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

BY JACOB DEVLIN, MING-WEI CHANG, KENTON LEE, KRISTINA TOUTANOVA

Presentation by René Caky

BERT

- **B**idirectional
- **E**ncoder
- **R**epresentations
- from
- **T**ransformers

CONTENT OF THE PRESENTATION

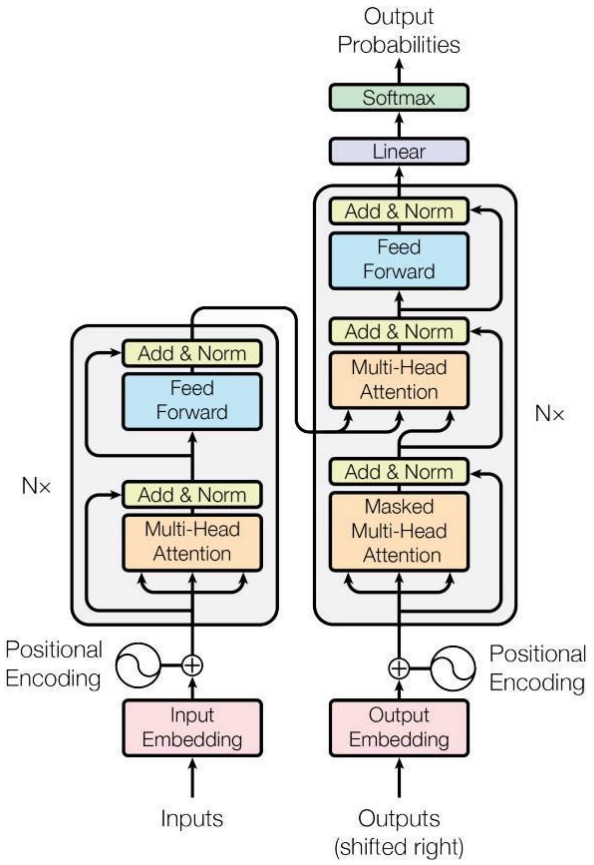
- Feature-based vs Fine-tuning LM in general
- Background knowledge
- BERT training
- Results
- Contribution
- Opinion
- Related Work

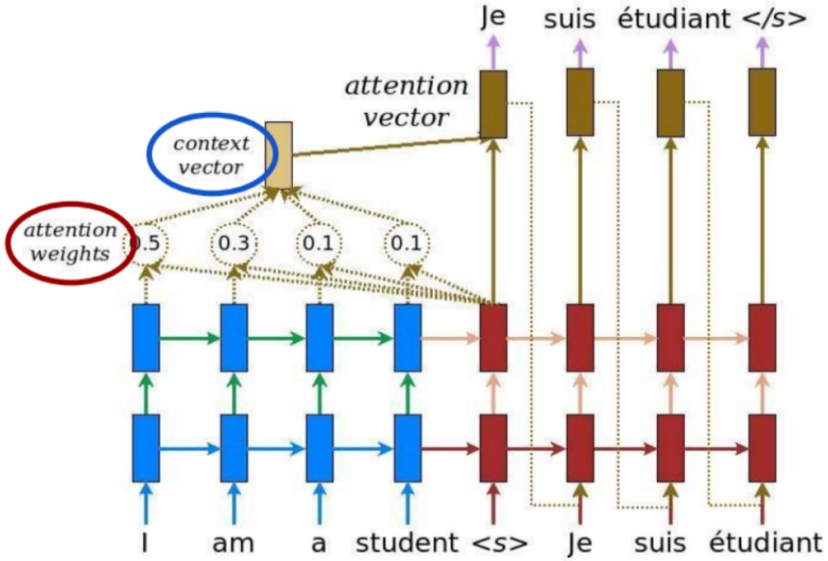
FEATURE-BASED

- Pre-trained model
- Fixed features are extracted from the pretrained model
- Many different option of extracting the features

FINE-TUNING

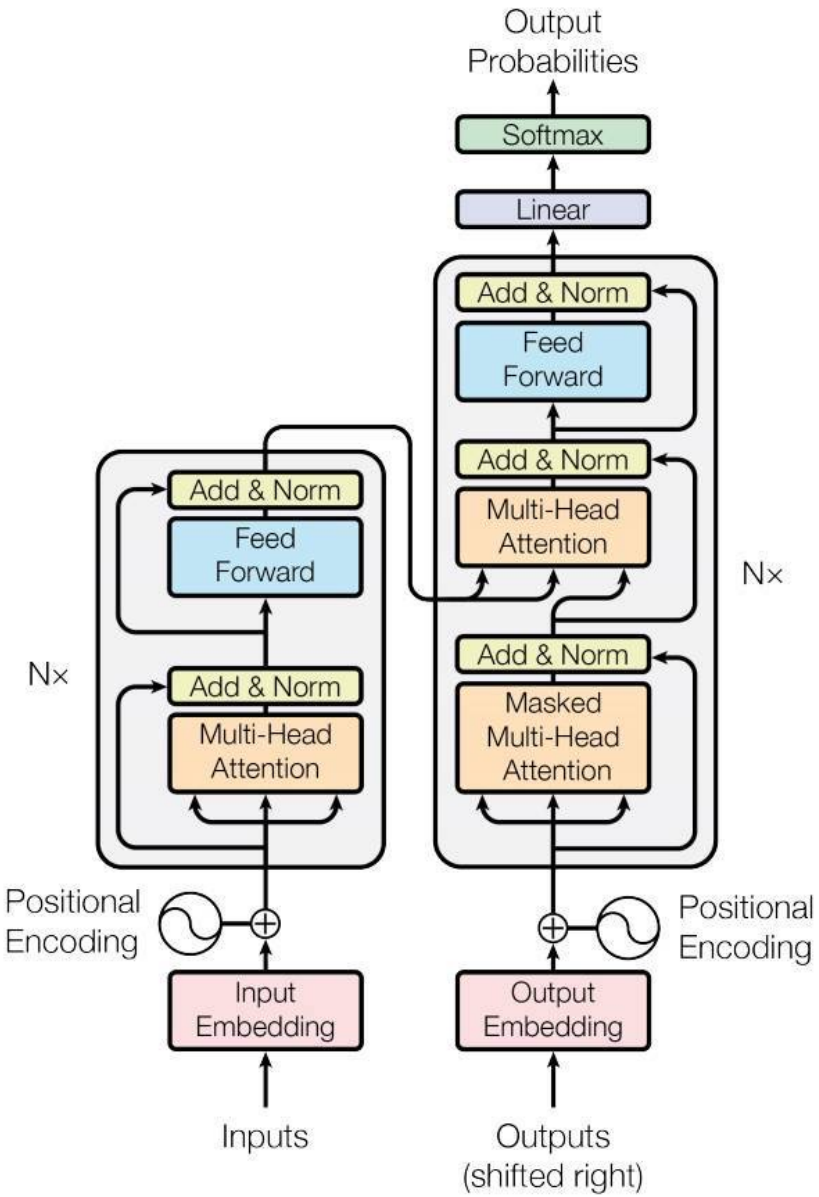
- Pre-trained model
- Add a layer
- Jointly train on a specific task

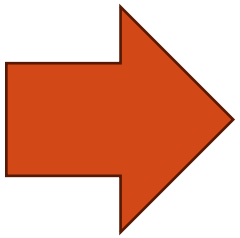


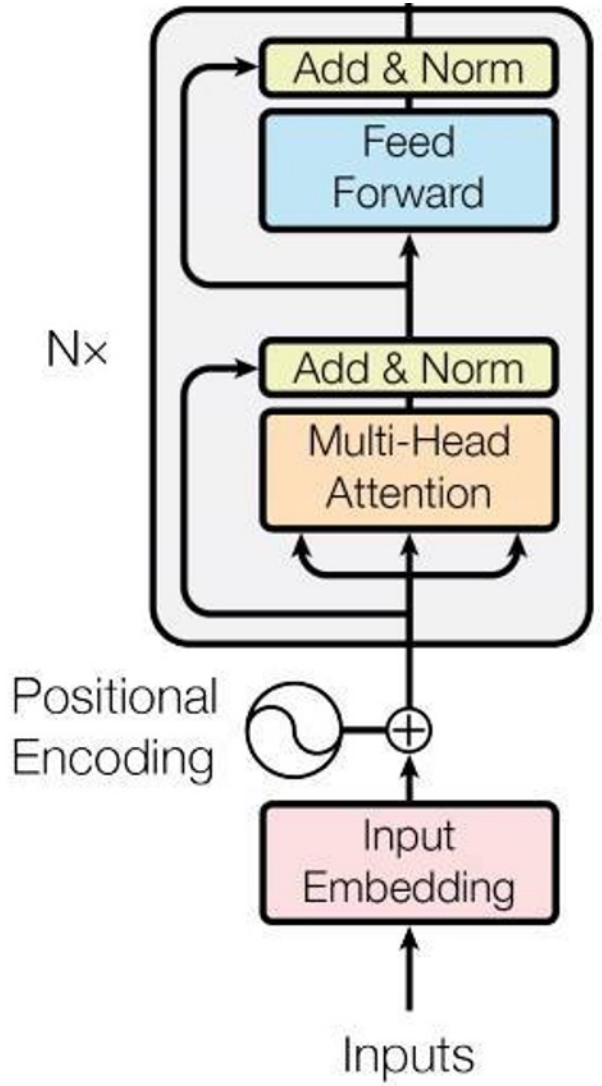


-Originally translation of sentences between languages
-Faster than LSTMs
-2 parts: Encoder and Decoder

BACKGROUND


Output Probabilities
Softmax
Linear
Add & Norm
Feed Forward
Add & Norm
Multi-Head Attention
Add & Norm
Masked Multi-Head Attention
Nx
Add & Norm
Feed Forward
Add & Norm
Multi-Head Attention
Nx
Positional Encoding
Positional Encoding
Input Embedding
Output Embedding
Inputs
Outputs (shifted right)



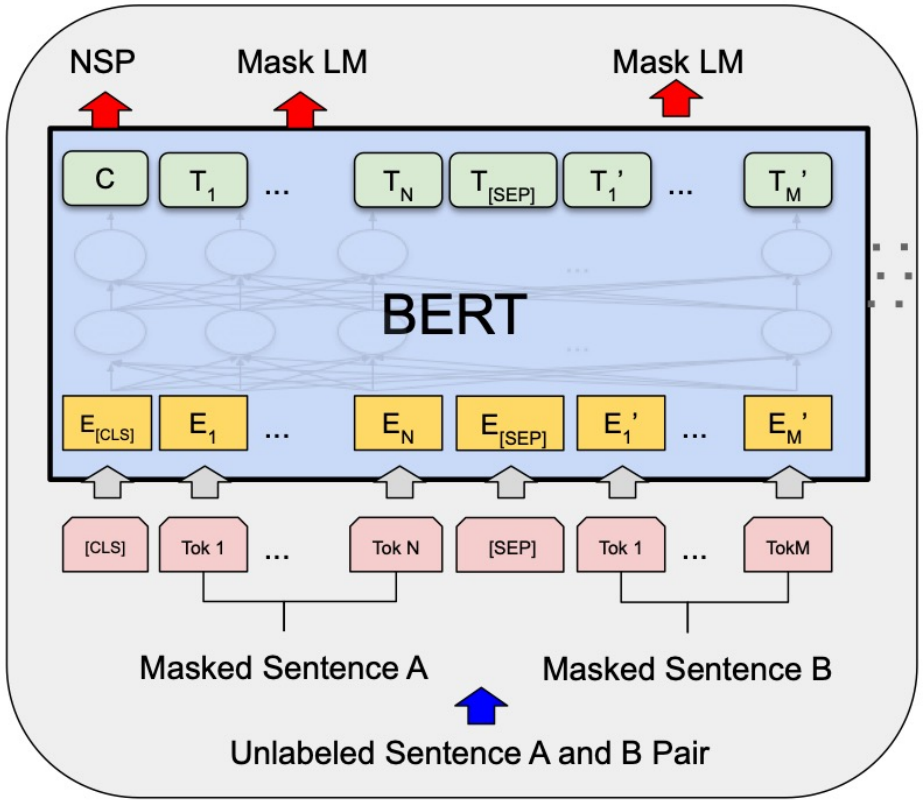

Add & Norm
Feed Forward
Nx
Add & Norm
Multi-Head Attention
Positional Encoding
Input Embedding
Inputs

2-PHASE BERT TRAINING

- 1.Pretraining to generally understand the language
- 2.Fine-tuning for a specific task:
  - Text Classification
  - Question Answering

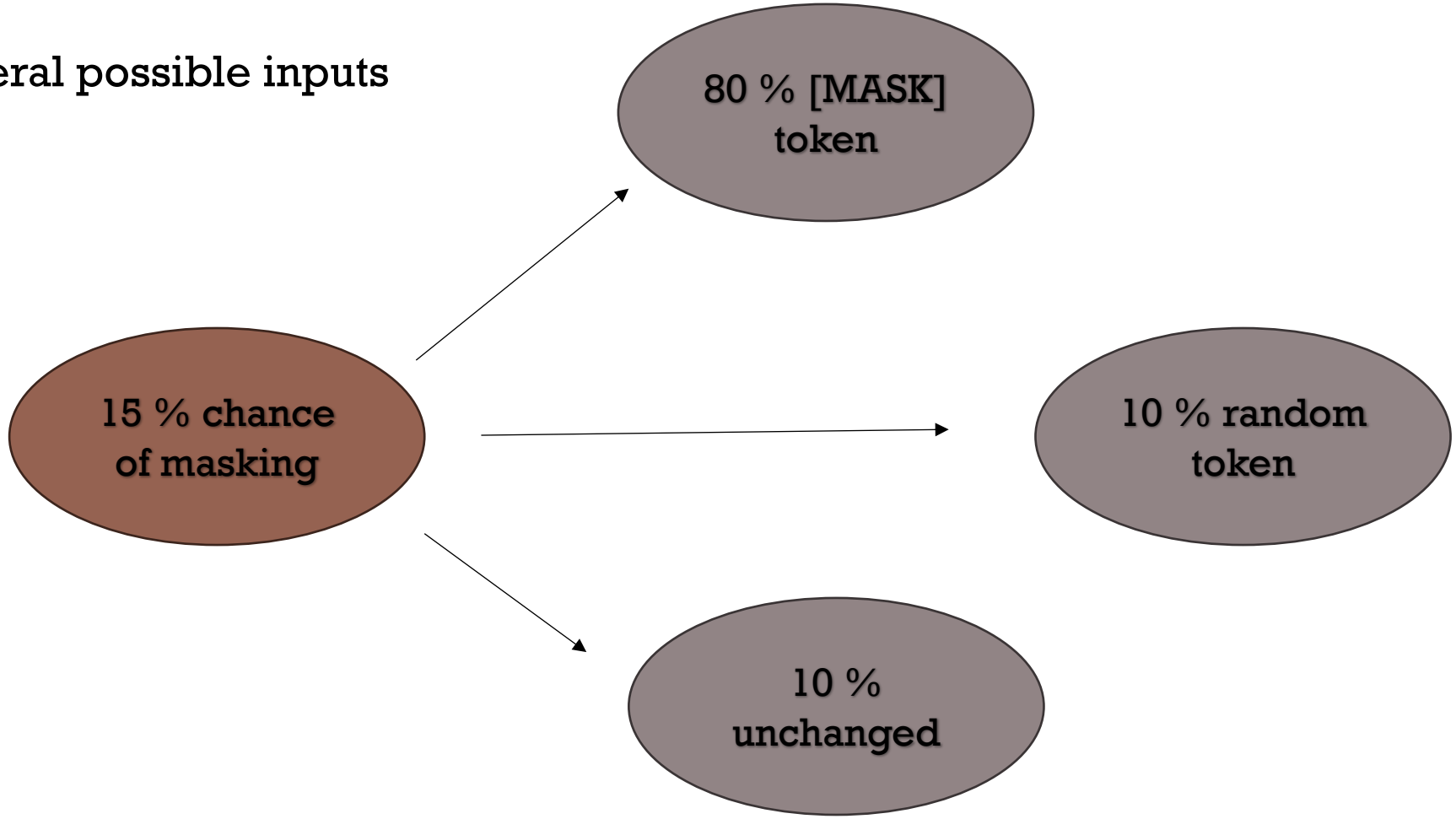
PRETRAINING BERT

- Masked Language Model (MLM)
- Next Sentence Prediction(NSP)

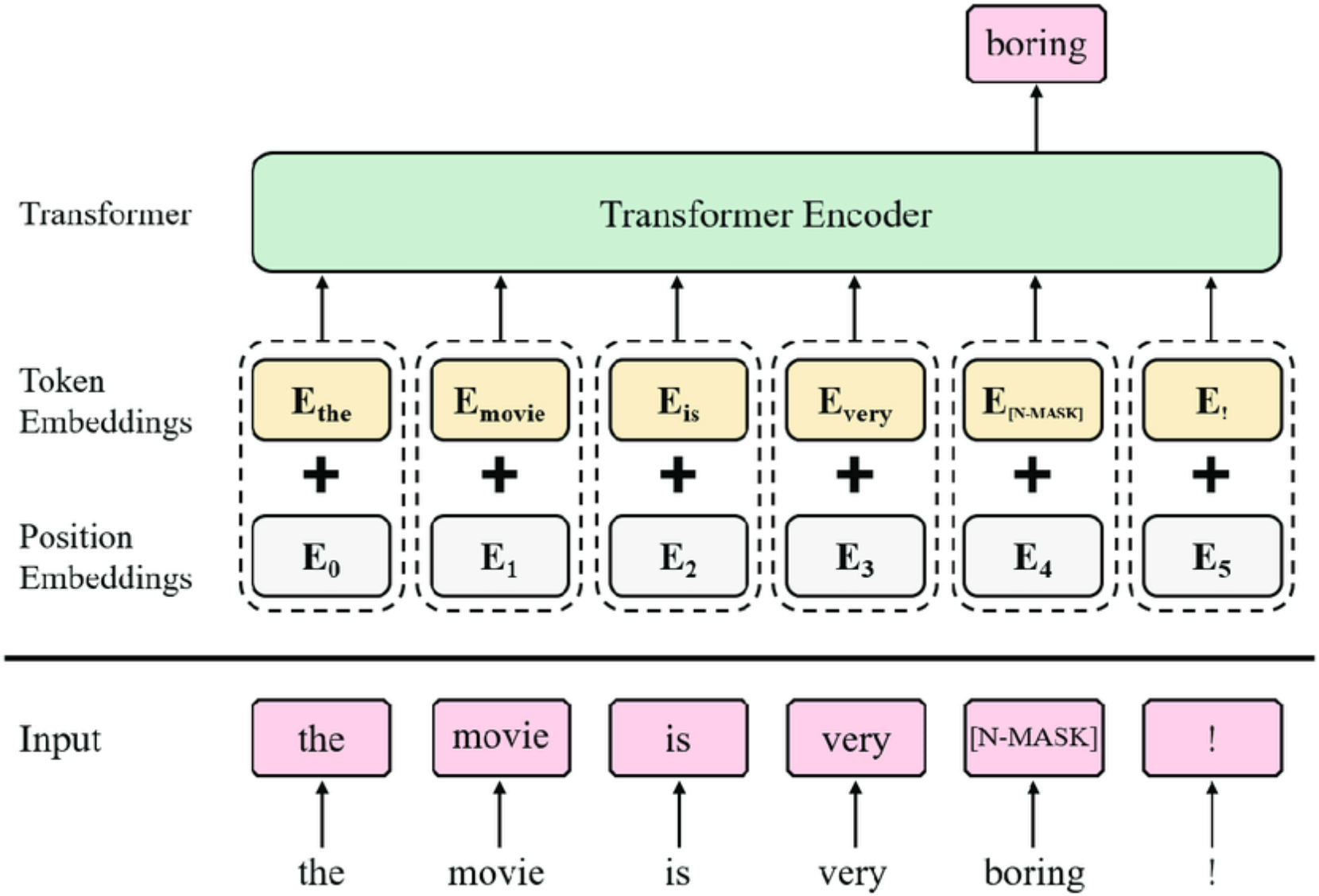


MLM

- Several possible inputs

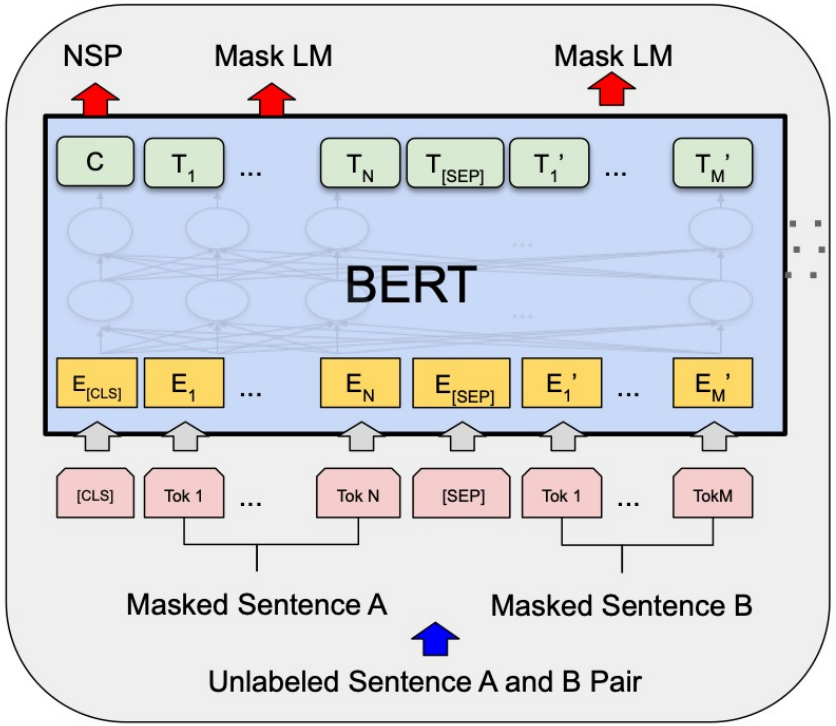


- Objective: output the predicted masked tokens (softmax)

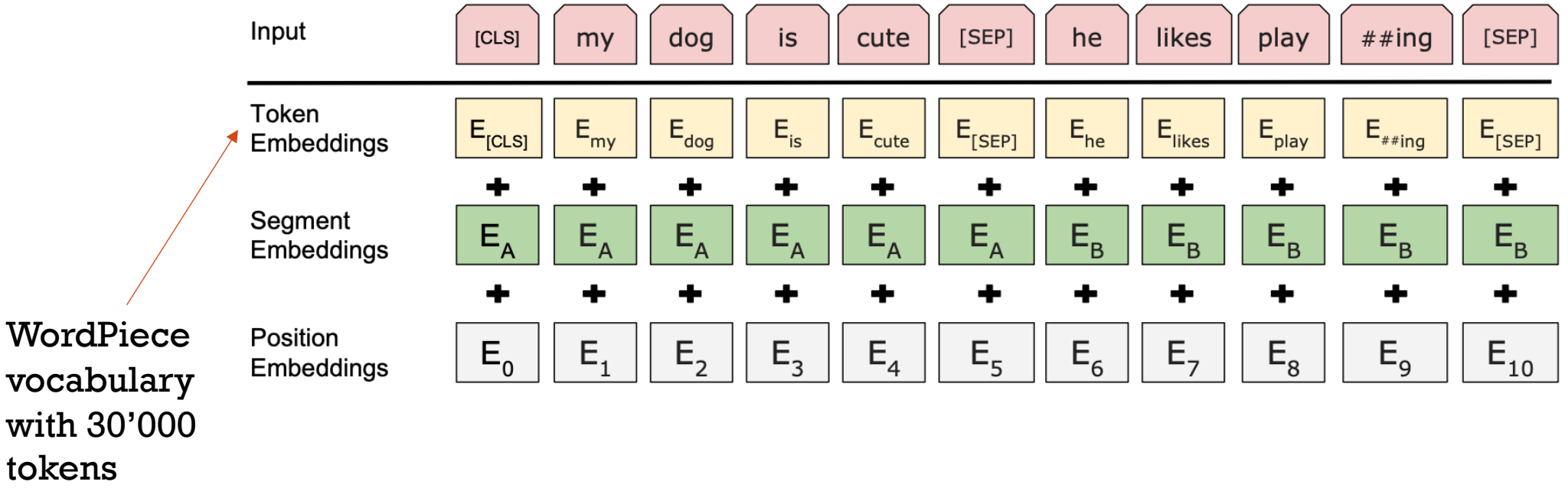

boring
Transformer
Transformer Encoder
Token Embeddings
Position Embeddings
E_{the}
E_{movie}
E_{is}
E_{very}
$E_{[N-MASK]}$
$E_{!}$
E_0
E_1
E_2
E_3
E_4
E_5
Input
the
movie
is
very
[N-MASK]
!
the
movie
is
very
boring
!

NSP

- Input: 2 sentences A and B
- 50 % : B is a sentence that actually follows A
- 50 % : B is a sentence that doesn't follow A
- Objective: Create a single embedding that captures the meaning of a sentence



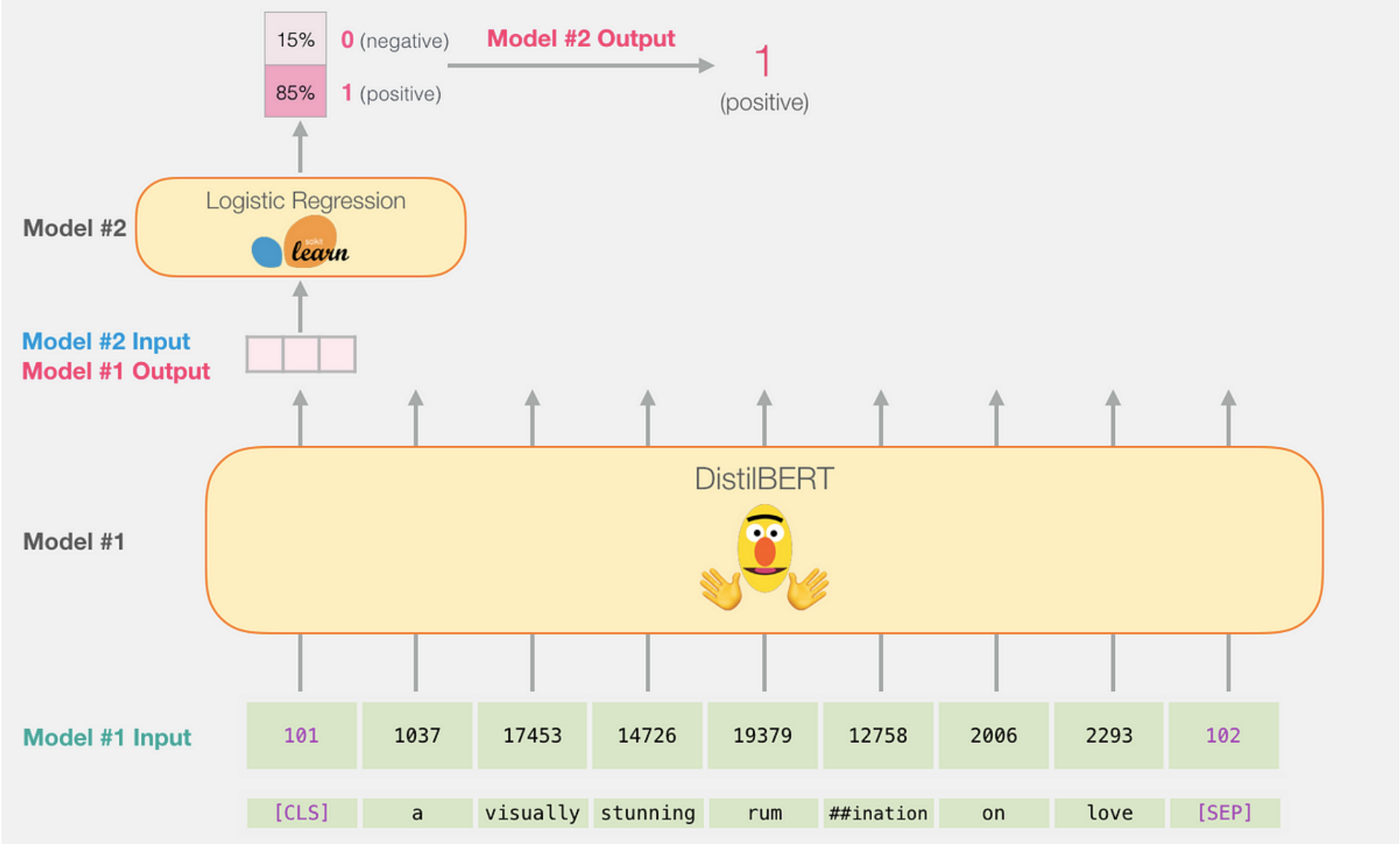
INPUT GENERATION FOR PRETRAINING



2-PHASE BERT TRAINING

- 1.Pretraining to generally understand the language
- 2.Fine-tuning for a specific task:
  - Text Classification
  - Question Answering

Fine-Tuning


15%
0 (negative)
85%
1 (positive)
Model #2 Output
1
(positive)
Model #2
Logistic Regression
Model #2 Input
Model #1 Output
Model #1
DistilBERT
Model #1 Input
101
1037
17453
14726
19379
12758
2006
2293
102
[CLS]
a
visually
stunning
rum
##ination
on
love
[SEP]

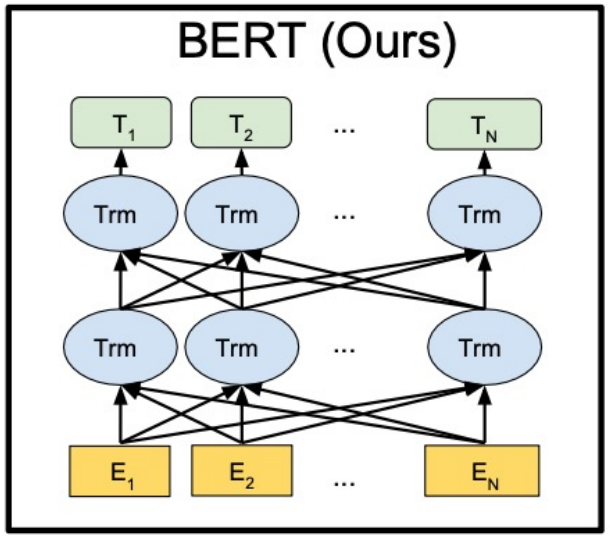
BERT-BASE

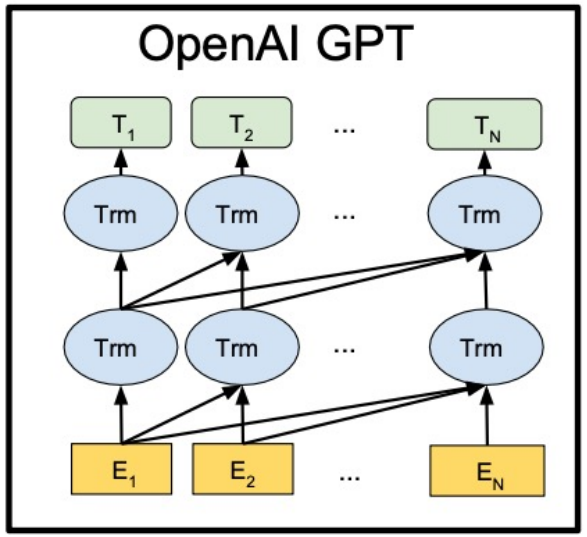
- L=12
- H=768
- A=12
- Total param.=110M

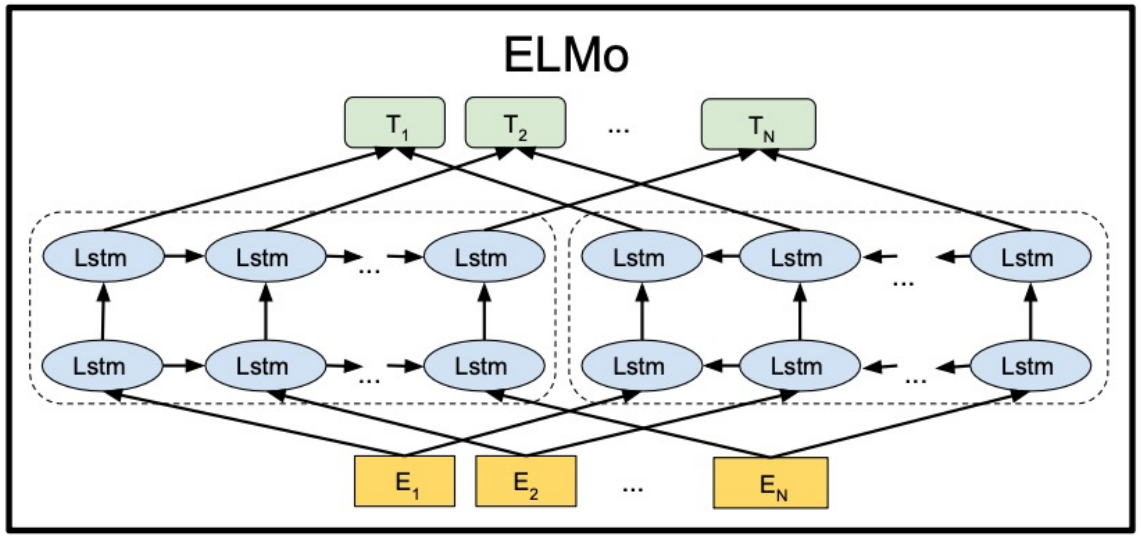
BERT-LARGE

- L=24
- H=1024
- A=16
- Total param.=340M

BERT VS. OTHER MODELS

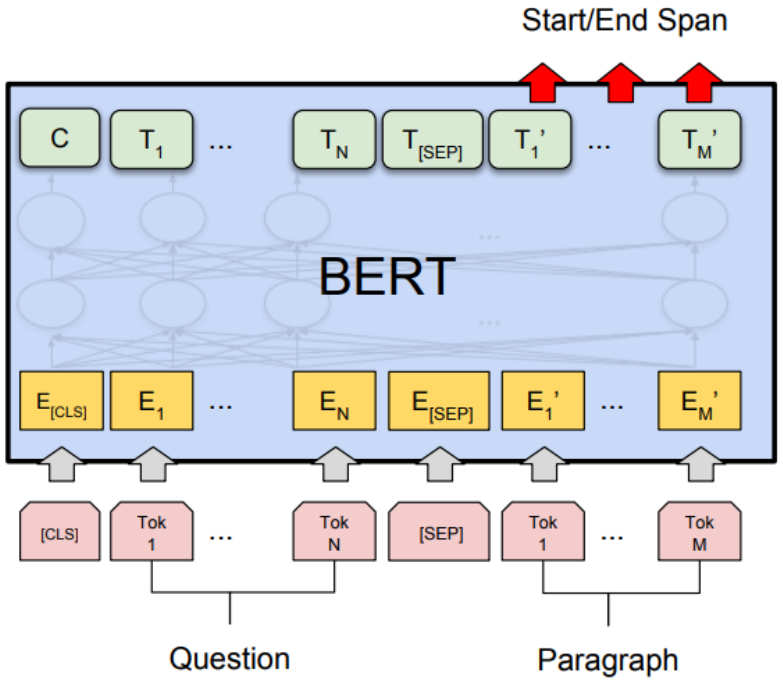






RESULTS

- Fine-Tuning for a specific task
- Mentioned in paper: GLUE, SQuAD 1.1, SQuAD 2.0, SWAG



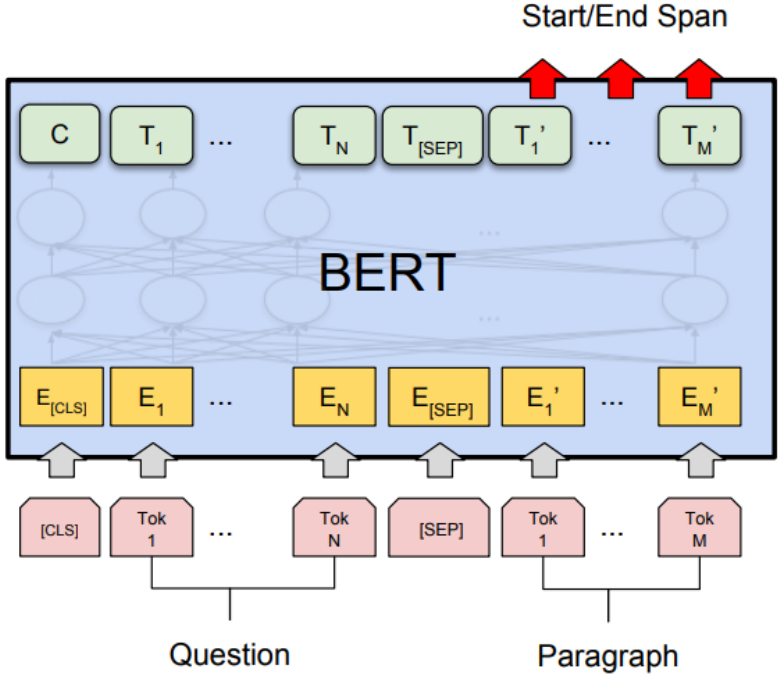
$$P_i = \frac{e^{S \cdot T_i}}{\sum_j e^{S \cdot T_j}}$$

$$S \cdot T_i + E \cdot T_j$$

System	Dev		Test	
	EM	F1	EM	F1
Top Leaderboard Systems (Dec 10th, 2018)				
Human	-	-	82.3	91.2
#1 Ensemble - nlnet	-	-	86.0	91.7
#2 Ensemble - QANet	-	-	84.5	90.5
Published				
BiDAF+ELMo (Single)	-	85.6	-	85.8
R.M. Reader (Ensemble)	81.2	87.9	82.3	88.5
Ours				
$\text{BERT}_{\text{BASE}}$ (Single)	80.8	88.5	-	-
$\text{BERT}_{\text{LARGE}}$ (Single)	84.1	90.9	-	-
$\text{BERT}_{\text{LARGE}}$ (Ensemble)	85.8	91.8	-	-
$\text{BERT}_{\text{LARGE}}$ (Sgl.+TriviaQA)	**84.2**	**91.1**	**85.1**	**91.8**
$\text{BERT}_{\text{LARGE}}$ (Ens.+TriviaQA)	**86.2**	**92.2**	**87.4**	**93.2**

FINE-TUNING

- Fine-Tuning for a specific task
- Mentioned in paper: GLUE, SQuAD 1.1, SQuAD 2.0, SWAG



$$s_{\text{null}} = S \cdot C + E \cdot C$$

$$\hat{s_{i,j}} = \max_{j \geq i} S \cdot T_i + E \cdot T_j$$

$$\hat{s_{i,j}} > s_{\text{null}} + \tau$$

System	Dev		Test	
	EM	F1	EM	F1
Top Leaderboard Systems (Dec 10th, 2018)				
Human	86.3	89.0	86.9	89.5
#1 Single - MIR-MRC (F-Net)	-	-	74.8	78.0
#2 Single - nlnet	-	-	74.2	77.1
Published				
unet (Ensemble)	-	-	71.4	74.9
SLQA+ (Single)	-		71.4	74.4
Ours				
$\text{BERT}_{\text{LARGE}}$ (Single)	78.7	81.9	80.0	83.1

GLUE

- Uses CLS token
- Additional classification layer weights K x H

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**

EFFECT OF PRE-TRAINING TASKS

- LTR and No NSP directly comparable to GPT
- Possible to concatanate LTR and RTL: unintuitive + expensive

	Dev Set				
Tasks	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

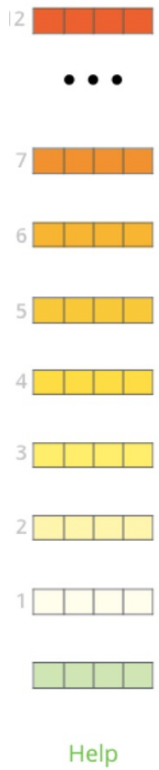
EFFECTS OF MODEL SIZE

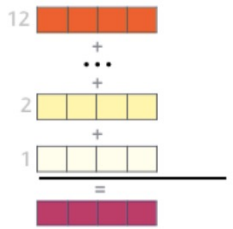
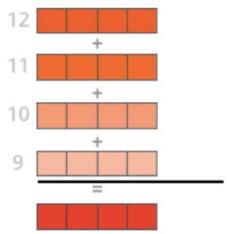
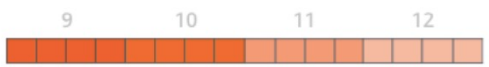
Hyperparams				Dev Set Accuracy		
#L	#H	#A	LM (ppl)	MNLI-m	MRPC	SST-2
3	768	12	5.84	77.9	79.8	88.4
6	768	3	5.24	80.6	82.2	90.7
6	768	12	4.68	81.9	84.8	91.3
12	768	12	3.99	84.4	86.7	92.9
12	1024	16	3.54	85.7	86.9	93.3
24	1024	16	3.23	86.6	87.8	93.7

FEATURE-BASED APPROACH WITH BERT

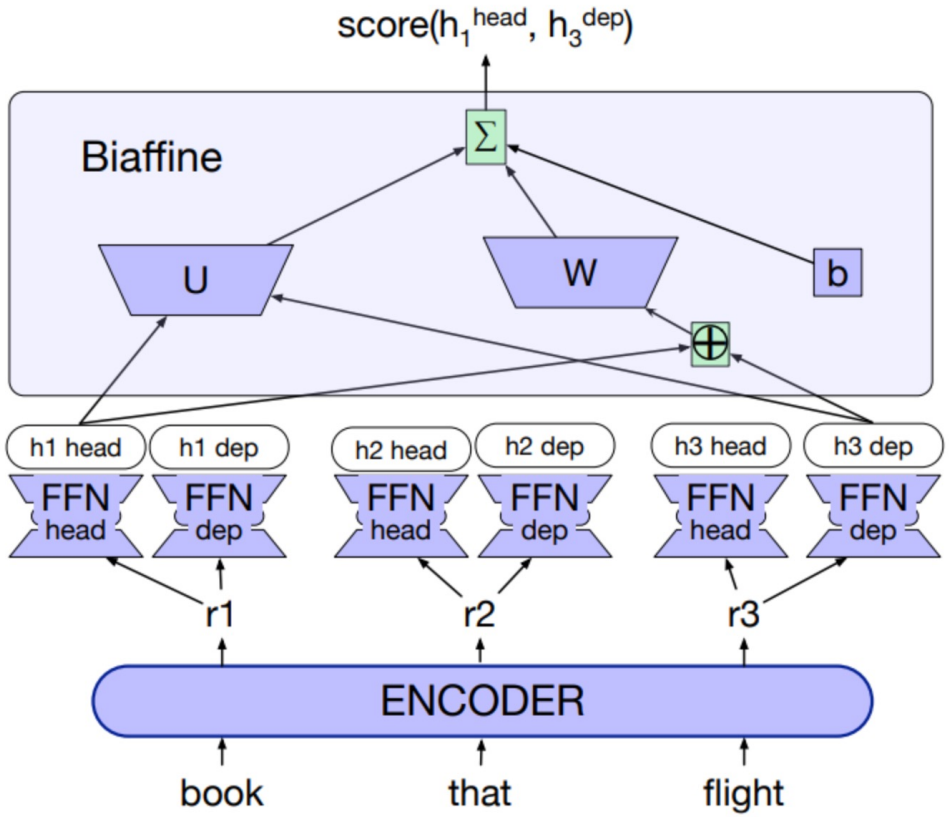
System	Dev F1	Test F1
ELMo (Peters et al., 2018a)	95.7	92.2
CVT (Clark et al., 2018)	-	92.6
CSE (Akbik et al., 2018)	-	**93.1**
Fine-tuning approach		
$\text{BERT}_{\text{LARGE}}$	96.6	92.8
$\text{BERT}_{\text{BASE}}$	96.4	92.4
Feature-based approach ($\text{BERT}_{\text{BASE}}$)		
Embeddings	91.0	-
Second-to-Last Hidden	95.6	-
Last Hidden	94.9	-
Weighted Sum Last Four Hidden	95.9	-
Concat Last Four Hidden	96.1	-
Weighted Sum All 12 Layers	95.5	-

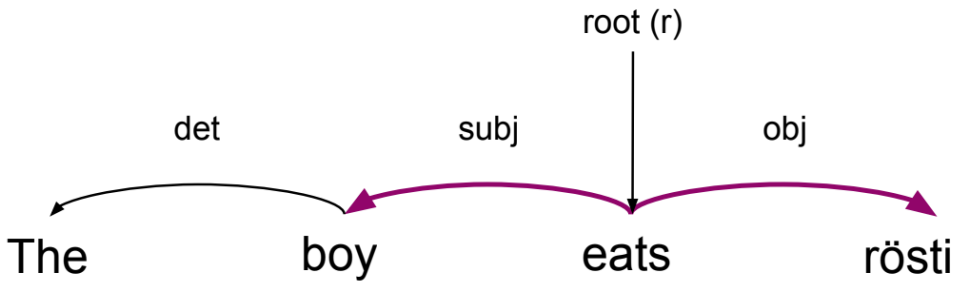
What is the best contextualized embedding for "Help" in that context?
For named-entity recognition task CoNLL-2003 NER



		Dev F1 Score
First Layer		91.0
Last Hidden Layer		94.9
Sum All 12 Layers		95.5
Second-to-Last Hidden Layer		95.6
Sum Last Four Hidden		95.9
Concat Last Four Hidden		96.1

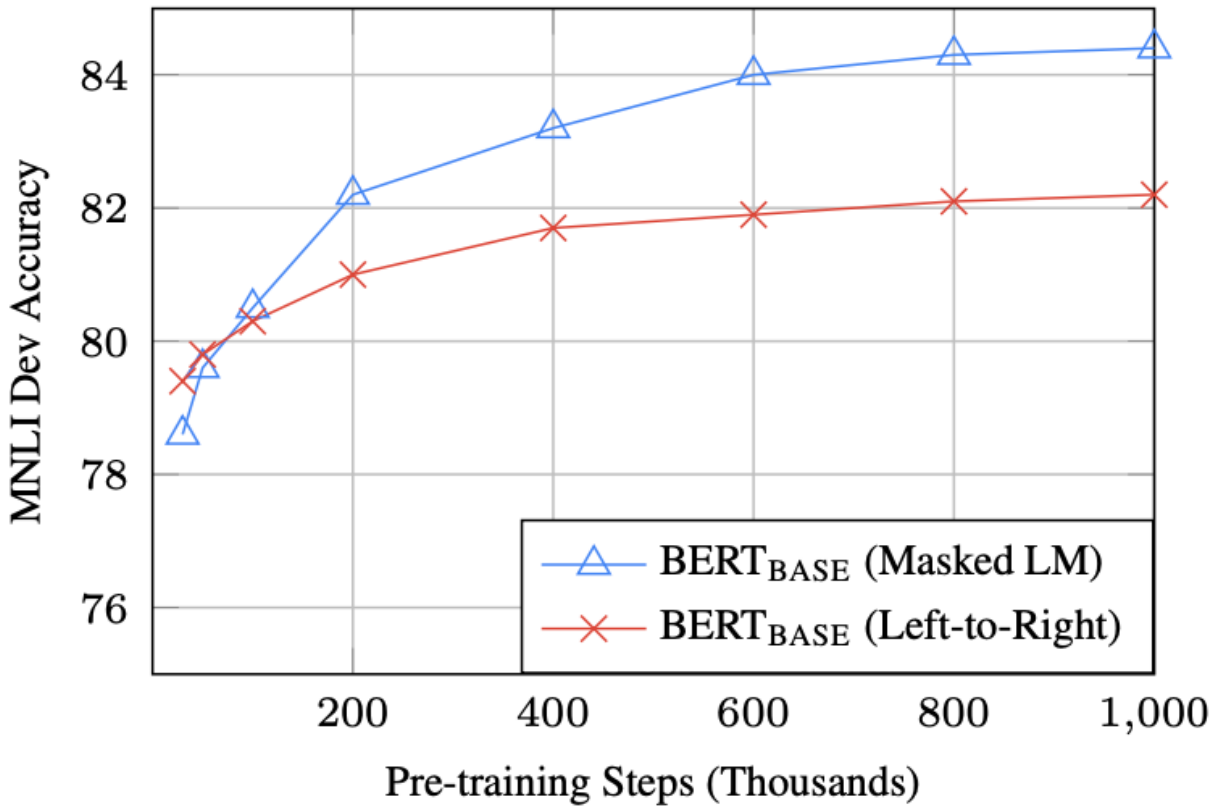
BIAFFINE SCORER





EFFECT OF NUMBER OF TRAINING STEPS

- LTR is better with small number of training steps
- Masked LM is better in the long run



CONTRIBUTIONS

- Bidirectional Language Model
- Generality and advantages of a fine-tuning approach
- Many other models created: RoBERTa, DistilBERT, ALBERT, BERT-multilingual, FinBERT

MY OPINION ON THE PAPER

- Lot of implementation details
- Background not explained, rather referenced to other papers
- Many applications
- Missing data in the results tables

RELATED WORK

- "Universal Language Model Fine-tuning for Text Classification" by Jeremy Howard and Sebastian Ruder (2018)
- "BERT for Coreference Resolution: Baselines and Analysis" by Mandar Joshi et al. (2019)
- "BERT for Joint Intent Classification and Slot Filling" by Bing Liu et al. (2019)

Questions?

F1 SCORE

$$\text{F1 Score} = \frac{2}{\frac{1}{\text{Precision}} + \frac{1}{\text{Recall}}} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

DIFFERENT MASKING STRATEGIES

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

Words As Vectors