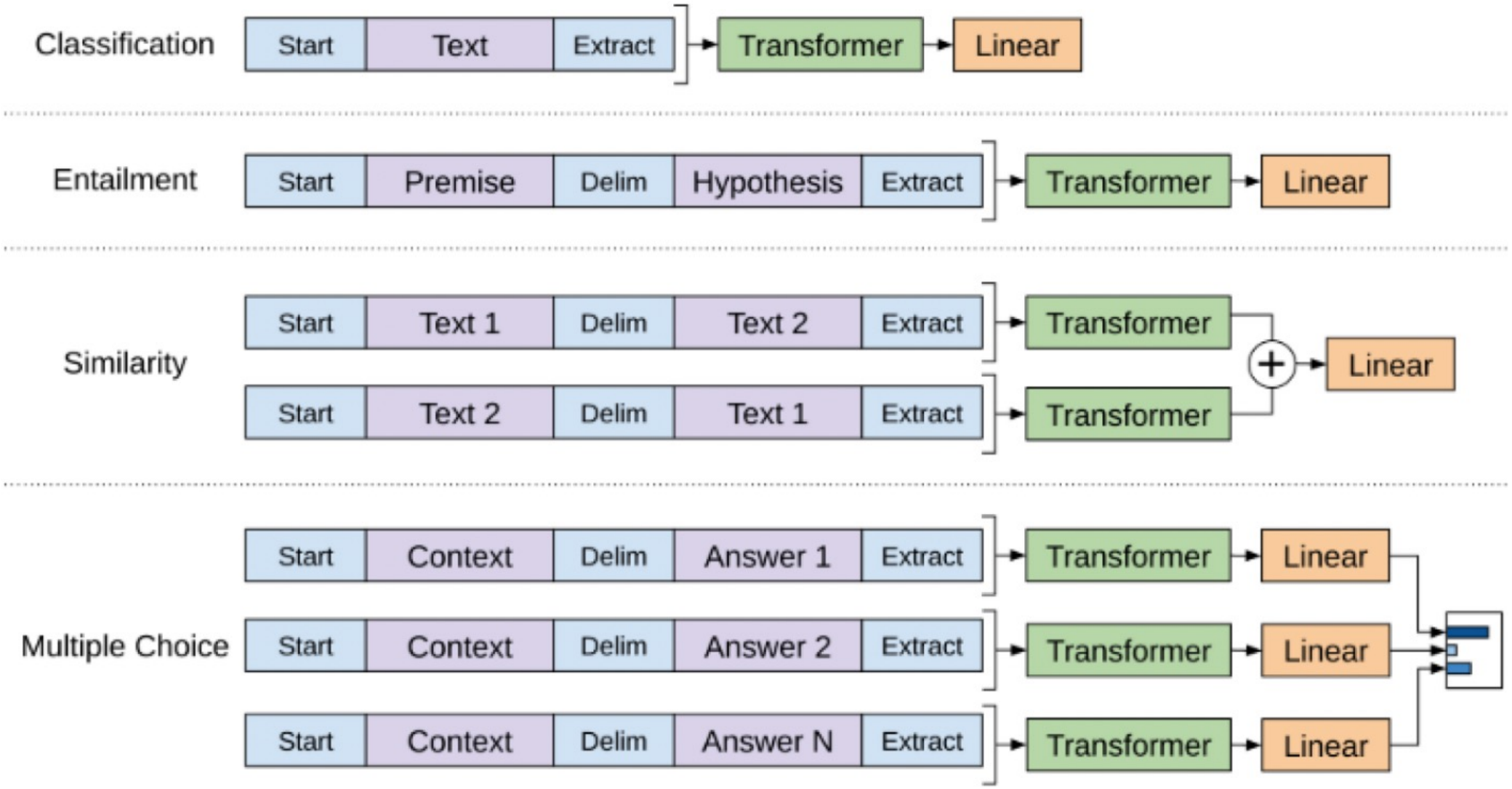
Radford et al. Improving Language Understanding by Generative Pre-Training. Technical Report 2018.

GPT: Generative Pre-Training



https://docs.graphcore.ai/projects/bert-training/en/latest/bert.html

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



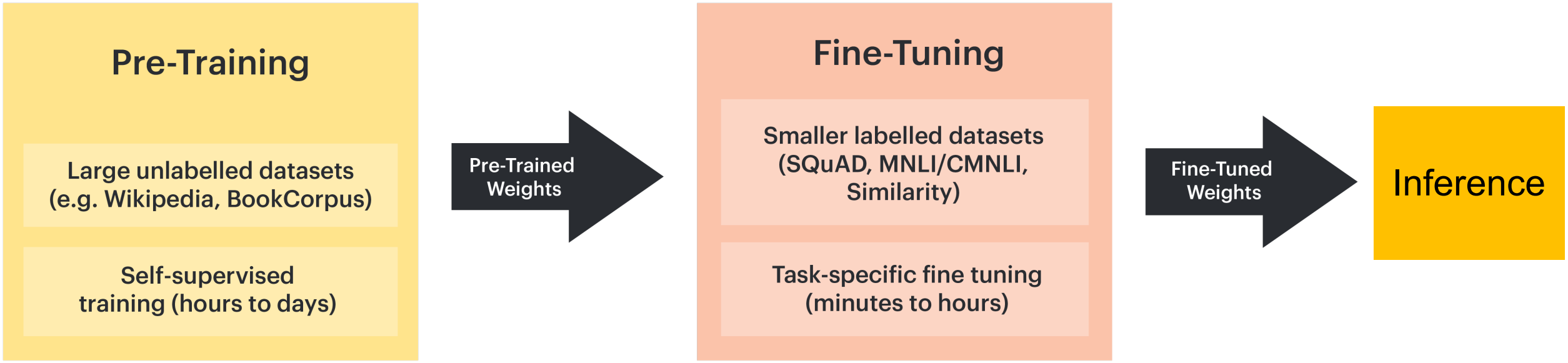
Radford et al. Improving Language Understanding by Generative Pre-Training. Technical Report 2018.

Experimental Findings

Achieved the best performance on 9 NLP dataset challenges

Radford et al. Improving Language Understanding by Generative Pre-Training. Technical Report 2018.

GPT: Generative Pre-Training



https://docs.graphcore.ai/projects/bert-training/en/latest/bert.html

Today’s Topics

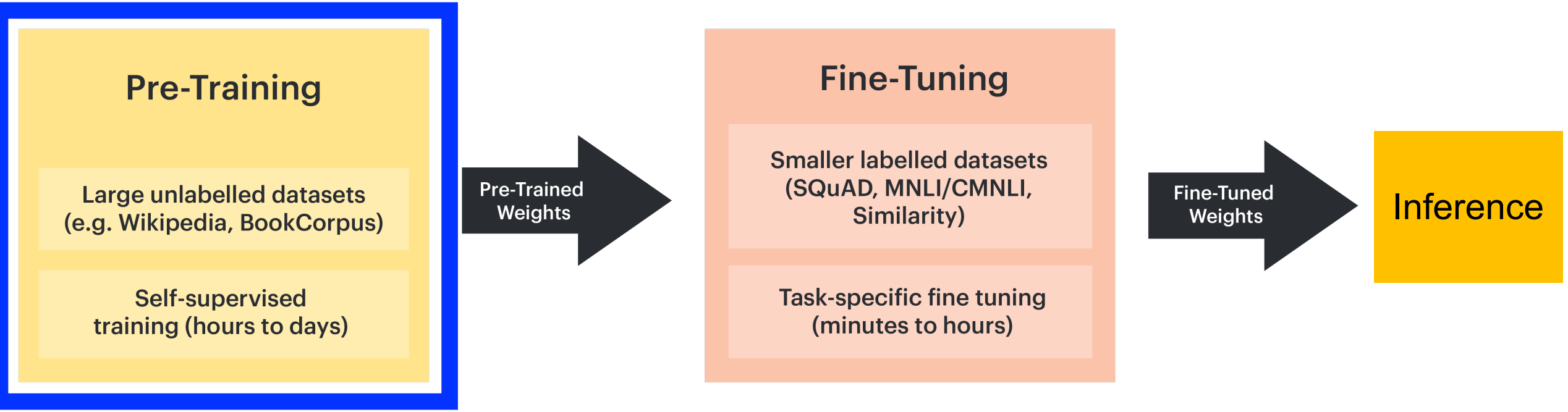
- Explosion of transformers
- GPT
- BERT
- ViT
- Programming tutorial

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



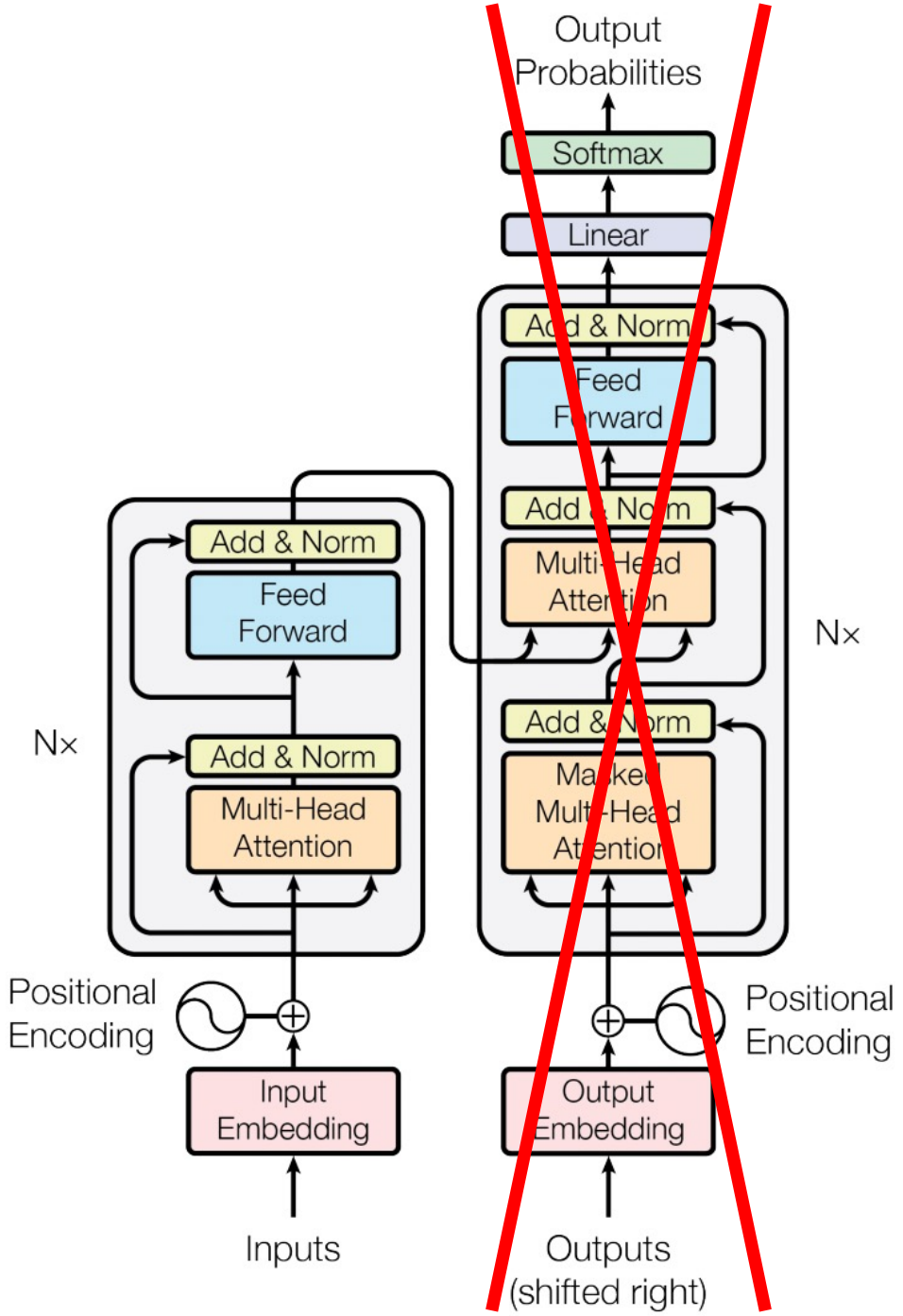
https://docs.graphcore.ai/projects/bert-training/en/latest/bert.html

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



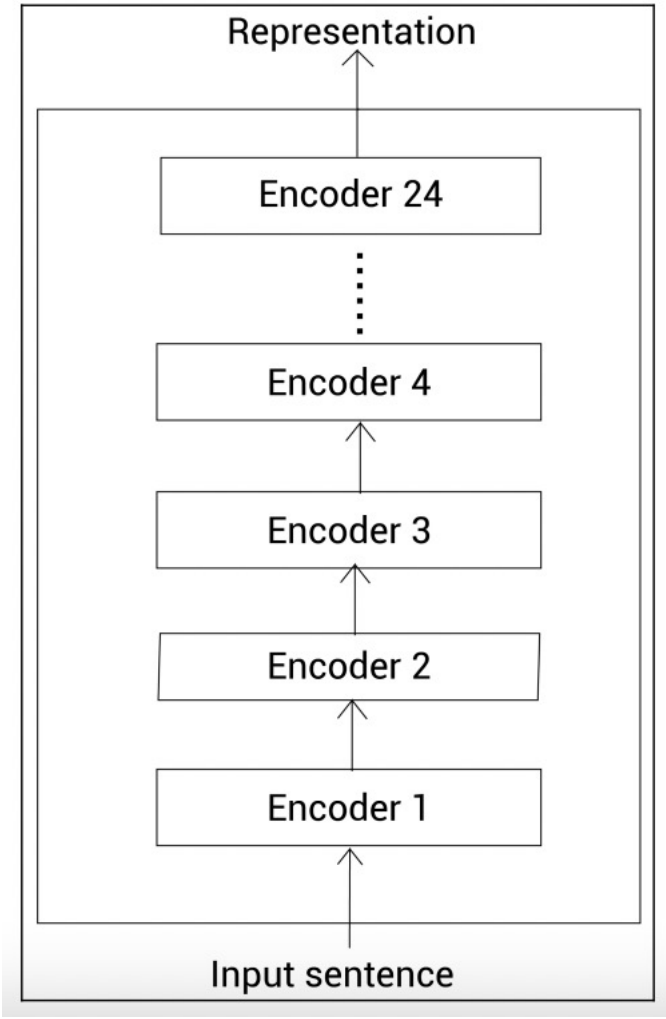
Vaswani et al. Attention Is All You Need. Neurips 2017.

Architecture: Variants

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

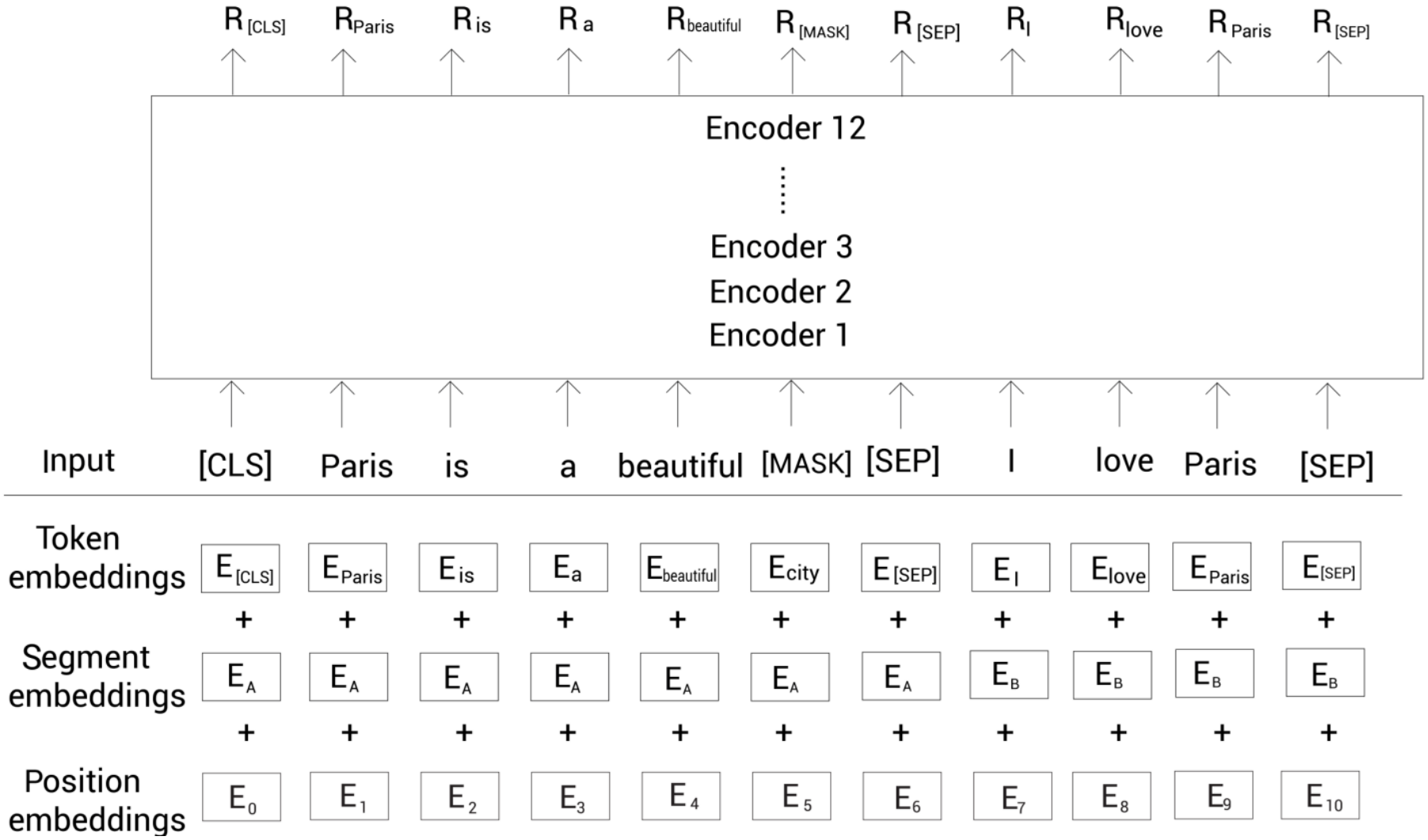
	H=128	H=256	H=512	H=768
L=2	2/128 (BERT-tiny)	2/256	2/512	2/768
L=4	4/128	4/256 (BERT-mini)	4/512 (BERT-small)	4/768
L=6	6/128	6/256	6/512	6/768
L=8	8/128	8/256	8/512 (BERT-medium)	8/768
L=10	10/128	10/256	10/512	10/768
L=12	12/128	12/256	12/512	12/768 (BERT-base)

BERT-large (H = 1024)



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

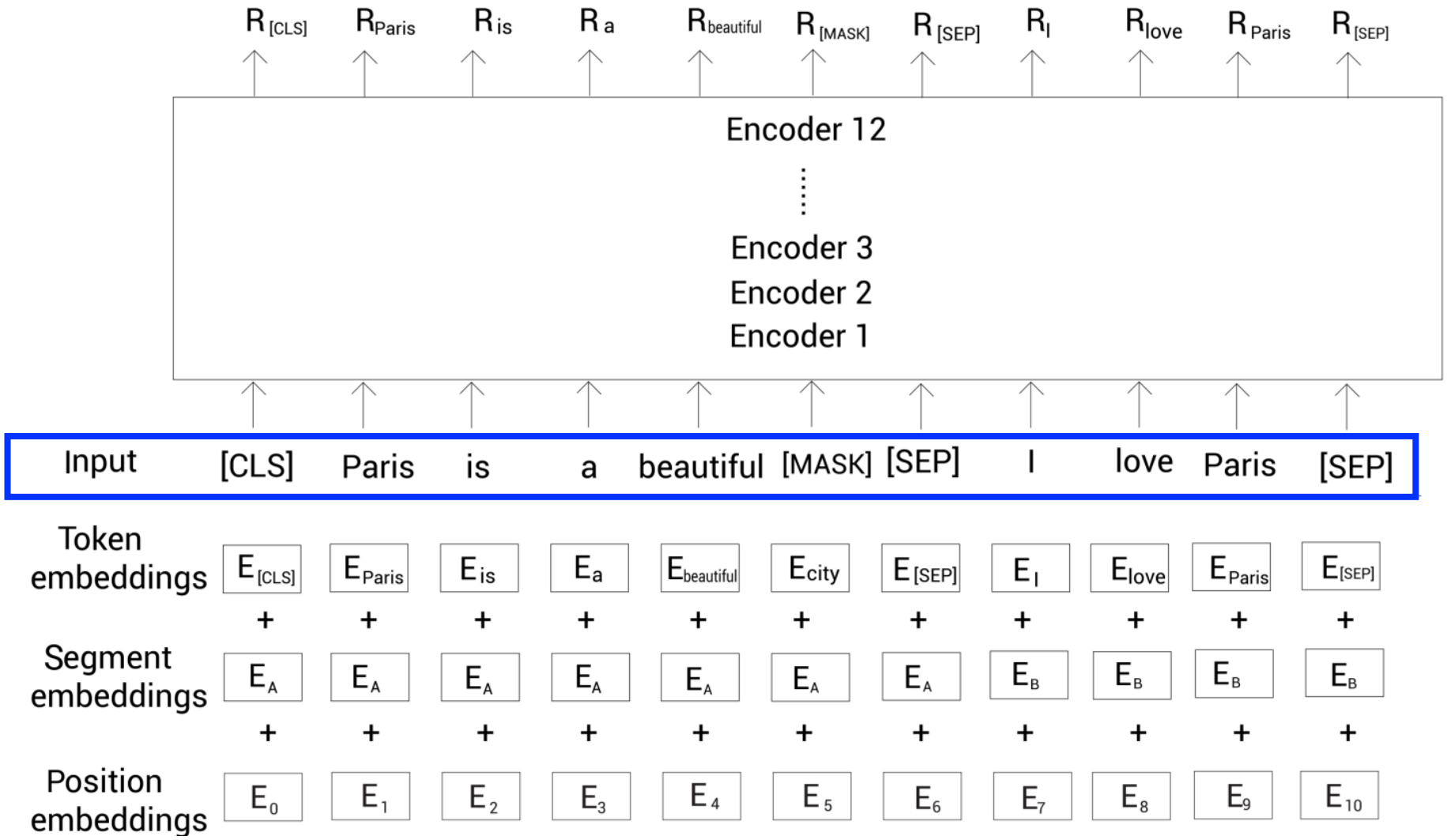
Architecture: BERT-Base (Matches Size of GPT)



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

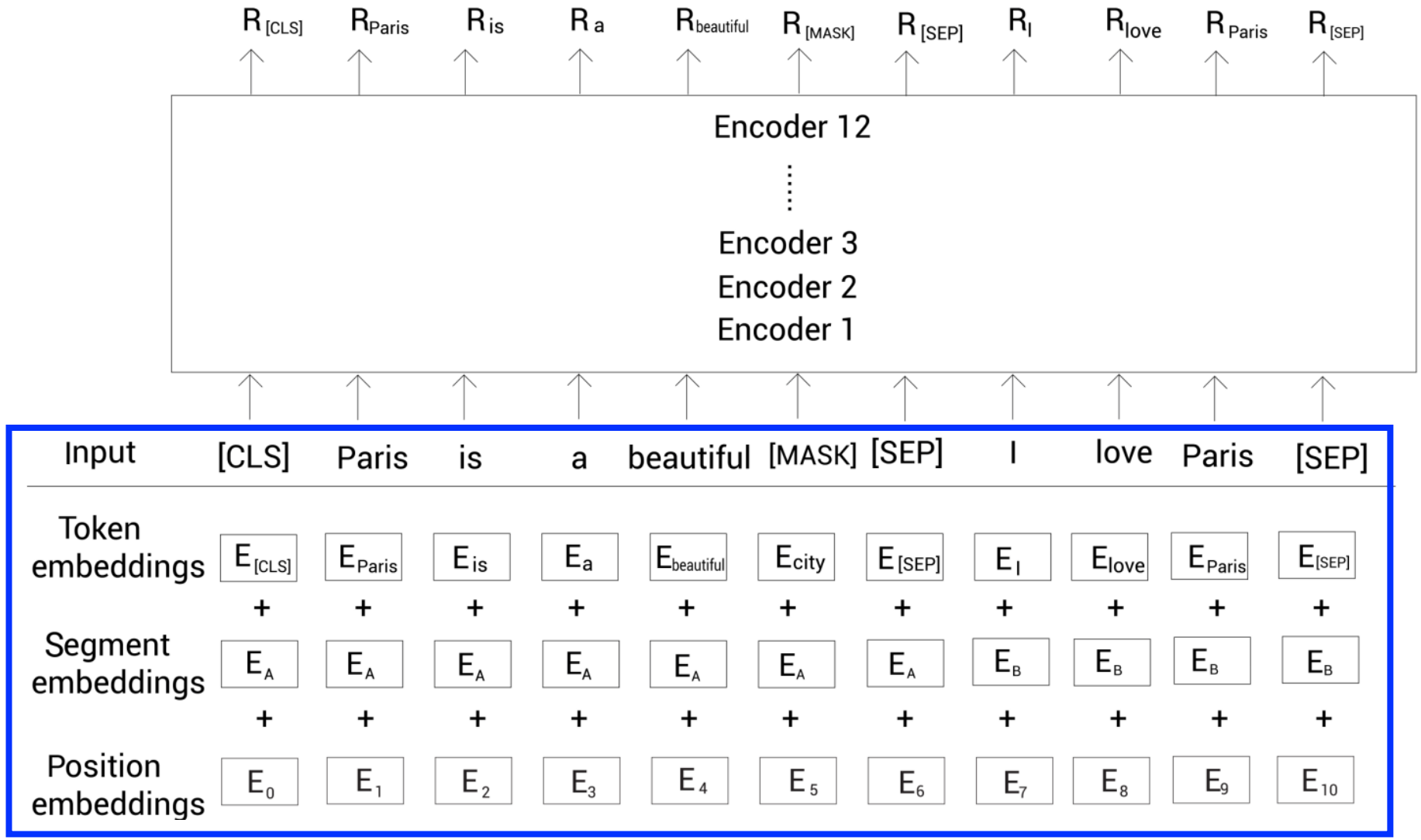
Architecture: Input

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



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

Architecture: Input

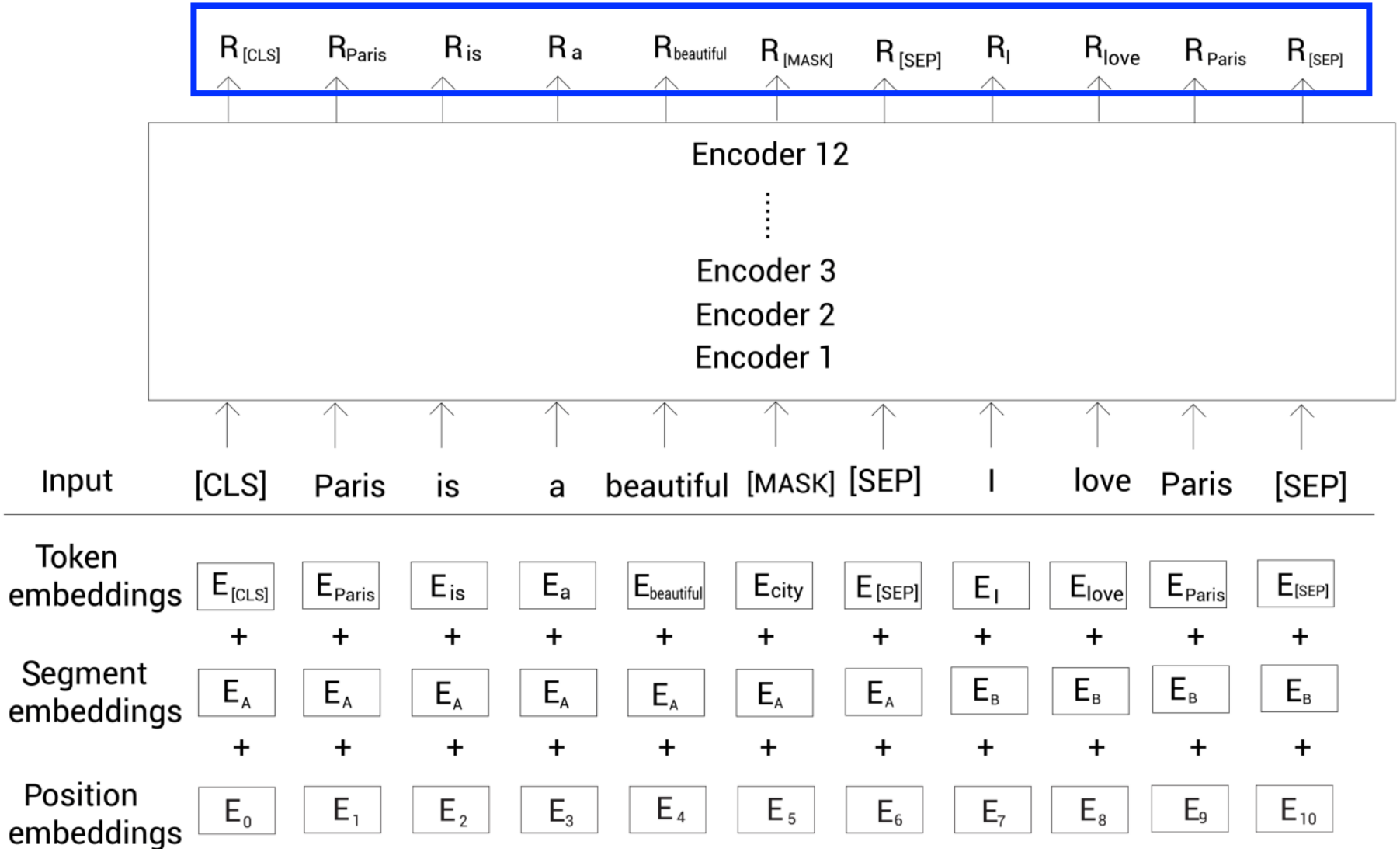


Input is addition of a segment embedding to the token and position embeddings (helps differentiate which tokens belong to which sentence)

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

Architecture: Output

New representation of each input that accounts for context



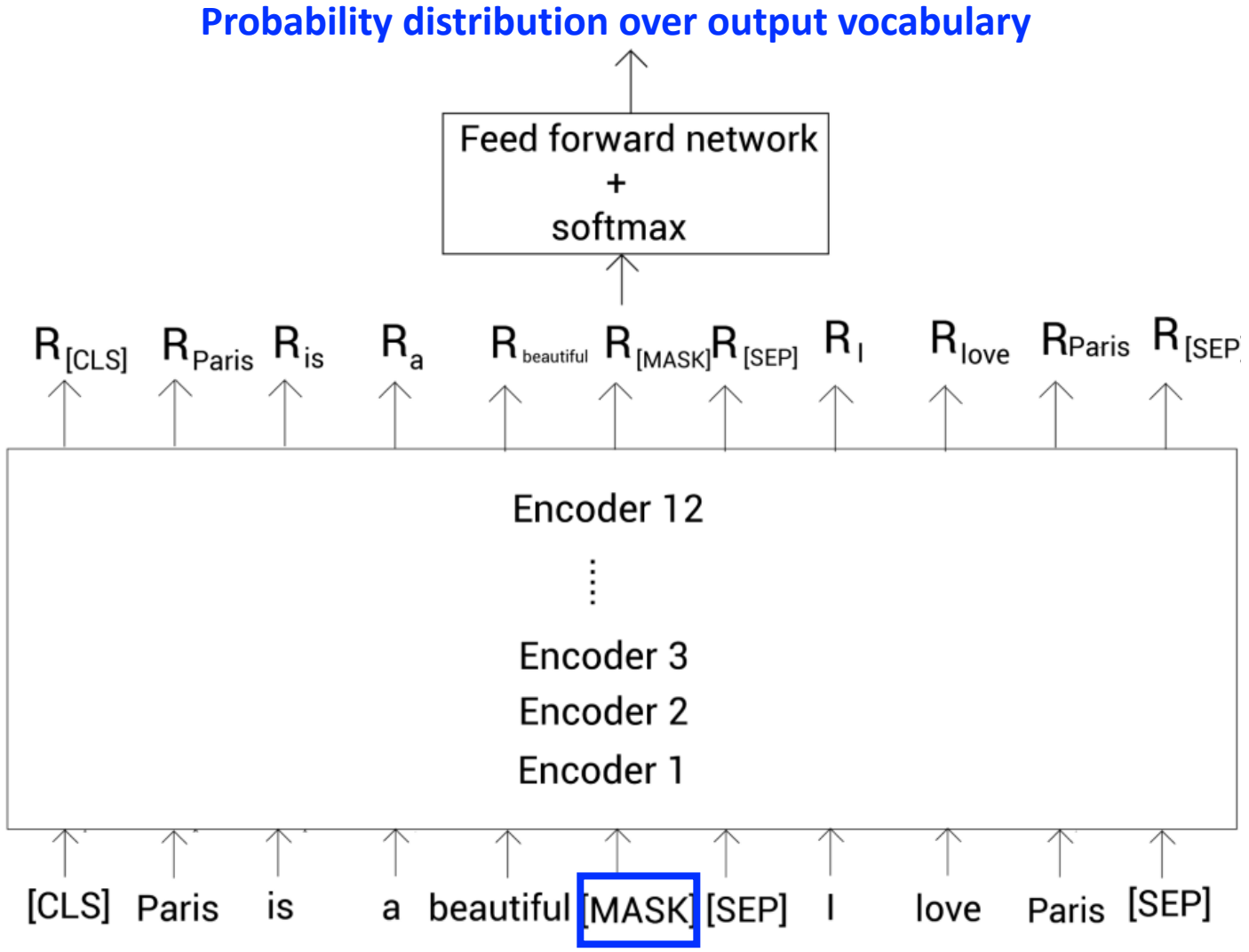
https://static.packt-cdn.com/downloads/9781838821593_ColorImages.pdf

Architecture: Predicting Masked Token Task

15% of random tokens from sequence masked

- 80% use [MASK]
- 10% use a random token
- 10% use original token

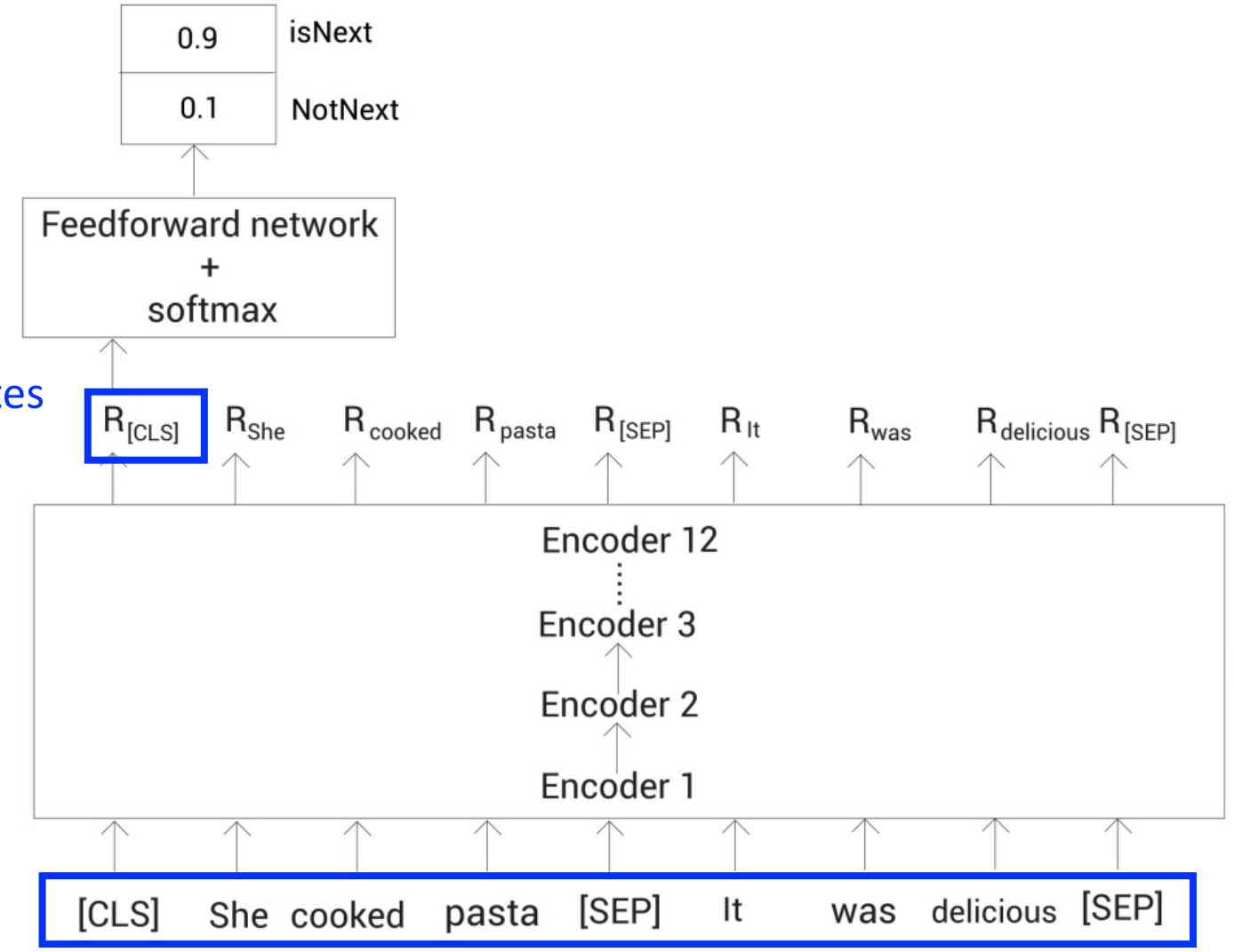
Multiple masking options encourage the model to pay attention to each token separately



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

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

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

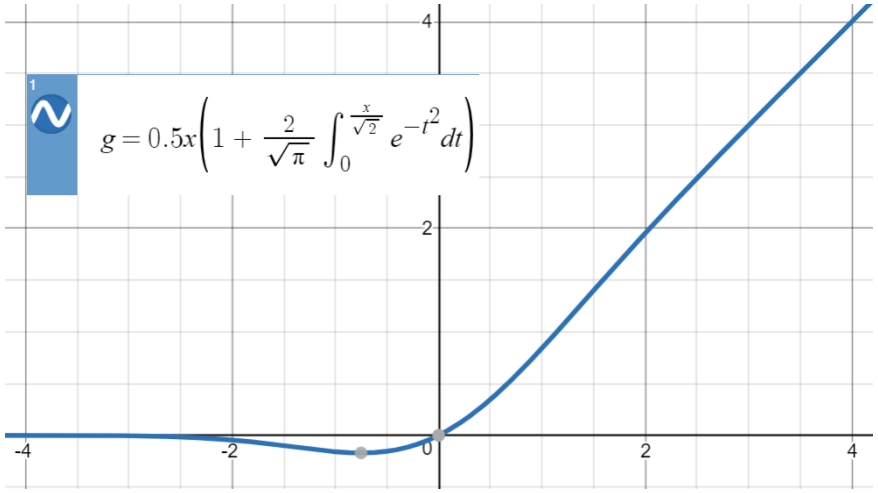
Training

- 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

Devlin et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv 2018.

Implementation Details: Mimics GPT

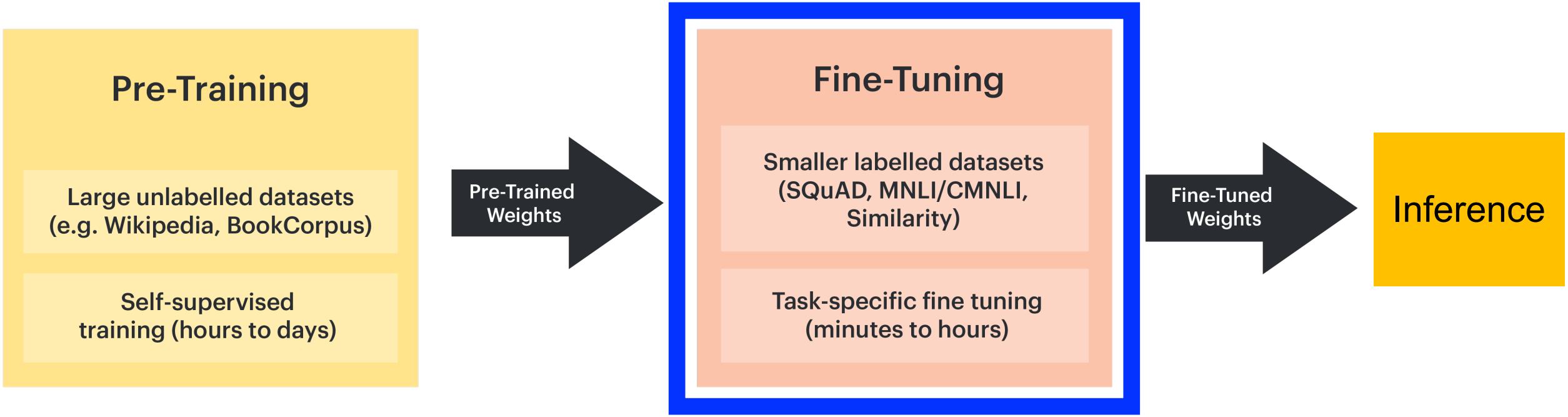
- Gaussian error linear unit (GELU) used as activation function in feedforward layers



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

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

BERT: Bidirectional Encoder Representation from Transformer



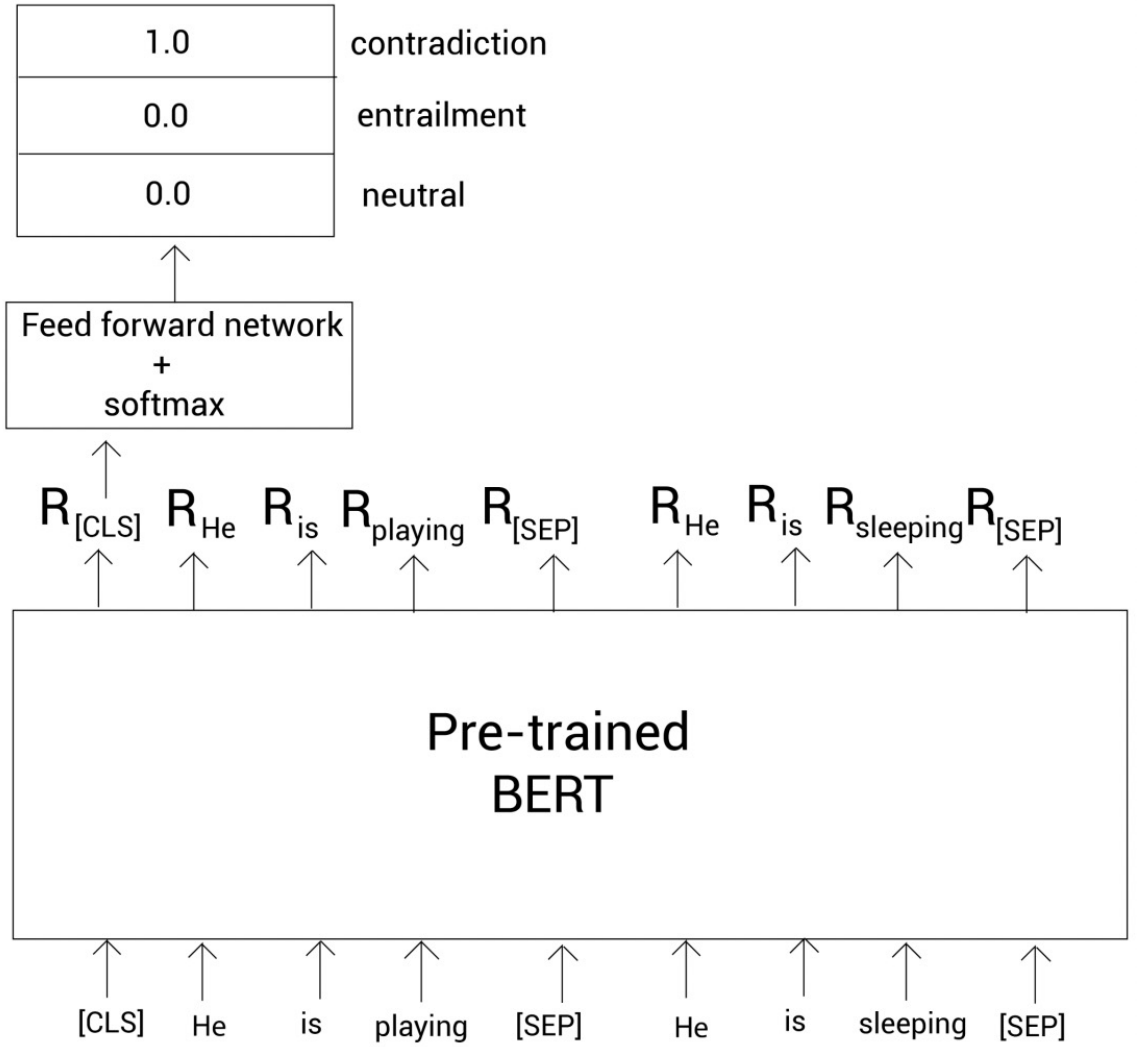
https://docs.graphcore.ai/projects/bert-training/en/latest/bert.html

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

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

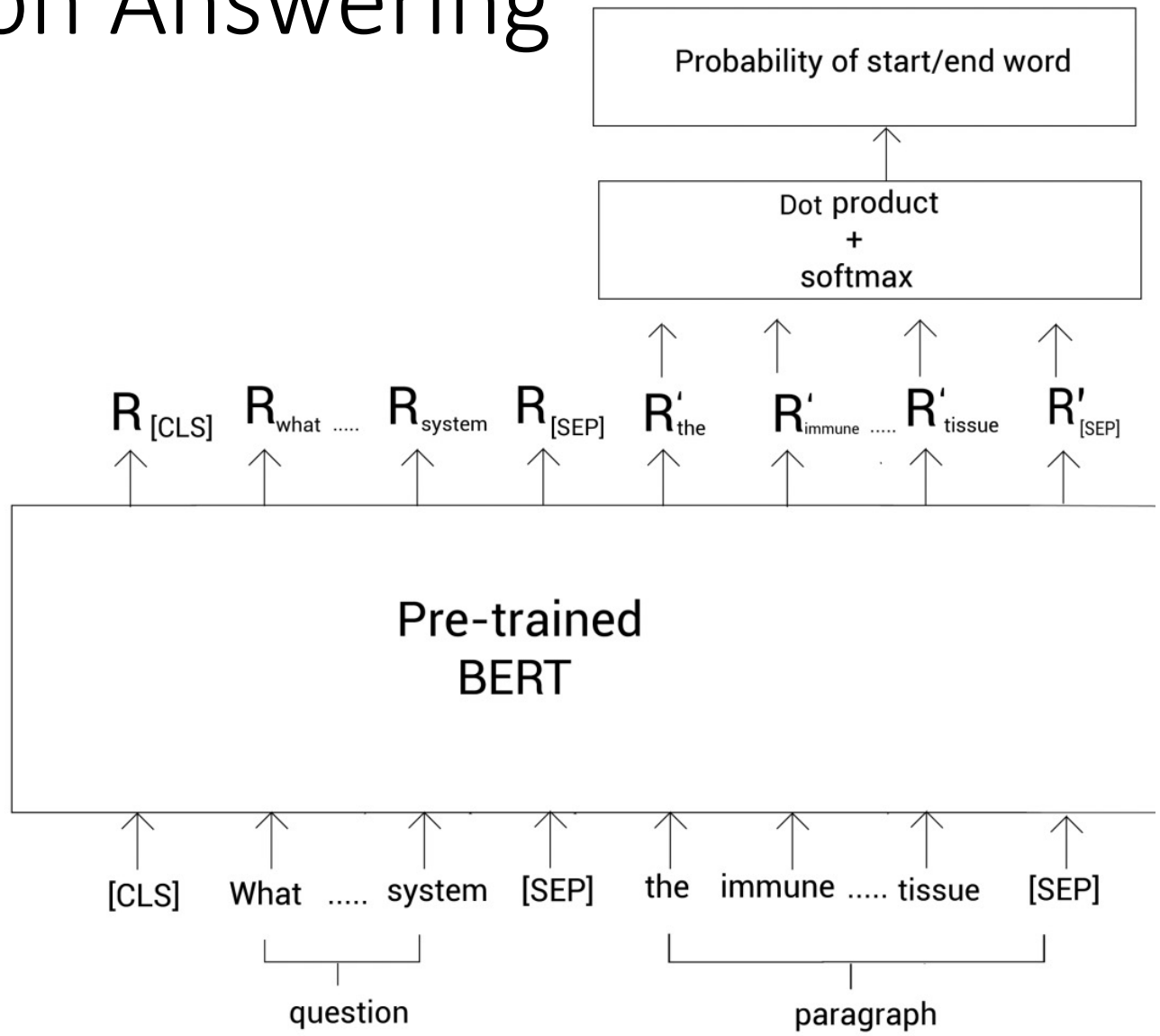
Fine-Tuning for Natural Language Inference



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

Fine-Tuning for Question Answering

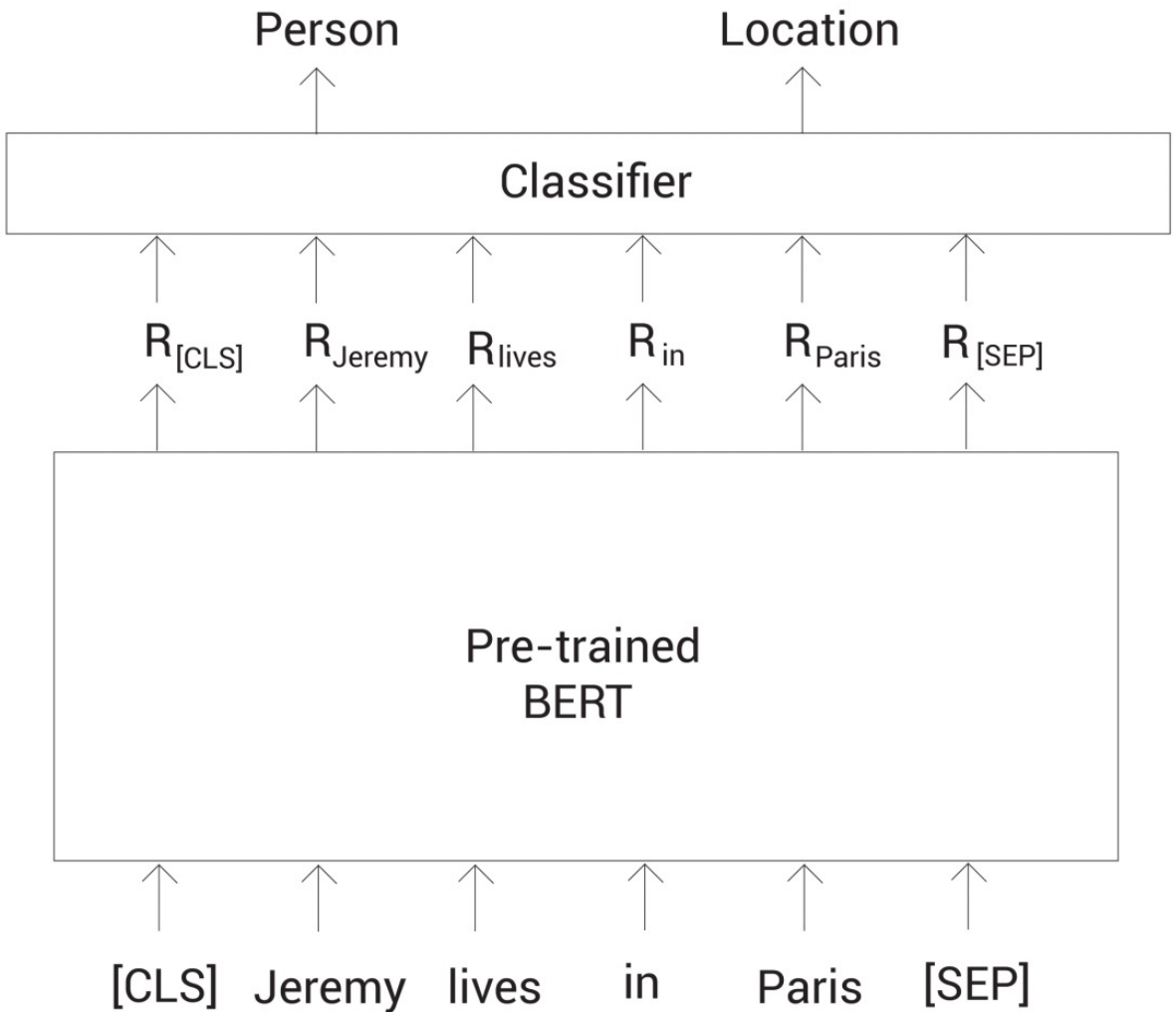
To find indices of the start and end words in the paragraph, two vector representations are learned that lead to the approximate softmax output when computing the dot product with each token



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

Fine-Tuning for Named Entity Recognition

Each token's new representation is passed to a classifier



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

Experimental Findings

Achieved the best performance on 11 NLP dataset challenges

Devlin et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv 2018.

Experimental Findings

Tasks	Dev Set MNLI-m (Acc)	QNLI (Acc)	MRPC (Acc)	SST-2 (Acc)	SQuAD (F1)
$\text{BERT}_{\text{BASE}}$	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

Devlin et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv 2018.

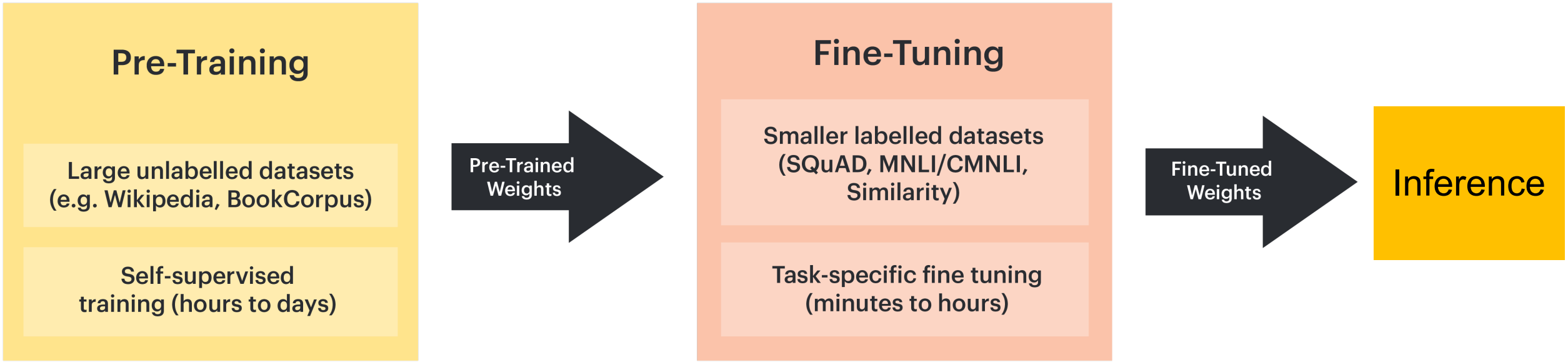
Experimental Findings

Tasks	Dev Set MNLI-m (Acc)	QNLI (Acc)	MRPC (Acc)	SST-2 (Acc)	SQuAD (F1)
$\text{BERT}_{\text{BASE}}$	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)

Devlin et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv 2018.

BERT: Bidirectional Encoder Representation from Transformer

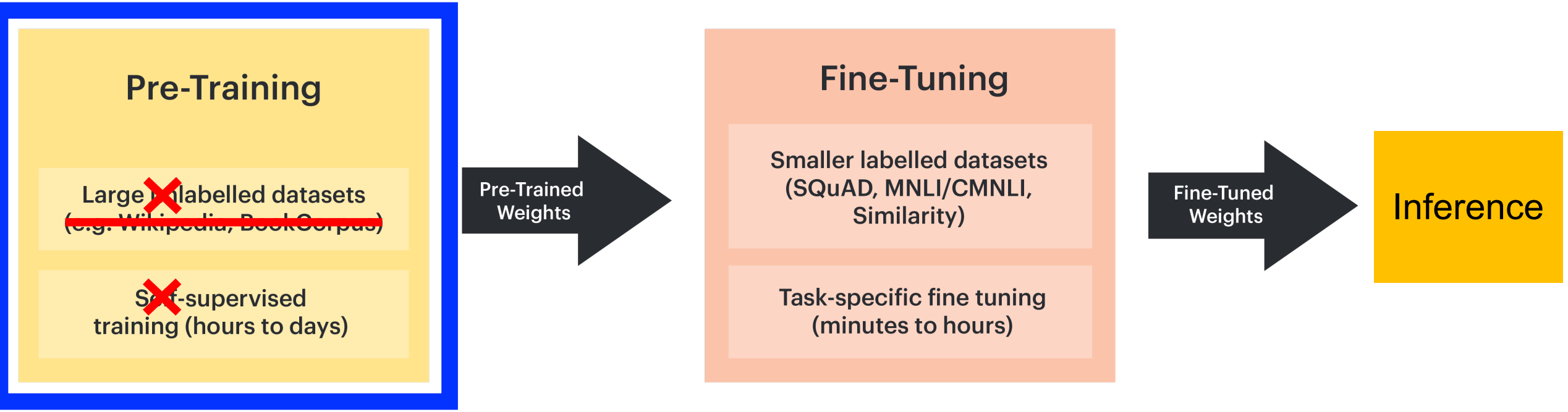


https://docs.graphcore.ai/projects/bert-training/en/latest/bert.html

Today’s Topics

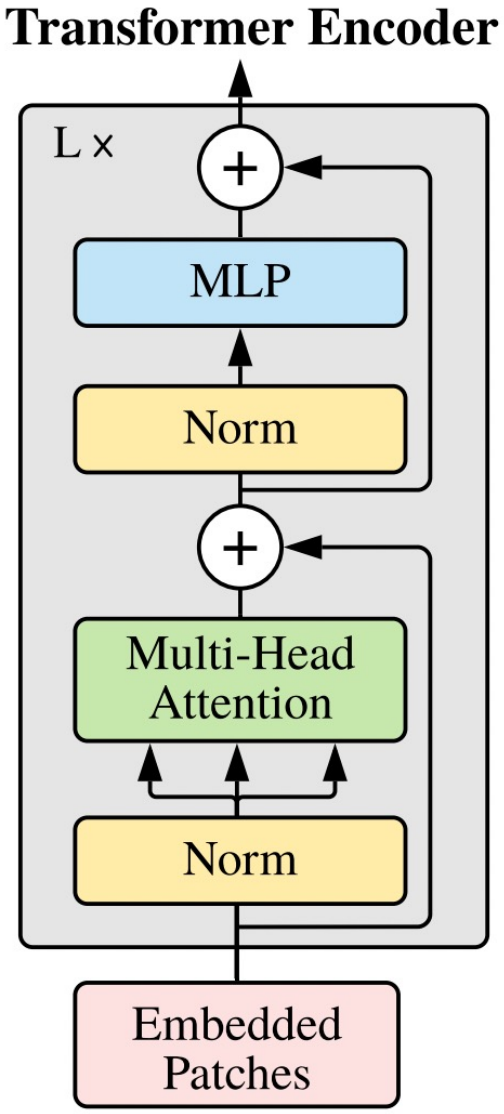
- Explosion of transformers
- GPT
- BERT
- ViT
- Programming tutorial

ViT: Vision Transformer



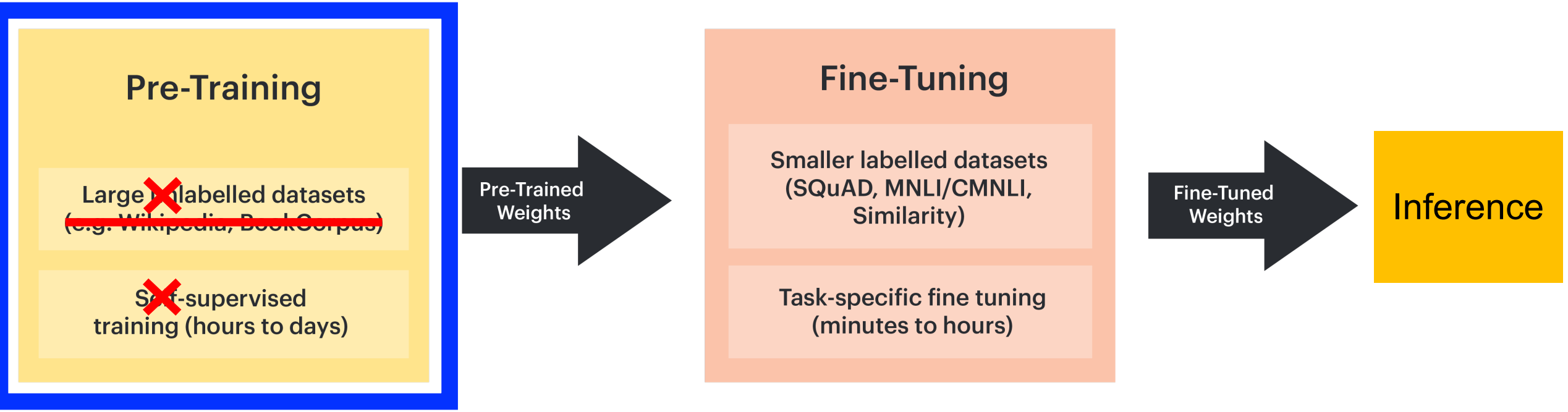
https://docs.graphcore.ai/projects/bert-training/en/latest/bert.html

Motivation: Transformers for Image Classification (Repurpose BERT)



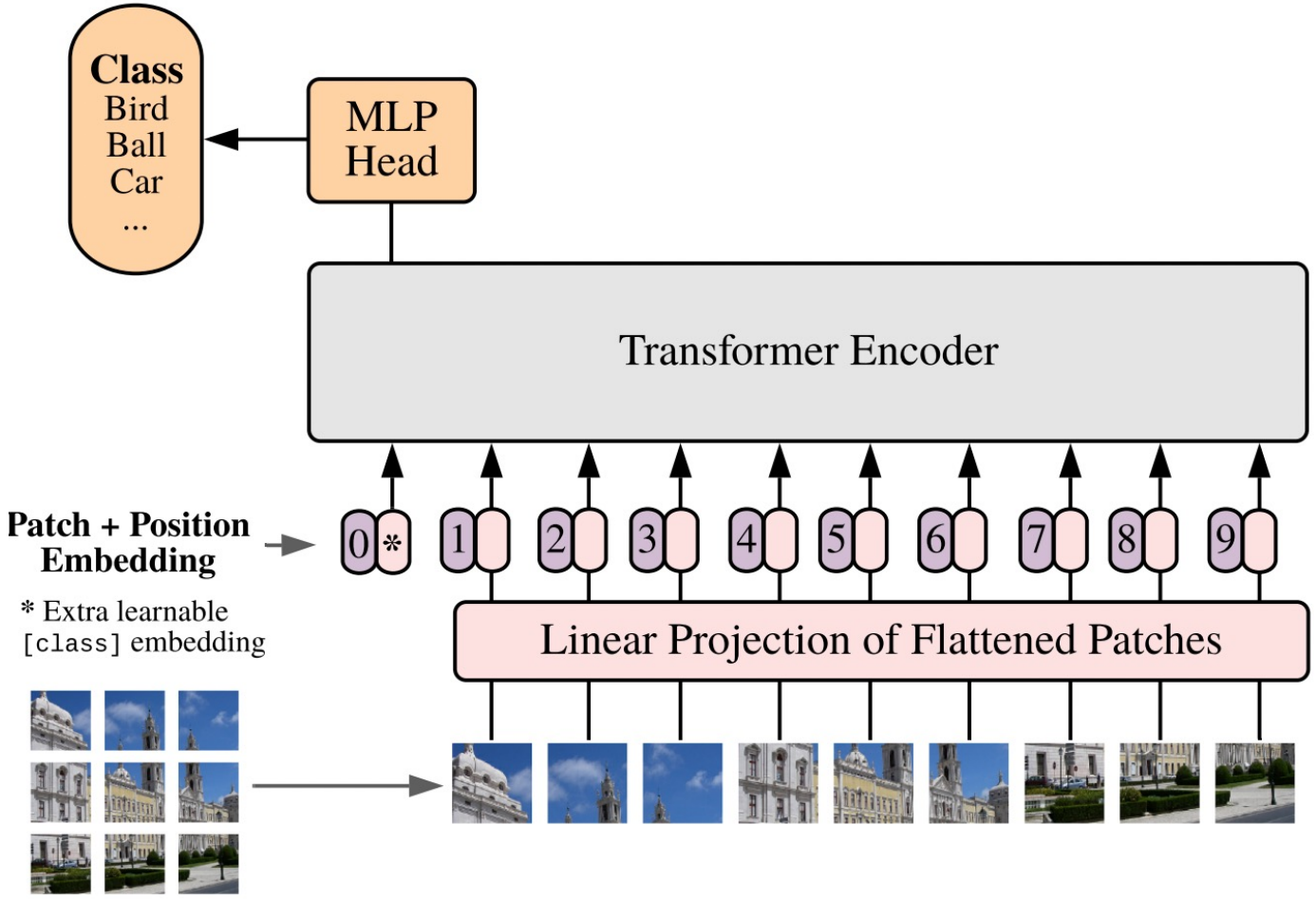
Dosovitskiy et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. ICLR 2021.

ViT: Vision Transformer



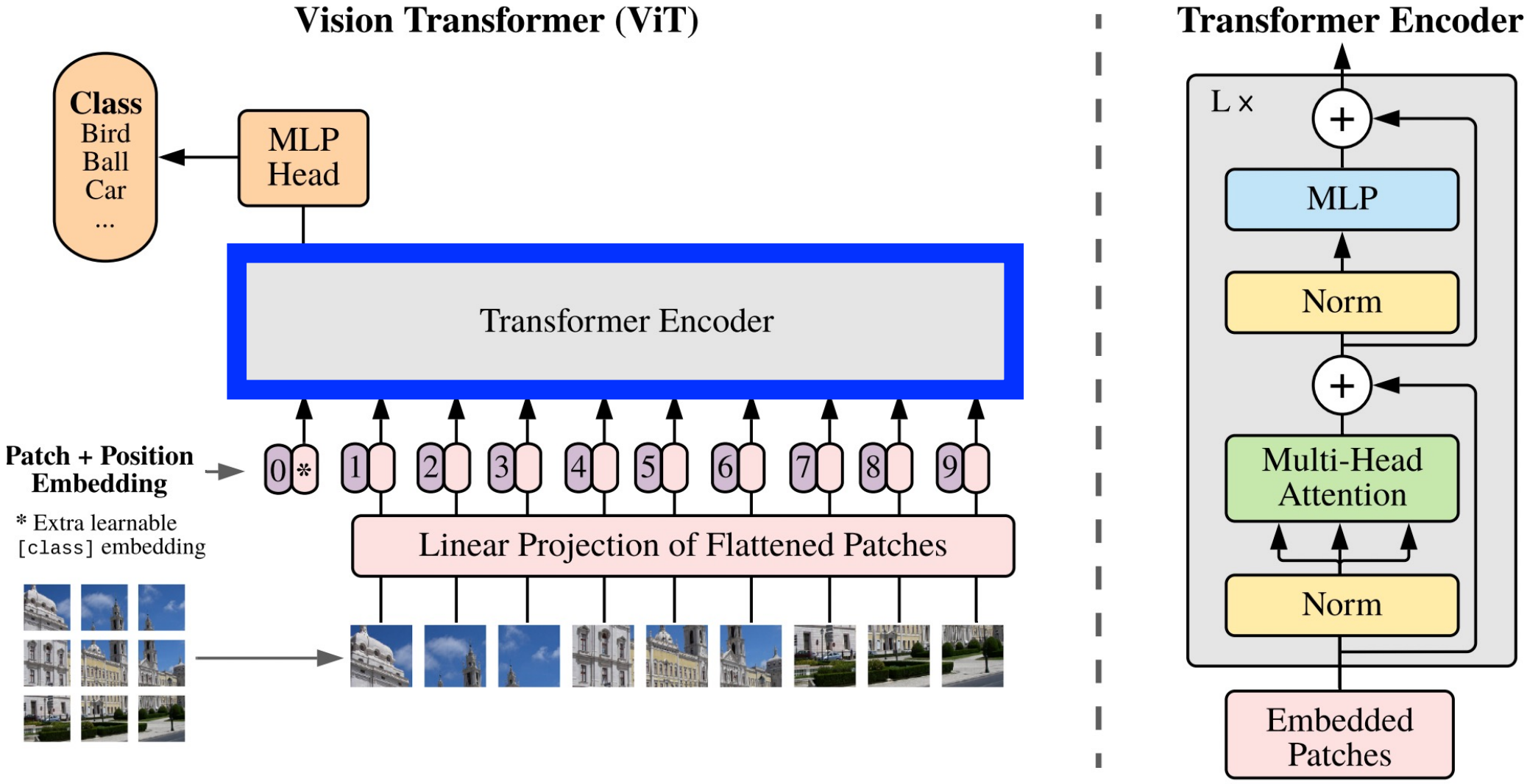
https://docs.graphcore.ai/projects/bert-training/en/latest/bert.html

Architecture



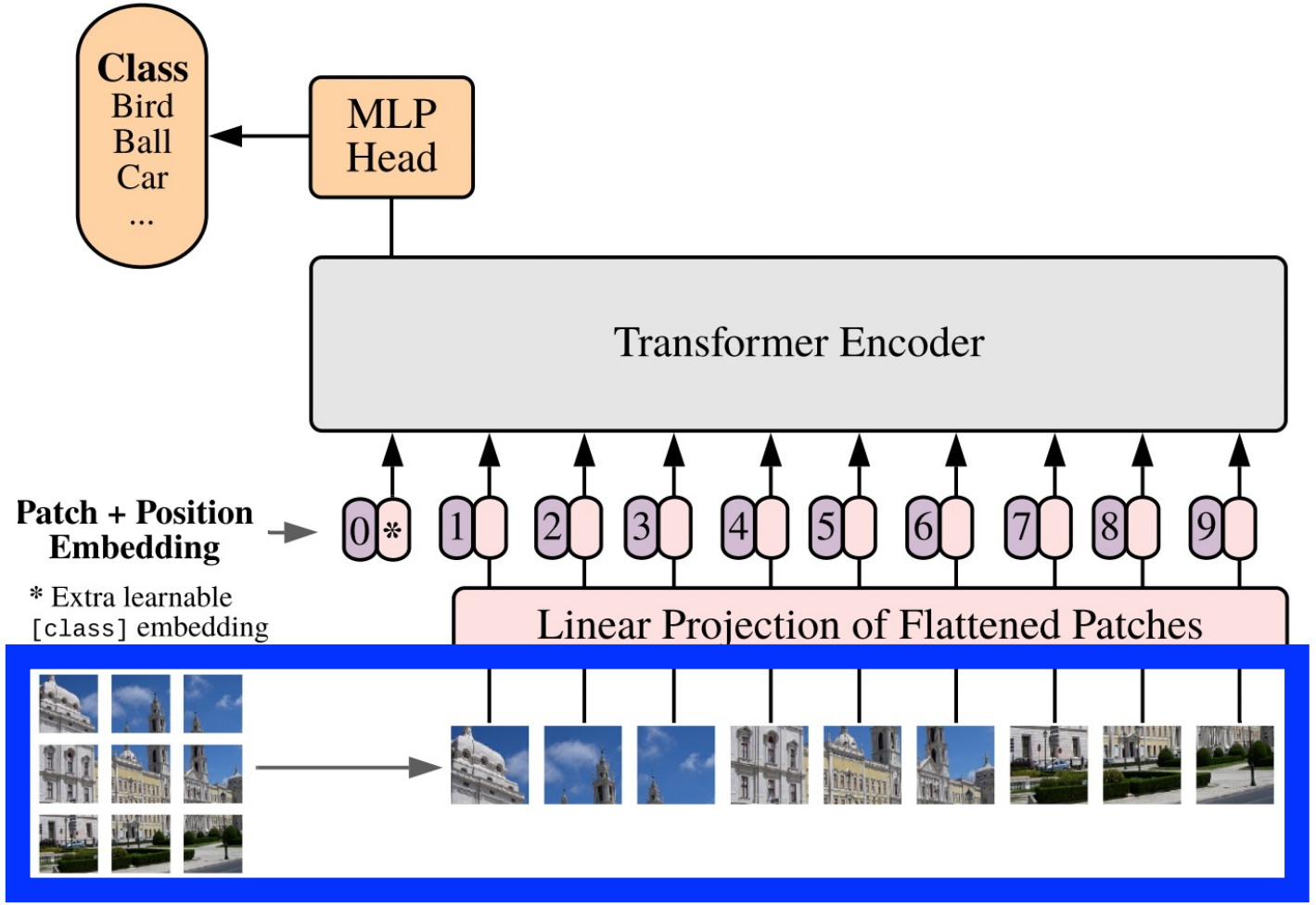
Dosovitskiy et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. ICLR 2021.

Architecture: BERT



Dosovitskiy et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. ICLR 2021.

Architecture: Input (Patches Instead of Pixels)

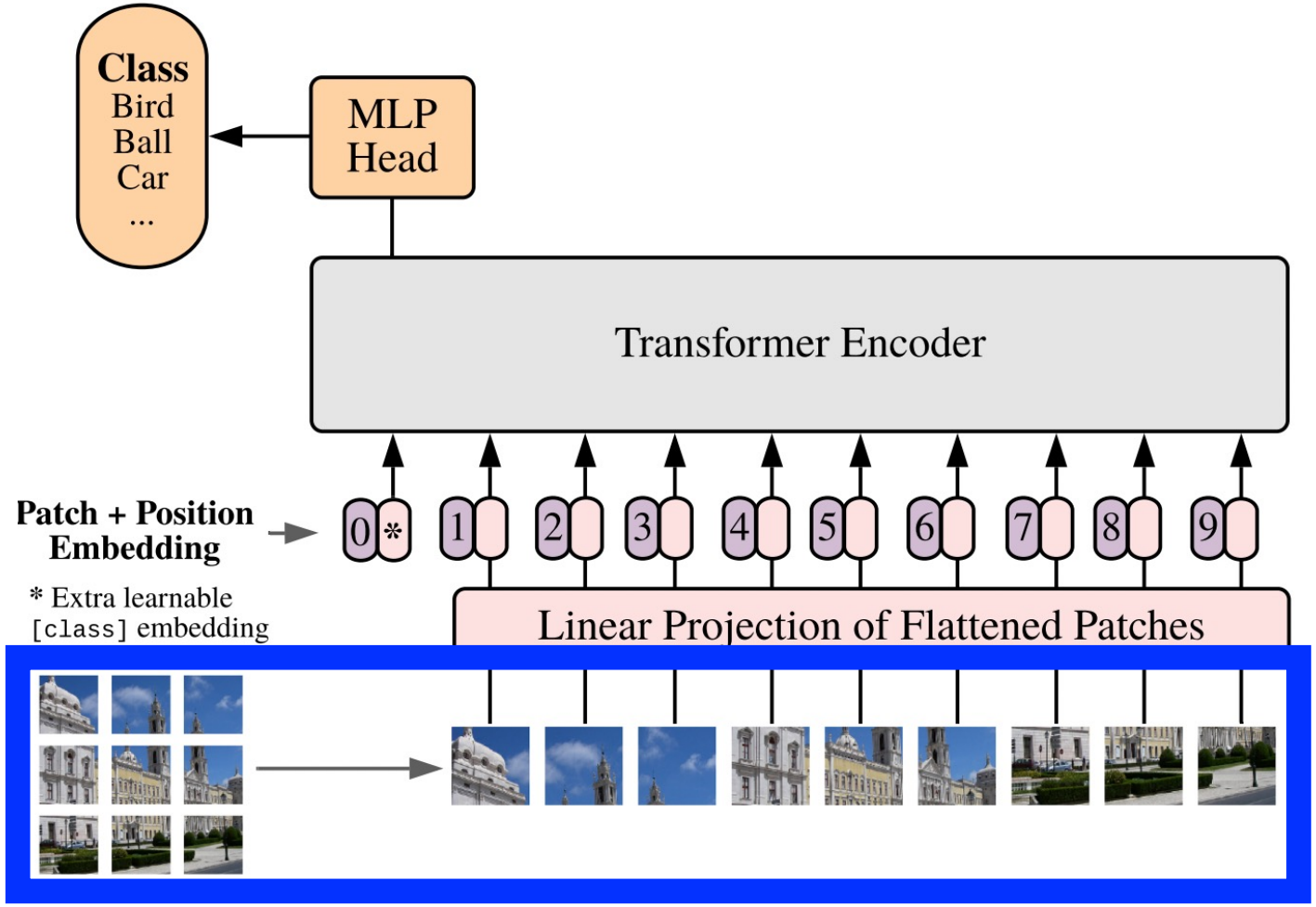


Assuming a 160 x 160 pixel image, how many patches (and so inputs) would be extracted?
- 100

image decomposed into 16x16 patches (example simplified); representations include "flattened" and ResNet features

Dosovitskiy et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. ICLR 2021.

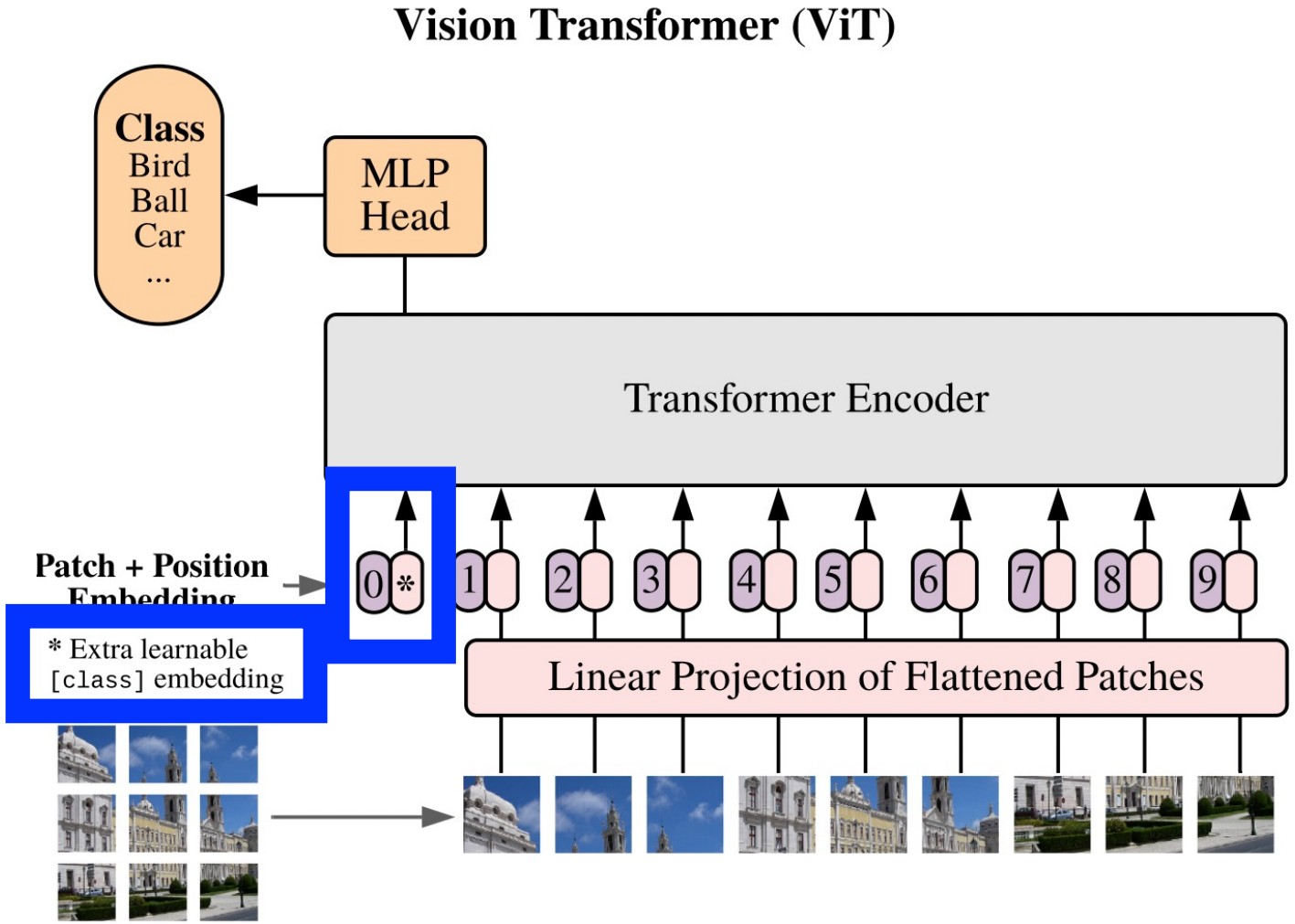
Architecture: Input (Patches Instead of Pixels)



Why not use the raw pixels as input?

Dosovitskiy et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. ICLR 2021.

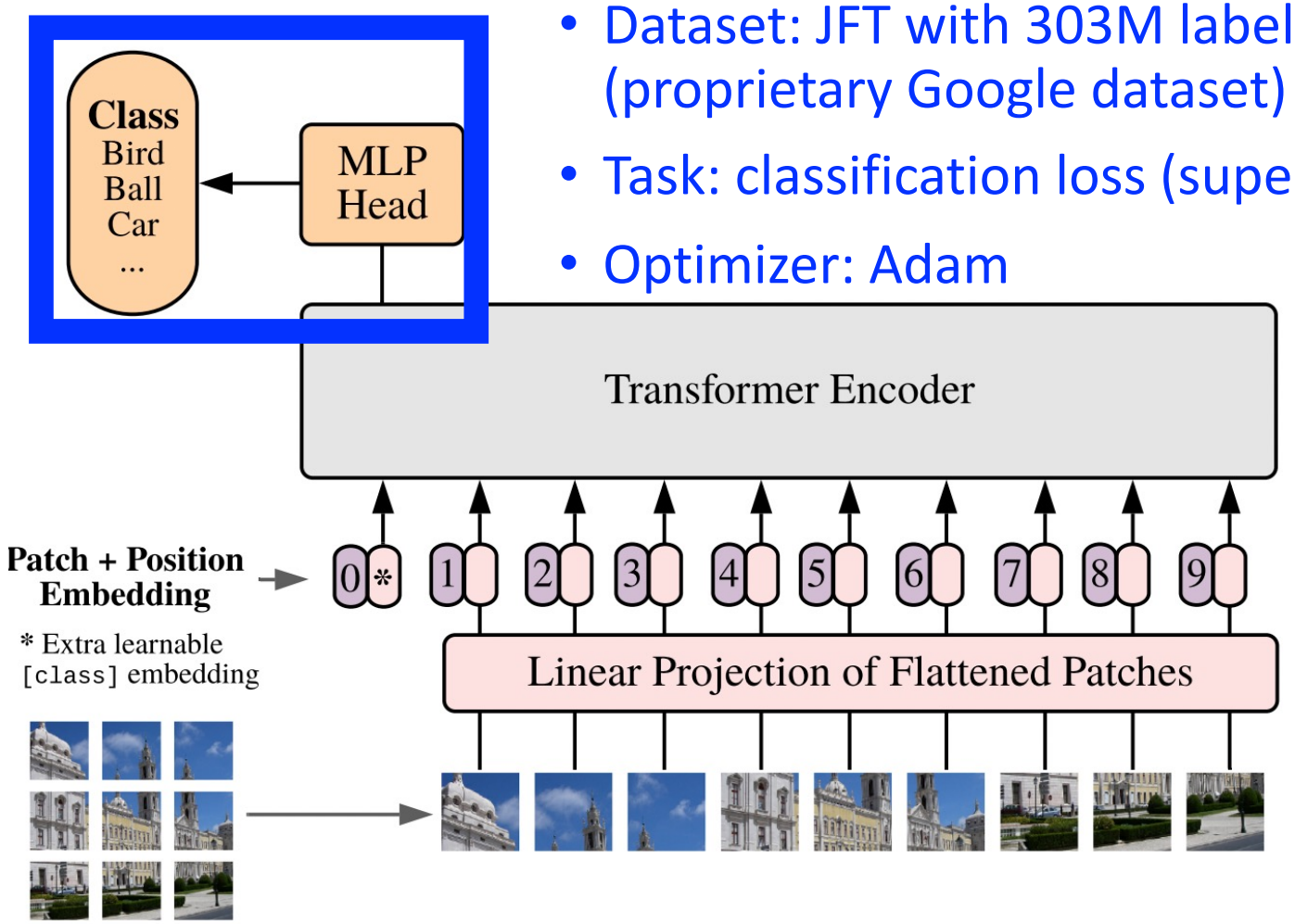
Architecture: Input CLS Token



[CLS] token represents entire image

Dosovitskiy et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. ICLR 2021.

Architecture: Training

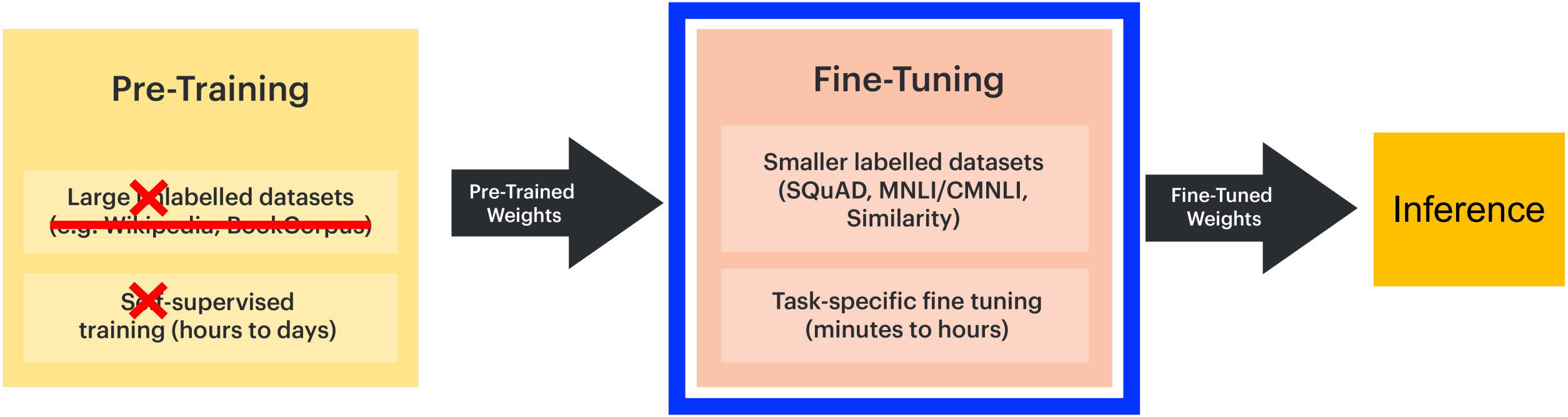


- Dataset: JFT with 303M labeled images (proprietary Google dataset)
- Task: classification loss (supervised)
- Optimizer: Adam

* Note: research also is exploring how smaller training datasets can be effective; e.g., data efficient image transformers (DeiT) from "Training data-efficient image transformers & distillation through attention"

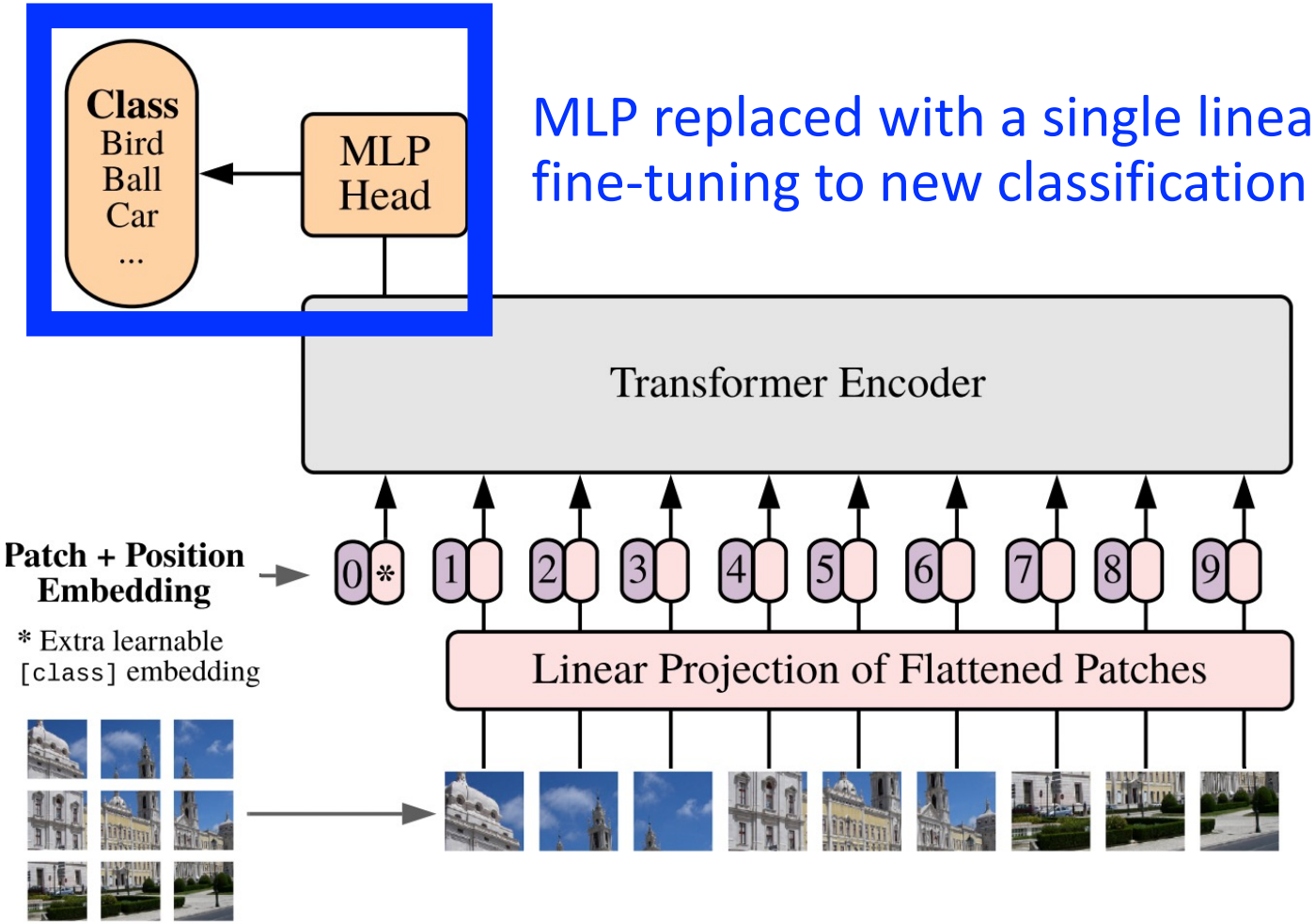
Dosovitskiy et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. ICLR 2021.

ViT: Vision Transformer



https://docs.graphcore.ai/projects/bert-training/en/latest/bert.html

Fine-Tuning for Other Image Classification Tasks



MLP replaced with a single linear layer when fine-tuning to new classification categories

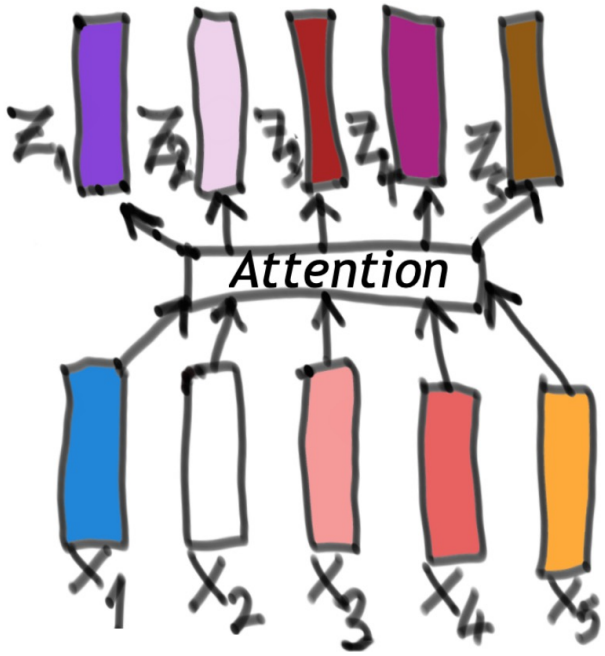
Experimental Findings

Achieved strong results on five image classification datasets

Devlin et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv 2018.

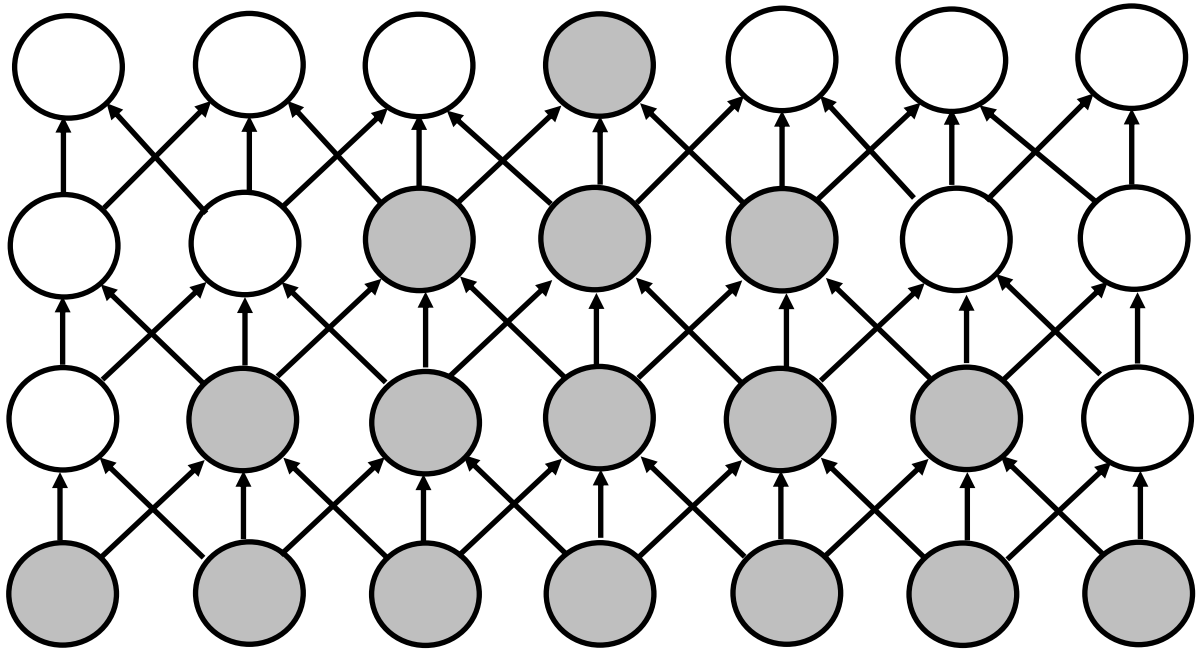
Transformers vs CNNs

Self-attention: each layer has a global receptive field



https://towardsdatascience.com/self-attention-5b95ea164f61

Convolutional layers: deeper layers have increasingly more global receptive fields



https://www.deeplearningbook.org/contents/convnets.html

Today’s Topics

- Explosion of transformers
- GPT
- BERT
- ViT
- Programming tutorial

Today’s Topics

- Explosion of transformers
- GPT
- BERT
- ViT
- Programming tutorial

The End