Google BERT

Pre-training of Deep Bidirectional Transformers for Language Understanding

20210419 발표자 이주영

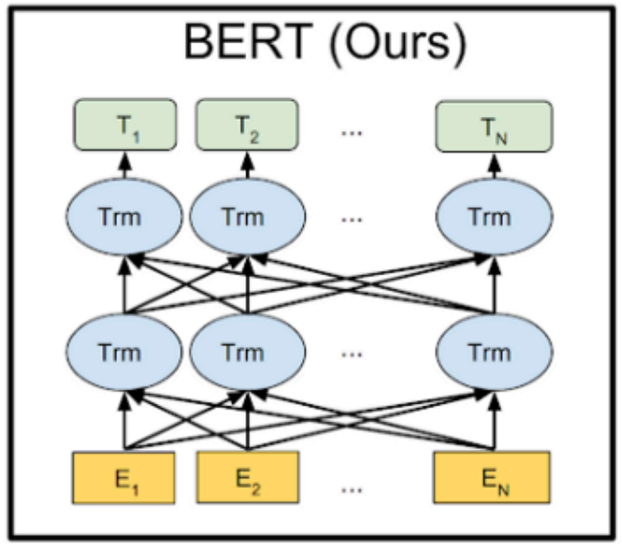
BERT?

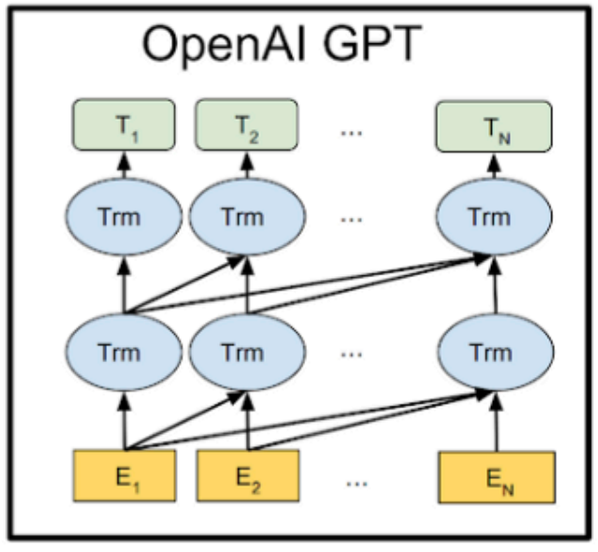
- 2018년 10월에 논문이 공개된 구글의 새로운 Language Representation Model
- 특정 분야에 국한된 기술이 아니라 모든 자연어 응용 분야에서 좋은 성능을 내는 범용 모델인 Language Model
- NLP의 11개 task에서 최고 성능을 기록

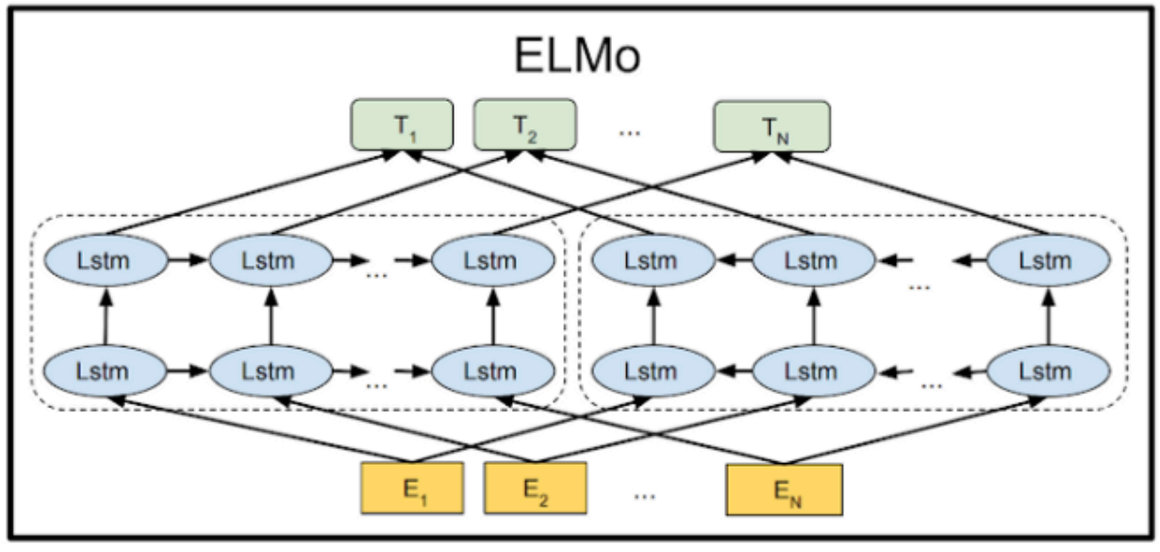
BERT is a method of pre-training language representations,
meaning that we train a general-purpose “language understanding”
model on a large text corpus (BooksCorpus and Wikipedia),
and then use that model for downstream NLP tasks (fine tuning)
that we care about (like question answering — SQuAD).

BERT?

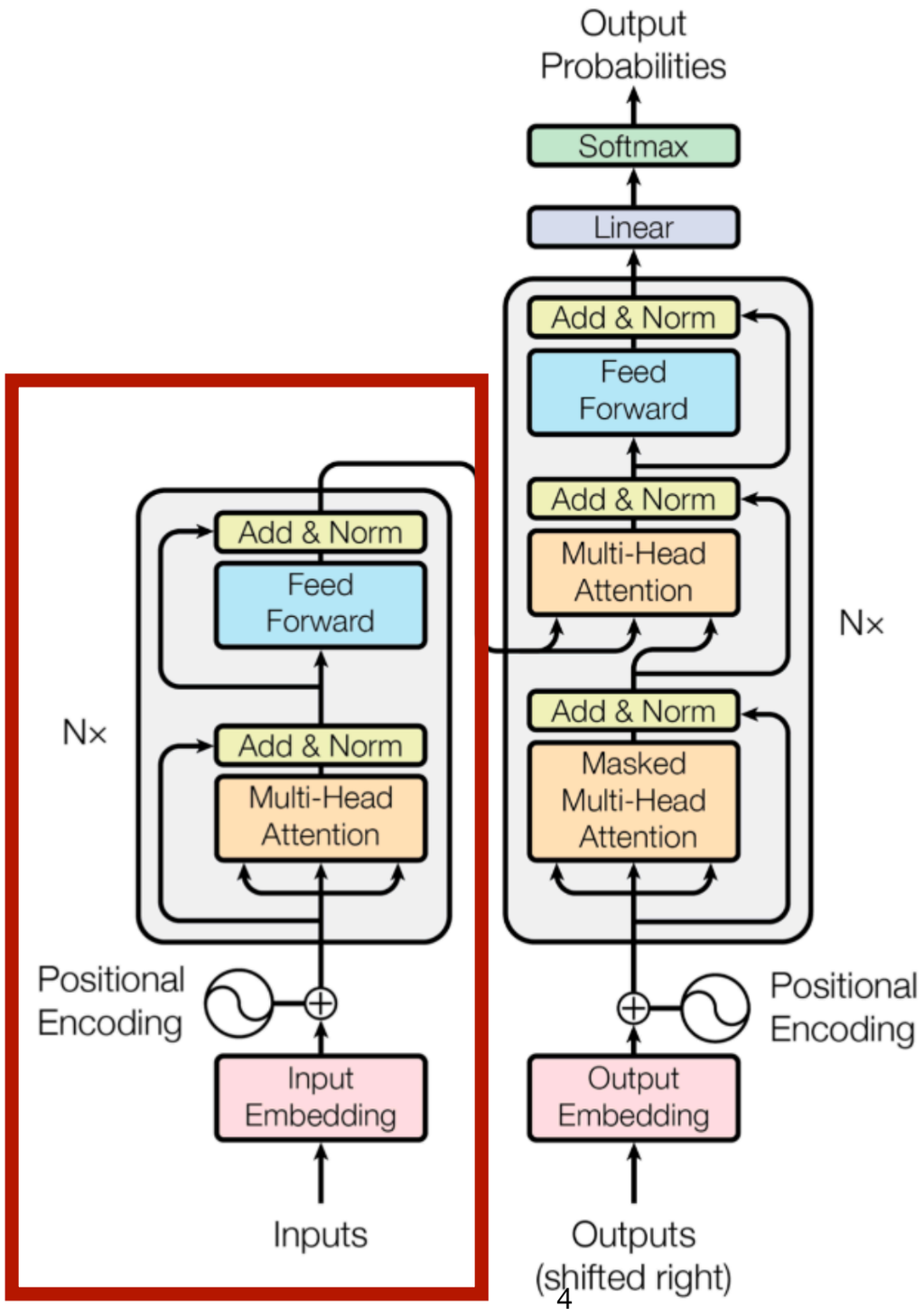
- **B**idirectional **E**ncoder **R**epresentations from **T**ransformer
- **Pre-trained, Fine-tuning**
- Masked Language Model, Next Sentence Prediction





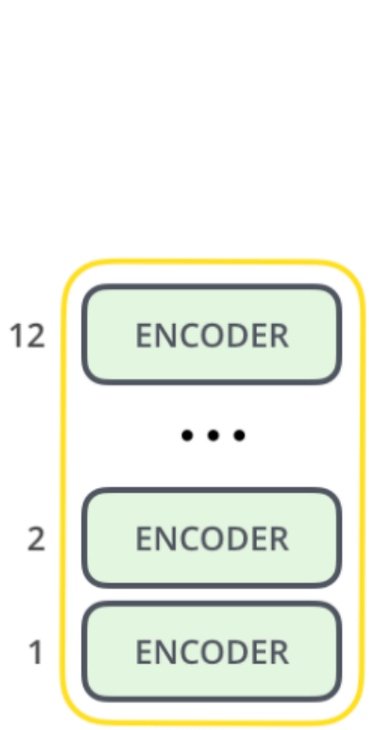


Model Architechture

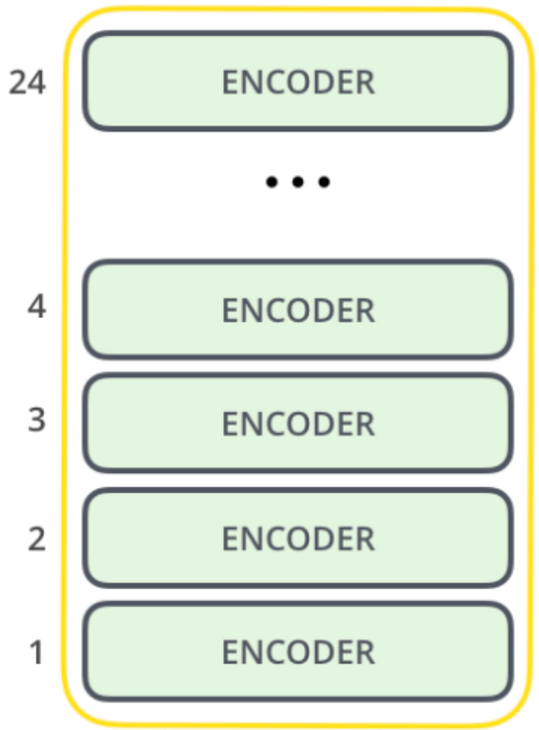


Model Architechture

Model	Layers (Transformer Blocks)	Hidden Size	Self-Attention Heads	Feed Forward/Filter Size	Total Parameters
BERT-Base	12	768	12	3072	110M
BERT-Large	24	1024	16	4096	340M



$BERT_{BASE}$



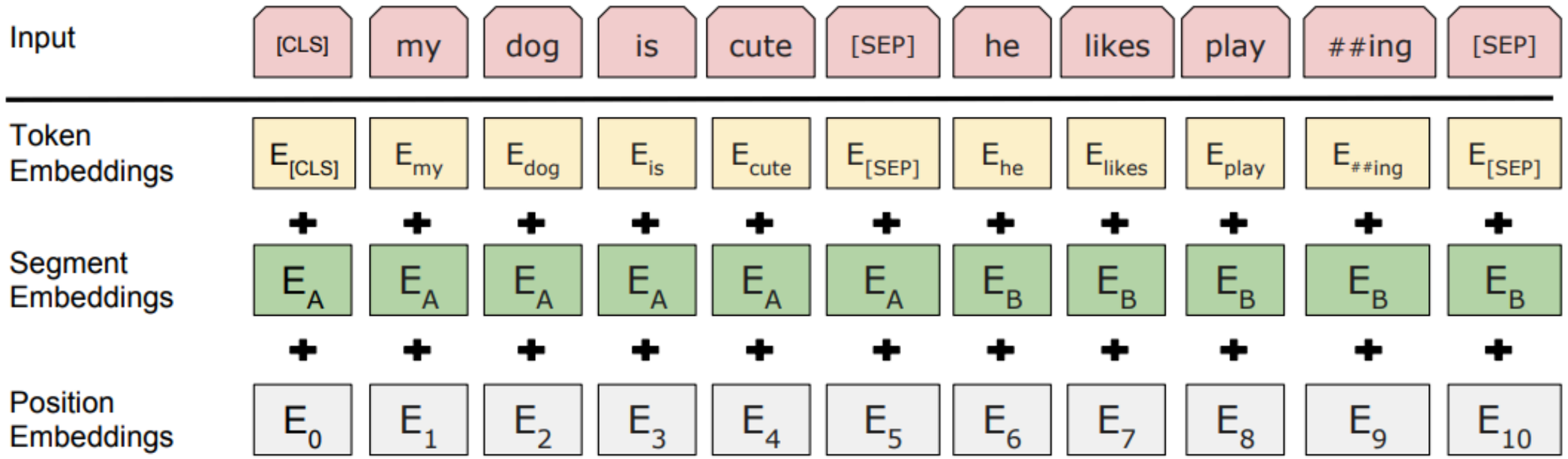
$BERT_{LARGE}$

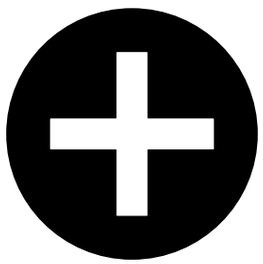
Training Data

- 총 33억 단어(8억 단어의 BookCorpus 데이터와 25억 단어의 Wikipedia 데이터)의 거대한 말뭉치를 이용하여 학습

- Wikipedia와 BookCorpus를 정제하기 위해 list, table, header를 제거. 그리고 문장의 순서를 고려해야 하므로 문단 단위로 분리하였고 많은 데이터 정제 작업을 수행

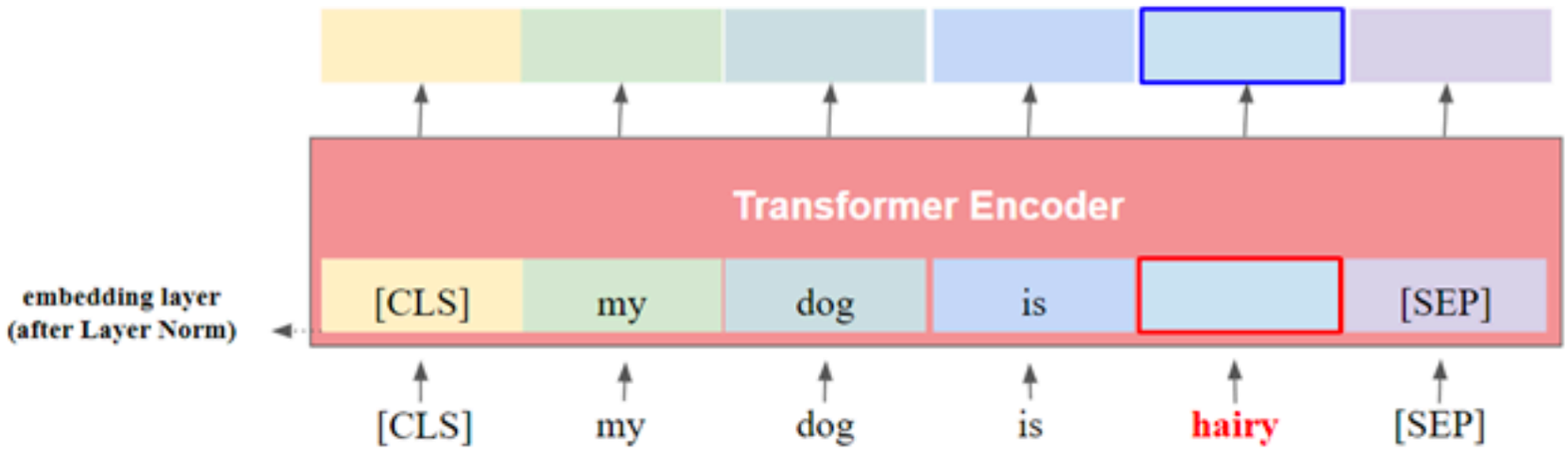
input Representation





Regularization, Drop out

MLM (Masked Language Model)



Mask **15%** of all WordPiece tokens in each sequence at **random**. (e.g., **hairy**)

[MASK] 80% of the time : Replace **[MASK]** token.

apple 10% of the time : Replace the word with a **random** word

hairy 10% of the time : Keep the word **unchanged**.

NSP(Next Sentence Prediction)

Input = [CLS] the man went to [MASK] store [SEP] → Sentence A

he bought a gallon [MASK] milk [SEP] → Sentence B

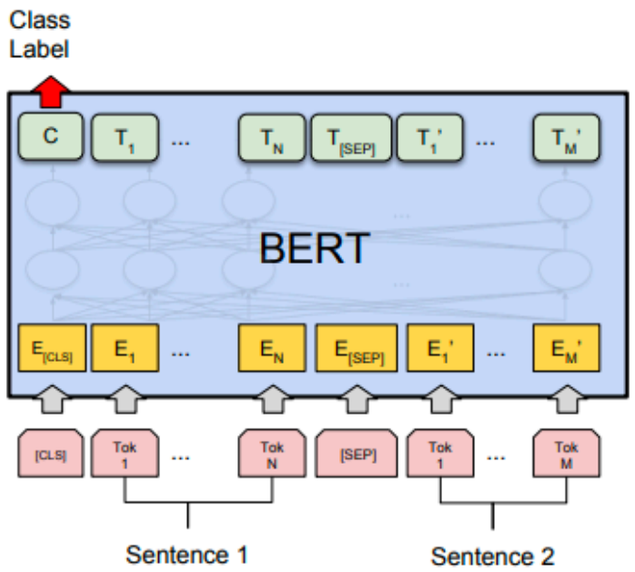
Label = IsNext

Input = [CLS] the man [MASK] to the store [SEP]

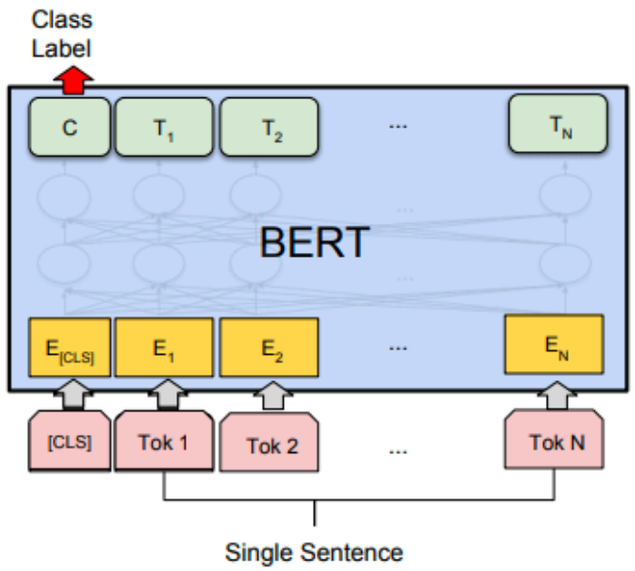
penguin [MASK] are flight ##less birds [SEP]

Label = NotNext

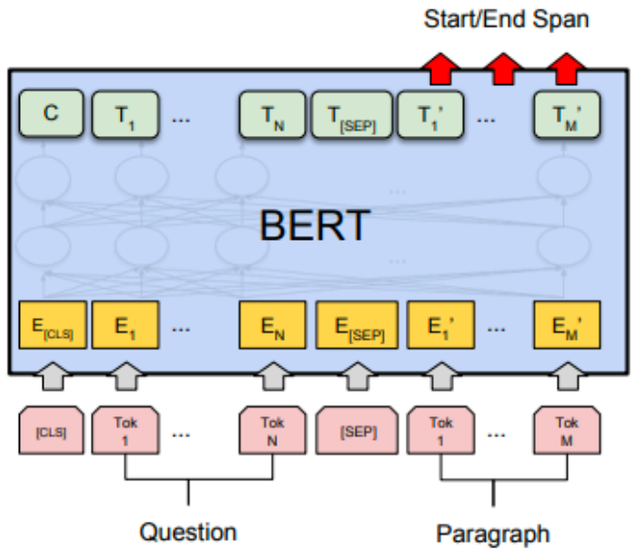
Transfer Learning



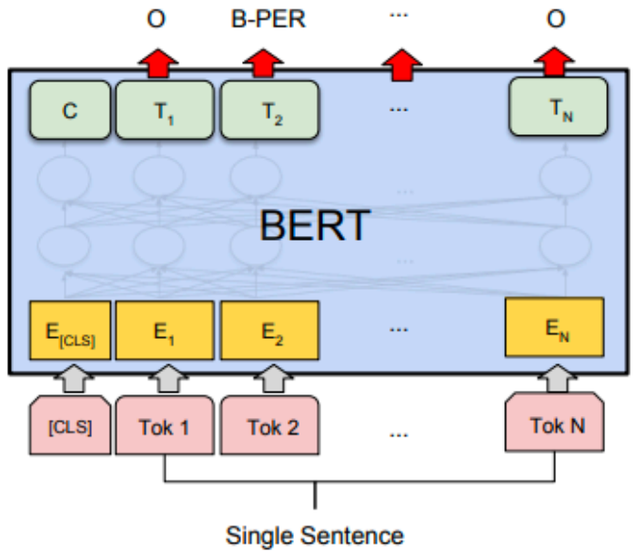
(a) Sentence Pair Classification Tasks:
MNLI, QQP, QNLI, STS-B, MRPC,
RTE, SWAG



(b) Single Sentence Classification Tasks:
SST-2, CoLA



(c) Question Answering Tasks:
SQuAD v1.1



(d) Single Sentence Tagging Tasks:
CoNLL-2003 NER

Result

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**

GLUE

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

Absolution Study

Result

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

Model Size 별 Accuracy

Conclusion

- 일반 NLP task 에는 강하지만, specific 한 domain 에 들어가면 잘 작동하지 않음
- Google 에서 공개한 pre-trained model 을 쓰면 편리하고 시간을 절약할 수 있으며 효율적임
- 개인이 BERT로 pre-trained model을 쓰려면 시간 및 resource가 너무 많이 들기 때문에, Google의 것을 training 시키는 것이 효율적임