BERT: PRE-TRAINING OF DEEP BIDIRECTIONAL TRANSFORMERS FOR LANGUAGE UNDERSTANDING

Jacob Devlin **Ming-Wei Chang** **Kenton Lee** **Kristina Toutanova**

Google AI Language

{jacobdevlin, mingweichang, kentonl, kristout}@google.com

Presented by **Nikita Nikolaev**

I. INTRODUCTION

I. INTRODUCTION

I. INTRODUCTION

BERT

Bidirectional **E**ncoder **R**epresentations from **T**ransformers

II. RELATED WORKS

II. RELATED WORKS

- The feature-based approach, such as ELMo (Peters et al., 2018a)

II. RELATED WORKS

- The feature-based approach, such as ELMo (Peters et al., 2018a)
- The fine-tuning approach, such as the Generative Pre-trained Transformer (OpenAI GPT) (Radford et al., 2018)

III. BERT

III. BERT

2 steps in the framework:

Pre-training

Fine-tuning

III. BERT: MODEL ARCHITECTURE

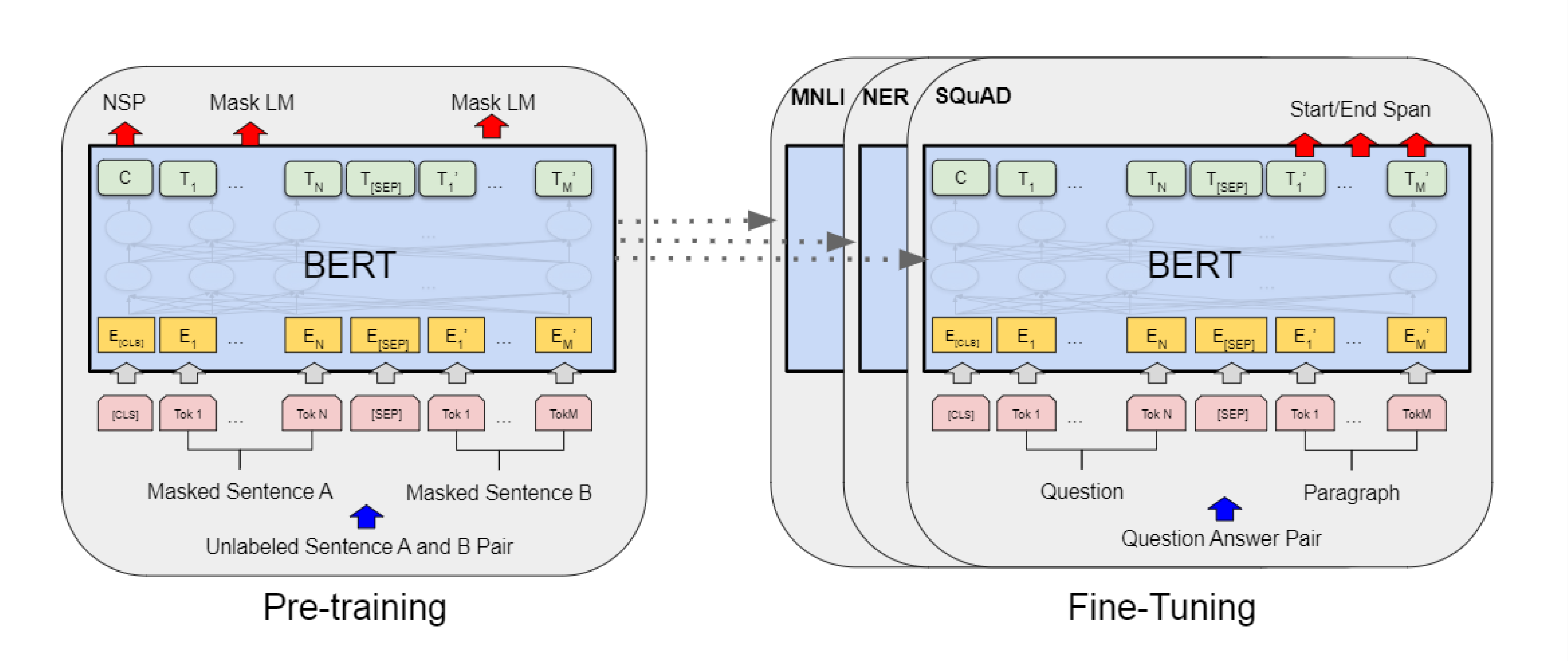
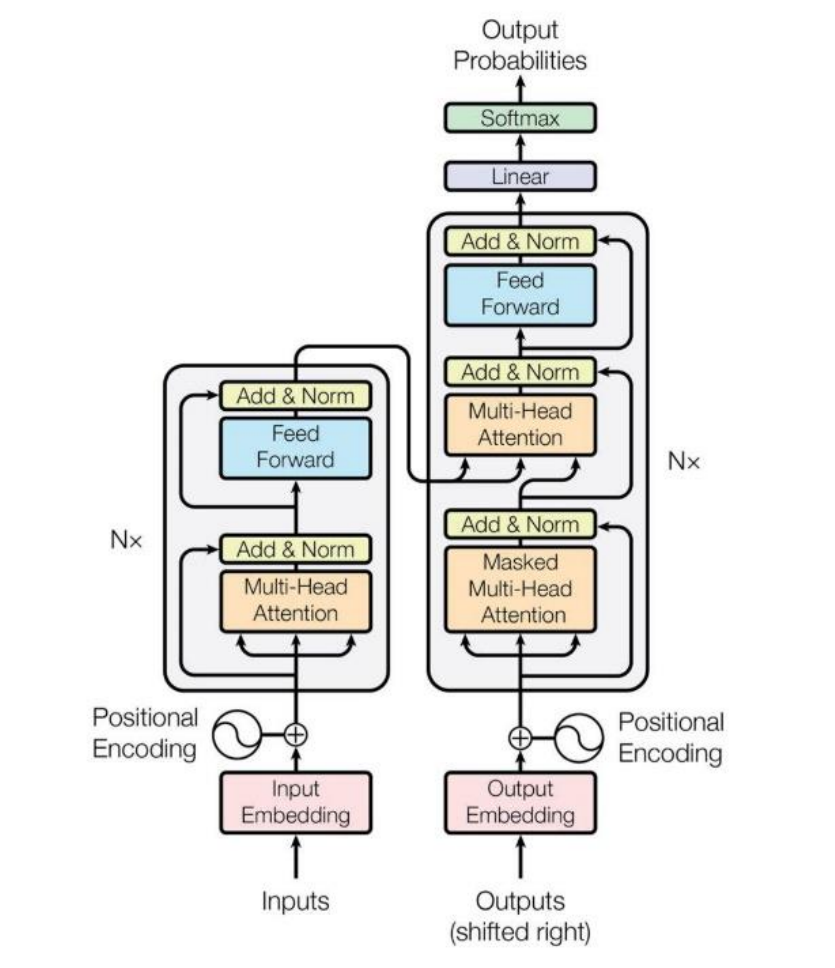


Figure 1: Overall pre-training and fine-tuning procedures for BERT. Apart from output layers, the same architectures are used in both pre-training and fine-tuning. The same pre-trained model parameters are used to initialize models for different down-stream tasks.

III. BERT: MODEL ARCHITECTURE TRANSFORMERS

Figure 2: The Transrormer – model architecture

From 'Attention Is All You Need' by Vaswani et al.



III. BERT: MODEL ARCHITECTURE INPUT REPRESENTATION

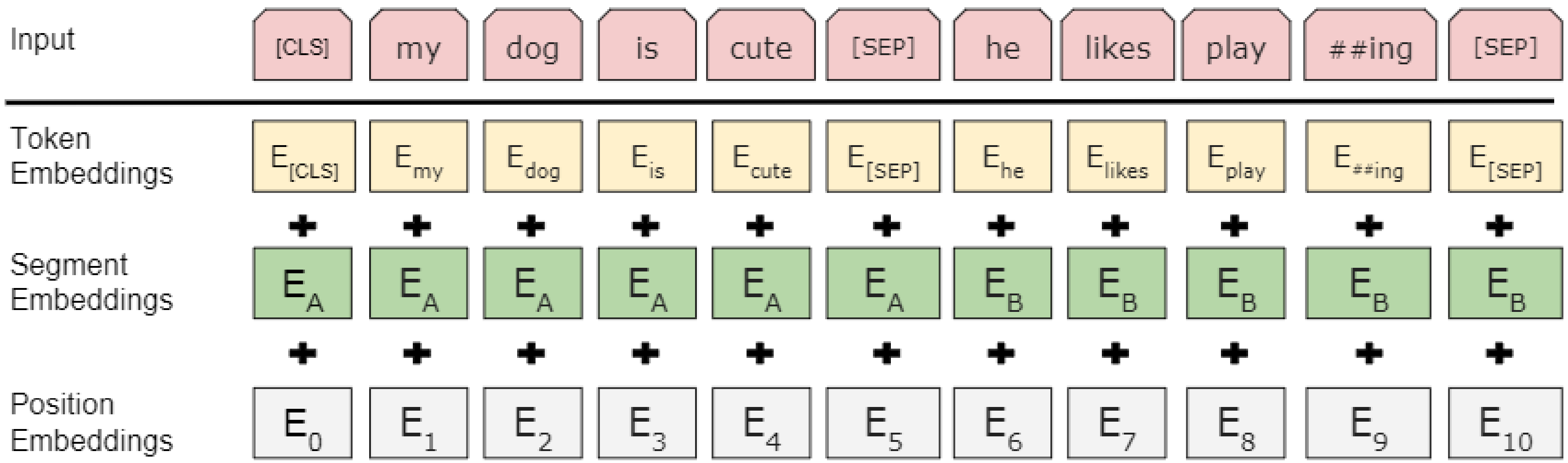
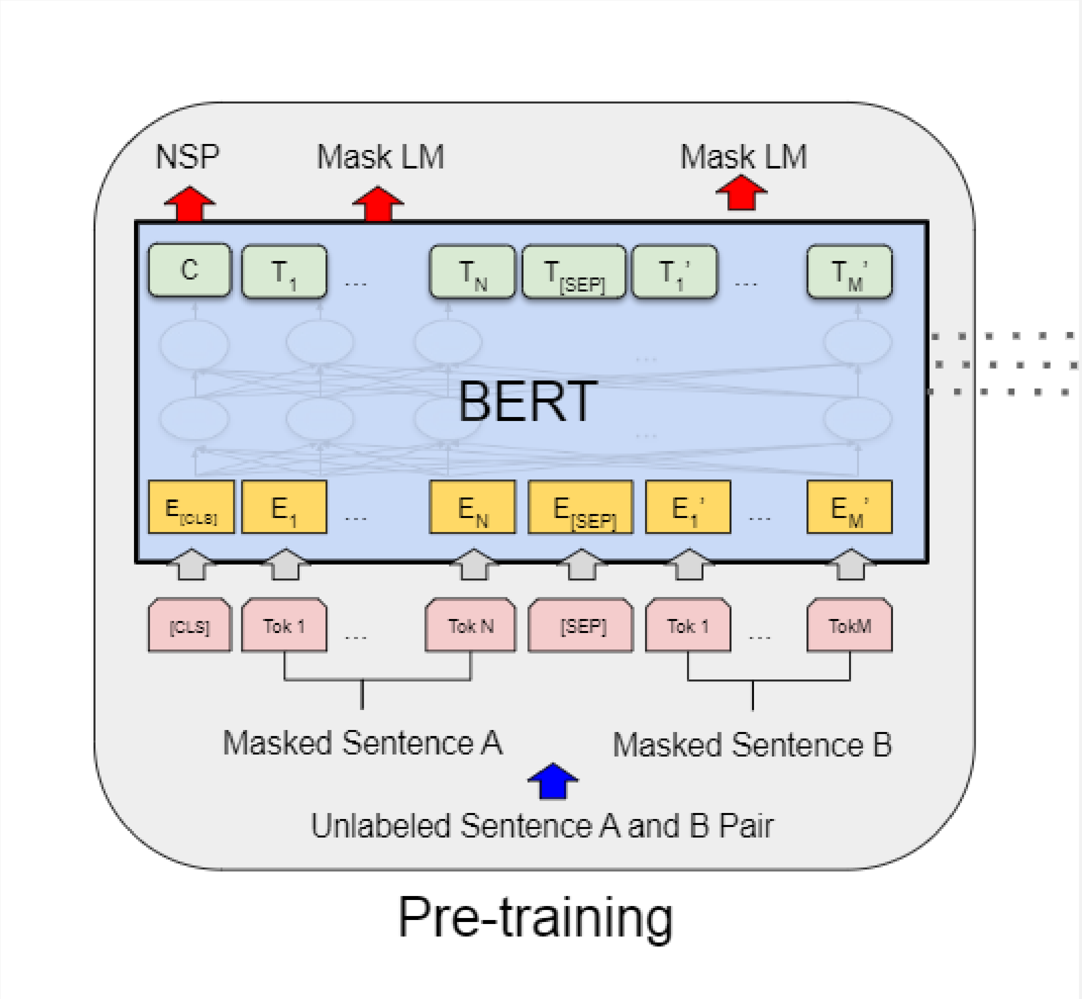


Figure 3: BERT input representation. The input embeddings are the sum of the token embeddings, the segmentation embeddings and the position embeddings.

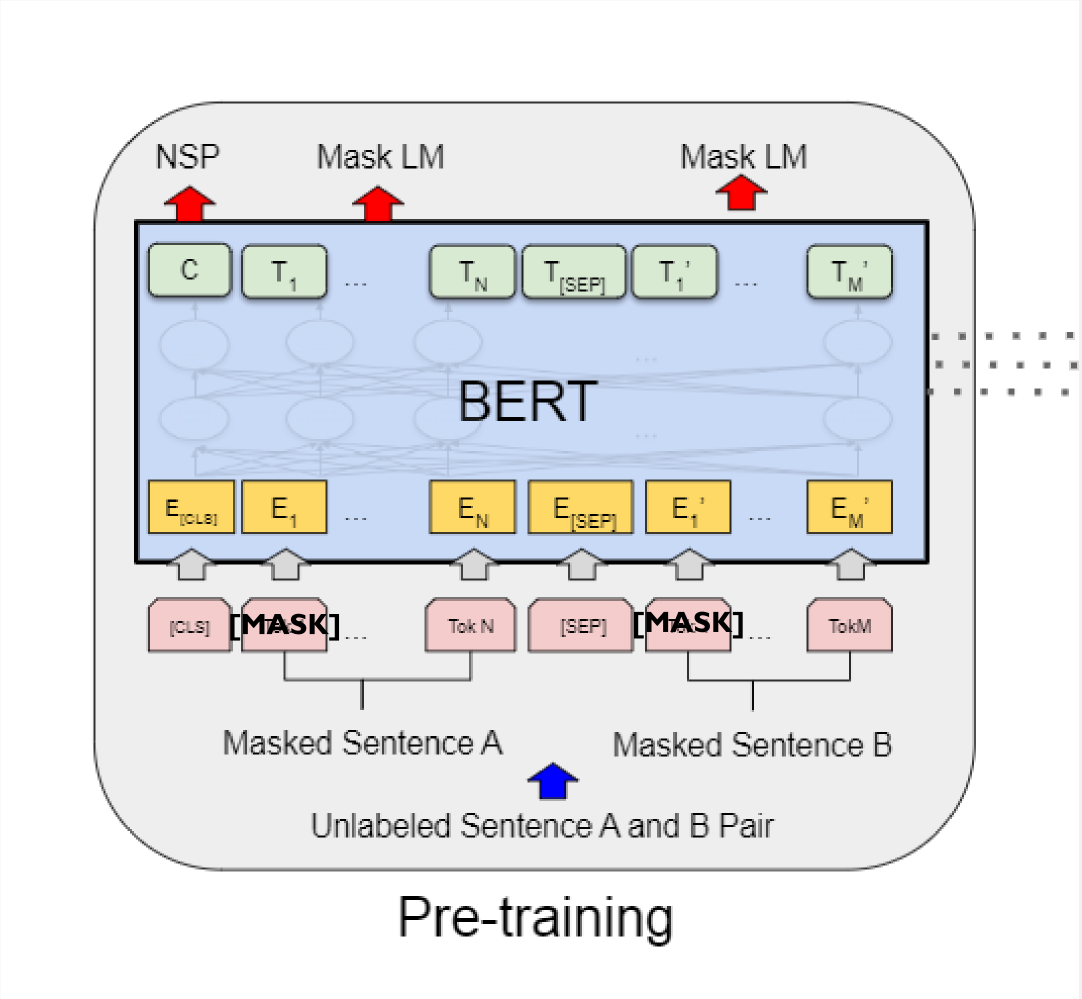
III. BERT: MODEL ARCHITECTURE

Figure 4: Overall pre-training procedure for BERT.



III. BERT: MODEL ARCHITECTURE MASKED LM

Figure 4: Overall pre-training procedure for BERT.



III. BERT: MODEL ARCHITECTURE
NEXT SENTENCE PREDICTION

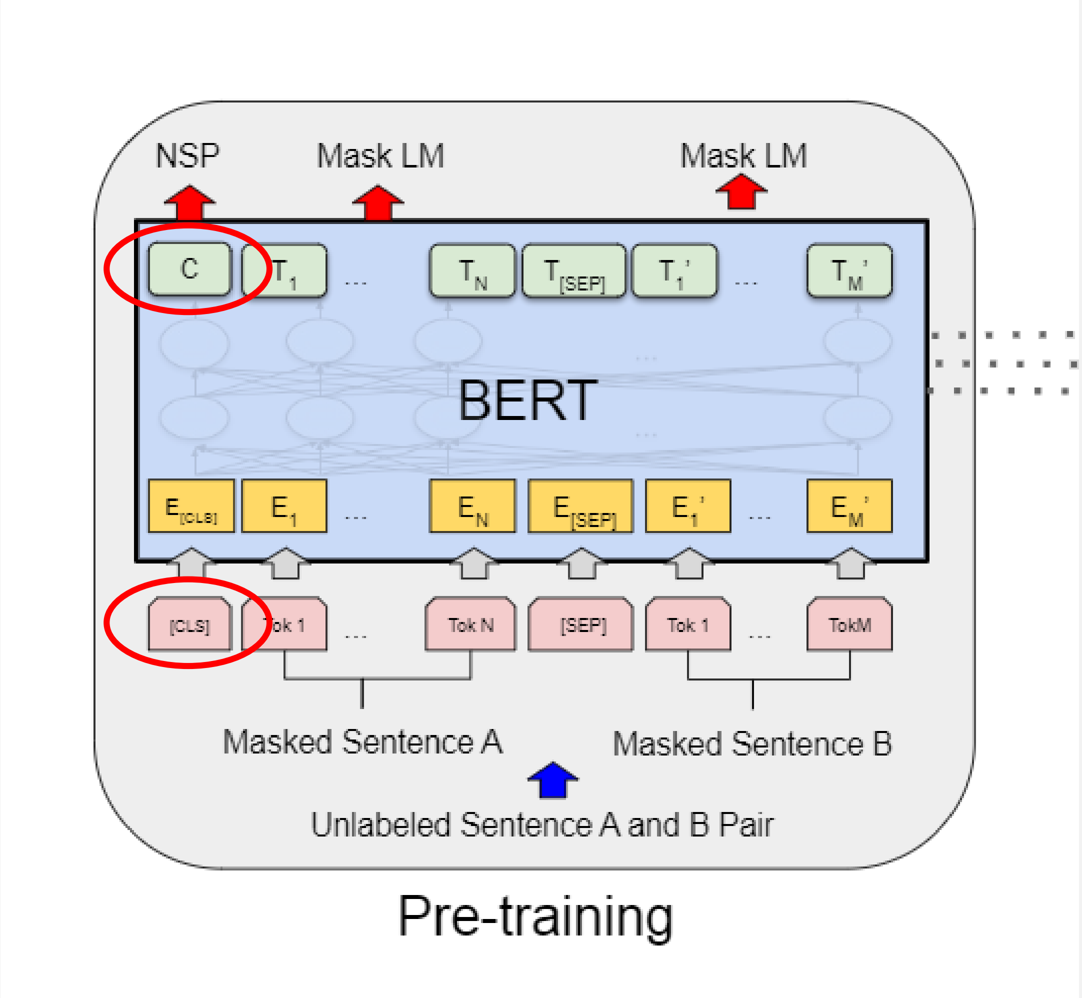


Figure 4: Overall pre-training procedure for BERT.

IV. PRE-TRAINING DATA

- BooksCorpus (800M words) (Zhu et al.,2015)
- English Wikipedia (2,500M words)

IV. BERT: FINE-TUNING

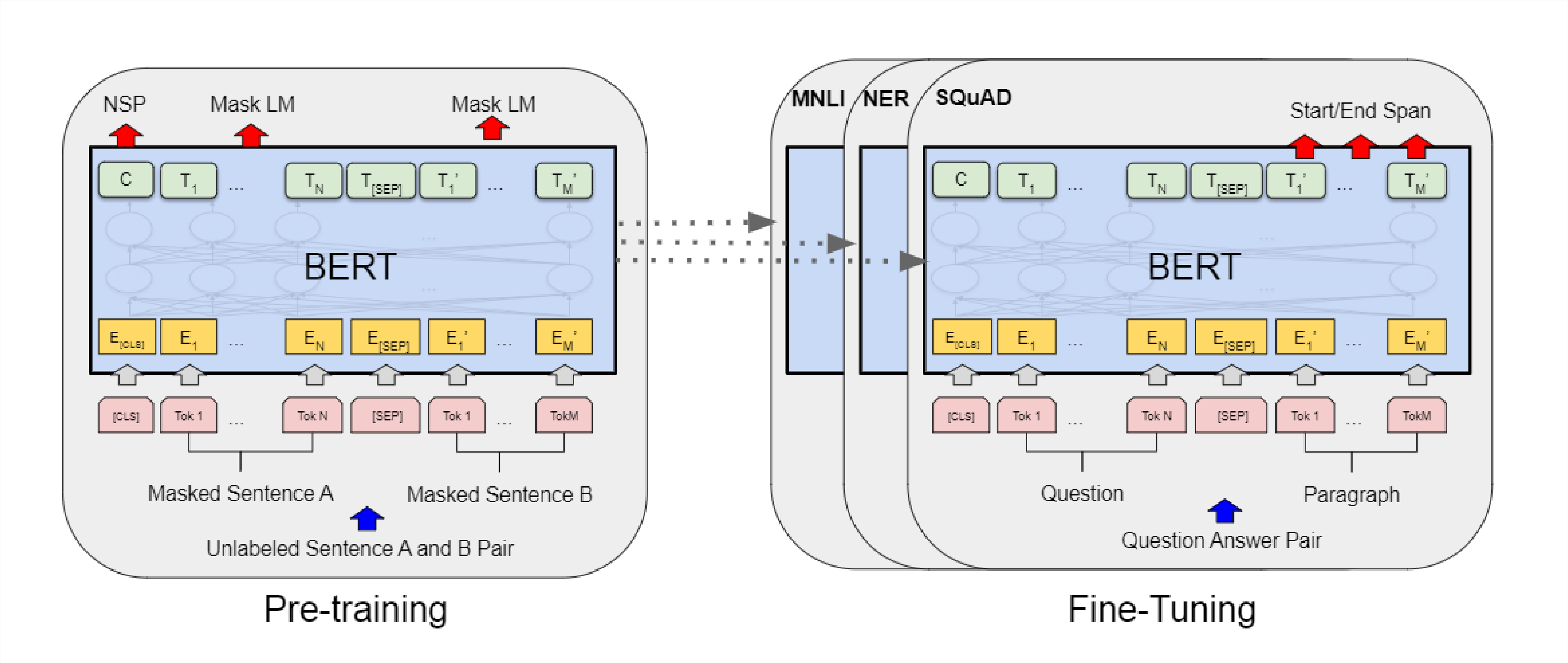


Figure 1: Overall pre-training and fine-tuning procedures for BERT. Apart from output layers, the same architectures are used in both pre-training and fine-tuning. The same pre-trained model parameters are used to initialize models for different down-stream tasks.

V. RESULTS

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	79.8	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.

THANK YOU FOR YOUR ATTENTION

~~THANK YOU FOR YOUR~~
ATTENTION IS ALL YOU NEED