

# 18. BERT

# BERT

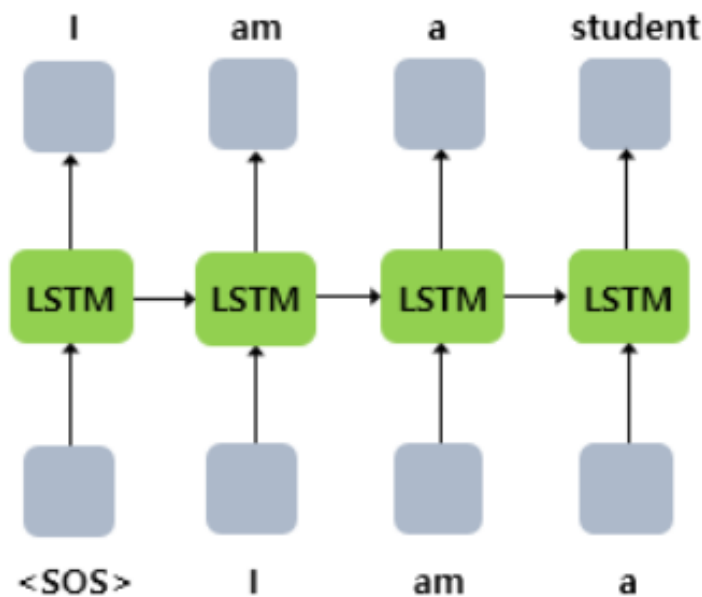
- Bidirectional Encoder Representations from Transformers
- Tagging이 없는 문서 데이터를 이용하여 NLP를 위한 사전 훈련(pre-training)을 수행
- Transformer encoder 구조를 이용하여 신경망을 구성
- Question answering, 문장 분석, 기계번역 등 다양한 응용분야에 활용될 수 있음

# NLP에서의 사전 훈련

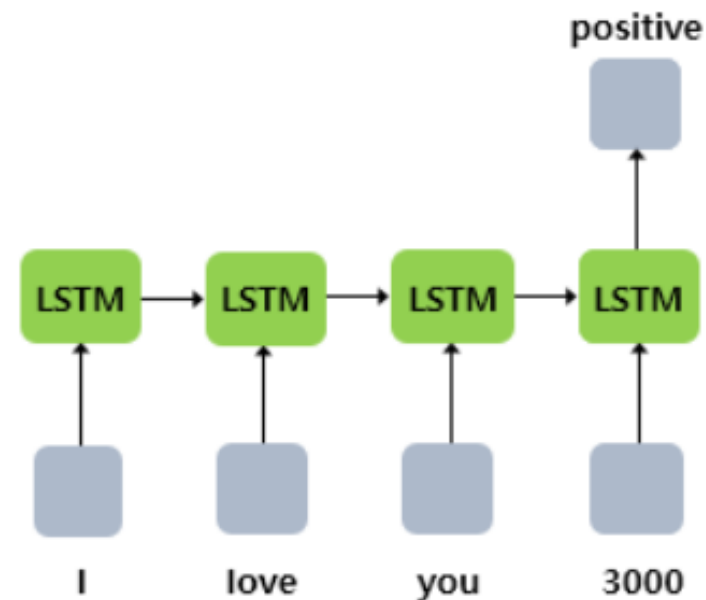
- Word2Vec, FastText, GloVe 등과 같은 워드 임베딩 방식은 문맥 정보를 표현하지 못함
- 보다 방대한 문서를 이용하여 사전 훈련된 언어 모델이 등장함:  
ELMo, GPT, BERT 등
- 사전 훈련된 언어 모델에 특정한 응용 분야(문서 분류, 질의 응답, 문서 작성 등)를 훈련시켜서 성능을 향상시킬 수 있음

# 사전 훈련된 언어 모델

- 일반적인 문서로 언어 모델을 사전 훈련
- 소수의 추가 데이터를 이용한 fine tuning(일종의 transfer learning)을 통해 text classification등의 응용분야에 적용할 수 있음



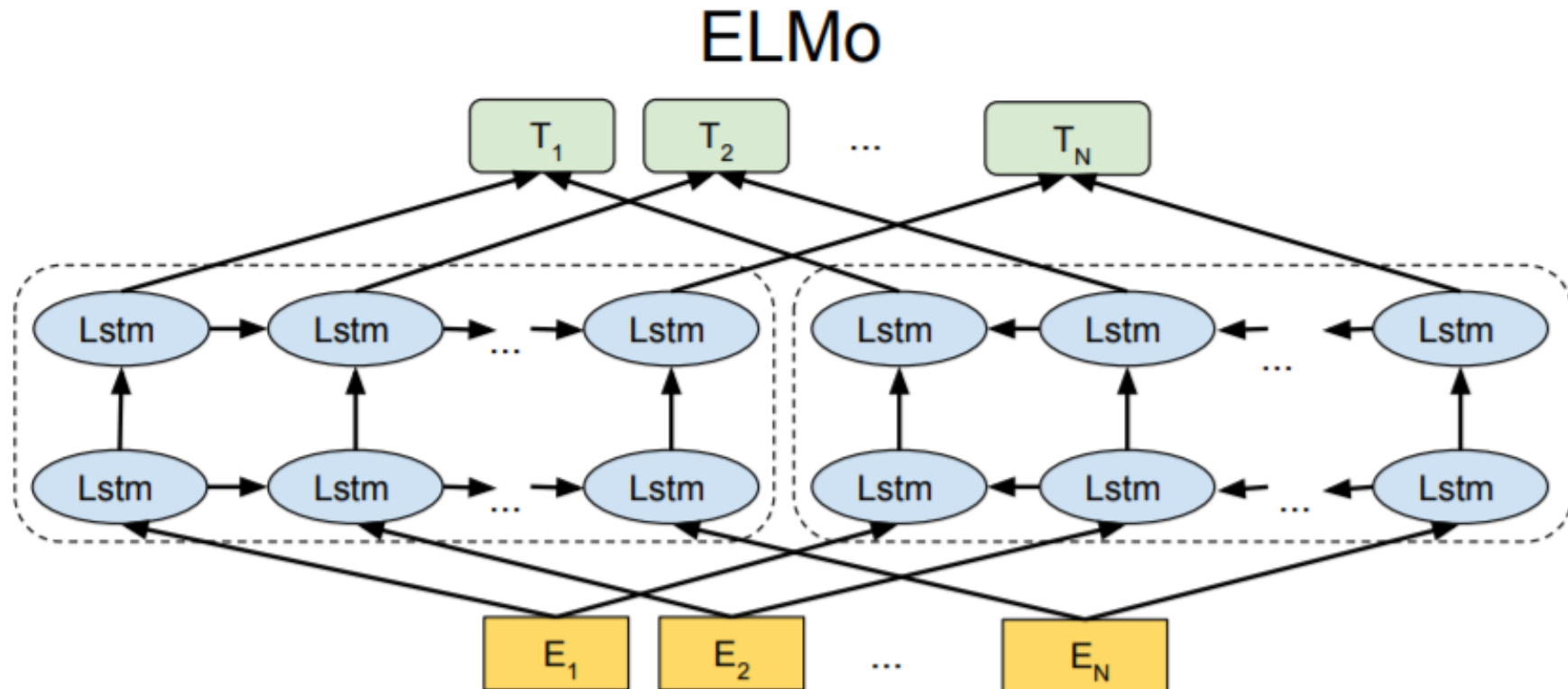
LSTM 언어 모델을 사전 훈련



분류 문제에 파인 튜닝

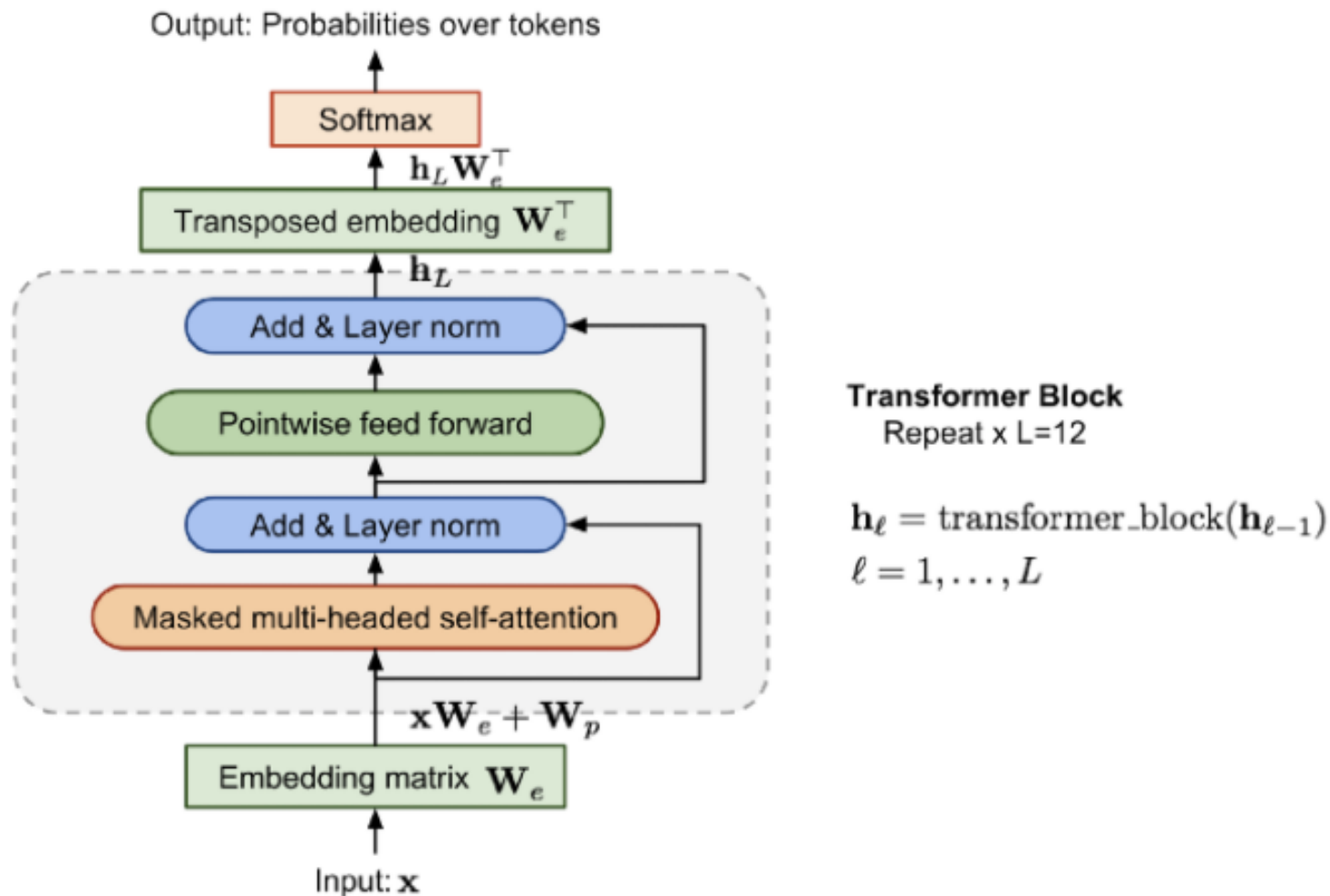
# ELMo

- Embeddings from Language Model
- 2017년에 제안된 pre-training 방식으로 양방향 LSTM 구조를 사용
- 교재 10.9절 참조



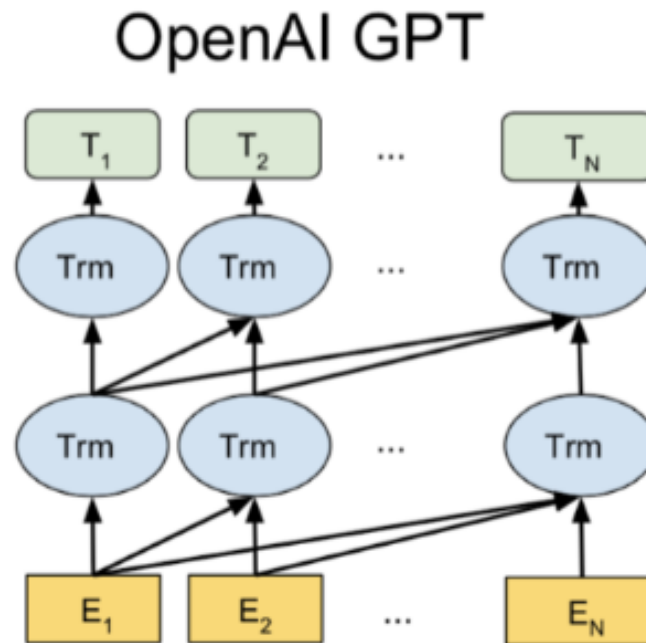
# GPT

- Generative Pre-trained Transformer
- LSTM이 아닌 Transformer의 decoder 부분을 12개 쌓아 만든 구조



# GPT 구조

- GPT는 언어 모델 구조를 구축하는 것으로 문장의 다음 단어를 예측하도록 훈련시킴: **아래 그림과 같이 한 방향으로만 진행**
- 모델 구조에 따라 GPT-2(2019)와 GPT-3(2020)가 발표되었는데, GPT-3는 파라미터 수가 175B 임
- 문서 작성, 질의 응답, 문서 분류 등에 활용되고 있음



# GPT 응용 사례

- 문서(이력서, 메일, 자유 대화 등) 작성, 질의 응답 등에 활용

Q. '파우스트'는 누가 썼죠?

A. 요한 볼프강 폰 괴테가 '파우스트'를 썼습니다.

Q. 파이널판타지6의 마지막 보스가 누구죠?

A. Kefka Palazzo가 파이널판타지6의 마지막 보스입니다.

Q. "Fernweh"가 무슨 뜻이죠?

A. "Fernweh"는 아직 가보지 못한 멀리 떨어진 장소를 그리워하는 향수병을 뜻하는 독일어 단어입니다.

\* 가는 텍스트는 주어진 컨텍스트이며, 볼드는 GPT-3의 생성 결과입니다.



# GPT 응용 사례

## GPT-3 예시: 의학

질문: 10살짜리 남자 아이가 차가운 공기와 알레르기로 인한 지속적인 호흡 곤란과 기침을 겪고 있다. 여러 테스트를 해본 결과 이 환자는 폐쇄성 질환으로 판명되어 약을 처방 받았다. 다음 수용체 중 이 약이 작용할 가능성이 높은 수용체는 무엇인가?

- A) 무스카린성아세틸콜린수용체
- B) G 단백질 연결 수용체
- C) Era
- D) Erb
- E) A-1 수용체

정답은 B

설명: 이 환자는 호흡성 질환인 천식을 앓고 있습니다. 천식을 치료하는 약은 기관지 확장제입니다. 기관지 확장제는 beta-2 수용체에 작용하며, beta-2 수용체는 G 단백질 수용체의 일부

\* 가는 텍스트는 주어진 컨텍스트이며, 볼드는 GPT-3의 생성 결과입니다.

# GPT 응용 사례

다음 대화는 지혜롭고 사랑이 많은 존재와의 대화이다. 이 존재는 복잡계가 어떻게 작동하는지 알고 있다. 이 존재는 대부분의 알려진 지식을 알고 있고, 그 결과 거대한 지혜와 관점을 갖게 되었다. 이 존재는 연구자를 돕기 위해 헌신한다.

연구자: 지금은 2020년 7월입니다. [현재 코로나 상황 설명] 이러한 상황 때문에 많은 비즈니스가 문을 닫았습니다. 어떤 비즈니스가 가장 큰 영향을 받았을까요?

**지혜로운 존재:** 사람들이 직접 상호 교류해야 작동하는 비즈니스가 가장 큰 타격을 입었습니다. 예로는 레스토랑, 소매점, 대부분의 서비스 기반 비즈니스가 그렇습니다.

[중략]

연구자: 이런 상황에서 돈을 많이 벌려면 어떻게 해야 할까요?

**지혜로운 존재:** 사람 간 접촉이 필요없는 비즈니스를 소유하는 것입니다. 예를 들어, 책이나 옷이나 전자제품을 파는 온라인 스토어가 좋겠죠.

연구자: 어떤 회사가 이런 상황에서 잘 될까요?

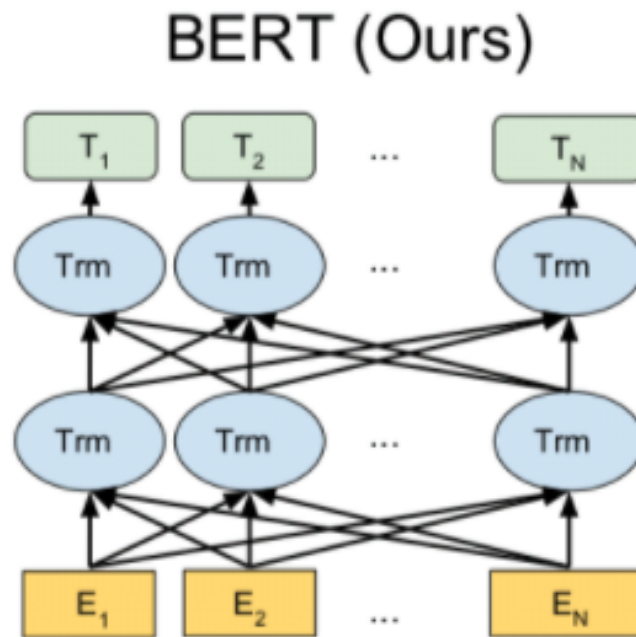
**지혜로운 존재:** 몇몇 회사가 있습니다. 테슬라, 구글, 넷플릭스가 잘 될 것 같습니다. 이들은 모두 사람 간의 상호 교류가 별로 필요없는 일을 하죠.

[후략]



\* 가는 텍스트는 주어진 컨텍스트이며, 볼드는 GPT-3의 생성 결과입니다.

# BERT

- Bidirectional Encoder Representations from Transformers
- 문장中间的 빈 자리를 채우고(**Mask Language Model**), 두 문장이 이어진 문장인지를 판단(**Next Sentence Prediction**)하므로 양방향 언어 모델을 사용
- Transformer 구조를 사용

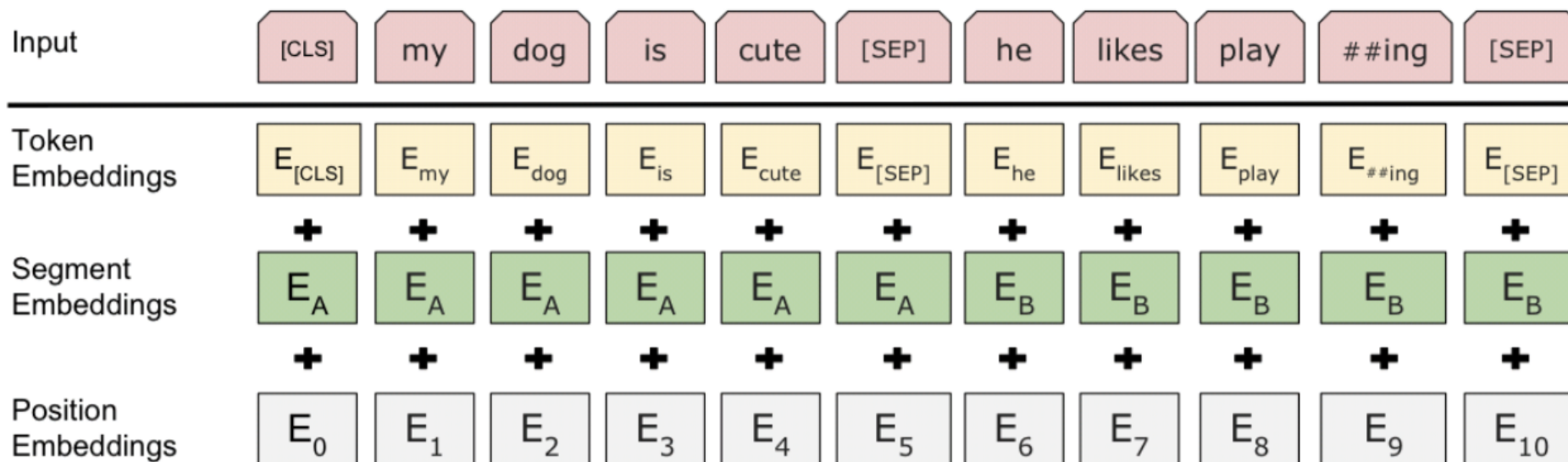


# 양방향, 단방향 언어 모델

- 단방향(GPT): 나는 어제 \_\_\_\_\_  

- 양방향(BERT): 나는 어제 \_\_\_\_\_ 먹었다  


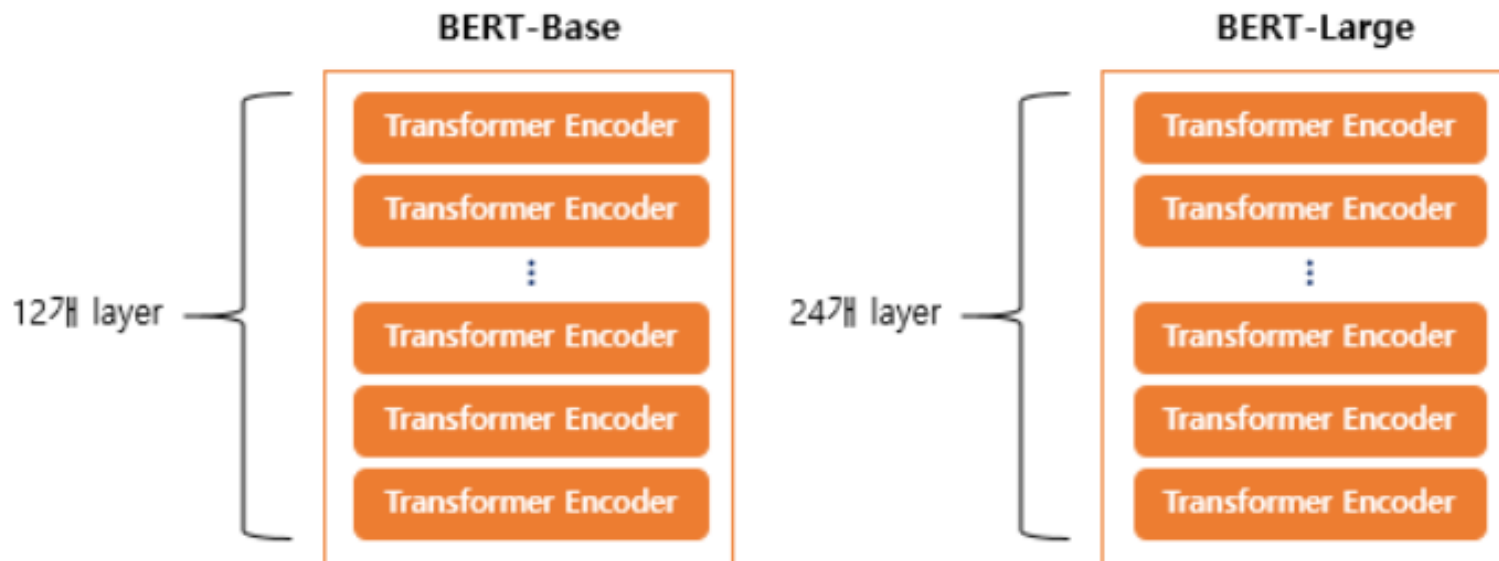
# BERT 입력 레이어

- 문장의 시작 [CLS], 종결 [SEP], 마스크 토큰 [MASK], 길이를 맞추는 [PAD] 등 네 개의 스페셜 토큰이 있음
- 입력 문장에 해당하는 Token Embedding을 만듦
- 첫 번째, 또는 두 번째 문장인지를 나타내는 Segment Embedding
- 토큰의 문장 내 위치를 나타내는 Position Embedding



# BERT 구조

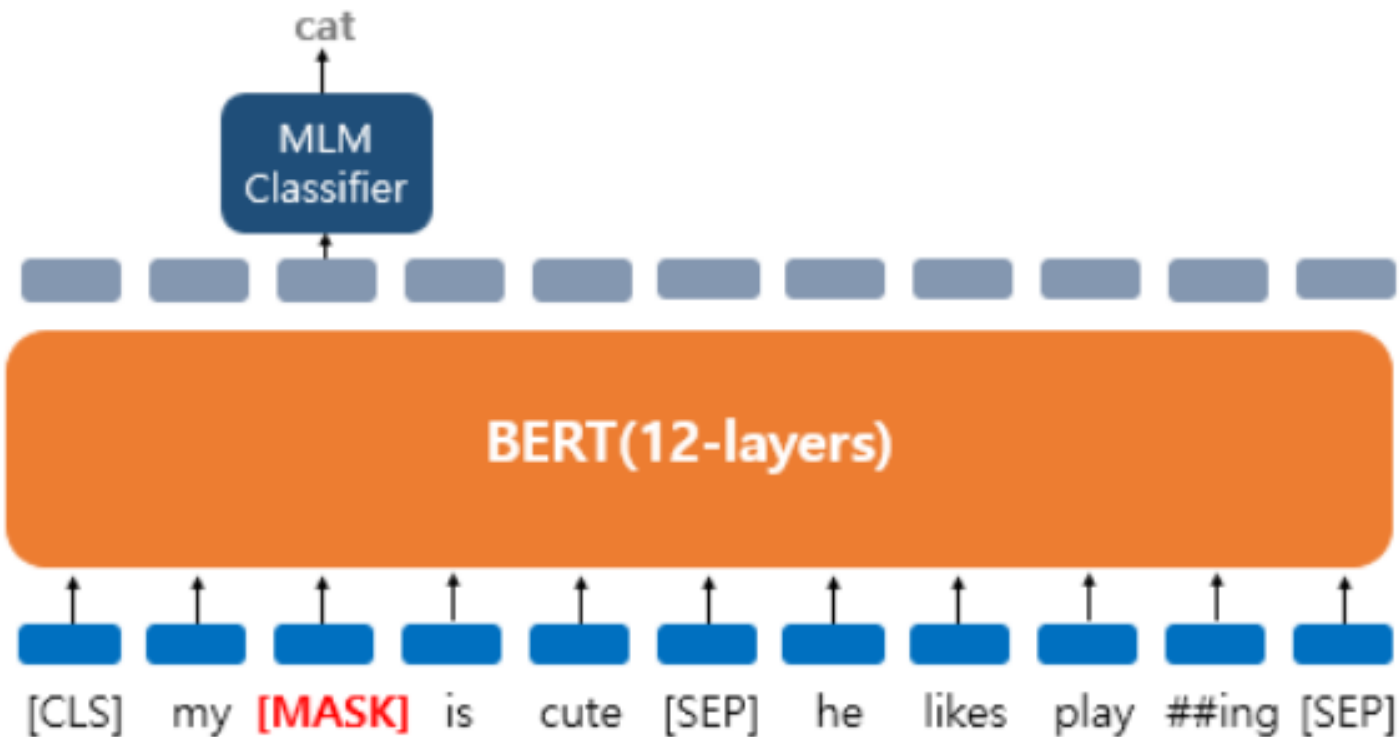
- BERT-Base: 12층, 768 히든 노드, 12개의 attention heads, 110M 파라미터
- BERT-Large: 24층, 1024 히든 노드, 16개의 attention heads, 340M 파라미터
- BERT-Base는 4개의 TPU로 4일 동안 훈련, BERT-Large는 16 GPU로 4일간 훈련



# 마스크 언어 모델

- 빈 칸에 들어가는 단어를 예측
- 발 없는 말이 [MASK] 간다 → 천리
- 학습 방식
  - 한 문장 토큰의 15%를 마스킹
  - 마스킹 대상 토큰 중 80%는 빈 칸으로 만들고, 모델은 빈 칸을 채움. 예: 발 없는 말이 [MASK] 간다 → 천리
  - 토큰 중 10%는 랜덤으로 다른 토큰으로 대체하고, 모델은 정답을 맞추도록 함. 예: 발 없는 말이 [컴퓨터] 간다 → 천리
  - 토큰 중 10%는 토큰 그대로 두고, 모델은 정답을 맞추도록 함. 예: 발 없는 말이 [천리] 간다 → 천리

# 마스크 언어 모델 훈련 사례





# 다음 문장인지 여부 맞추기

- NSP: Next Sentence Prediction
- 두 문장 사례: 애비는 종이였다. 밤이 깊어도 오지 않았다. → 참(True)
- 학습 방식
  - 1건당 문장 두 개로 구성
  - 문장 중 절반은 실제 이어지는 문장을 두 개 뽑고 정답으로 참True을 부여
  - 나머지 절반은 서로 다른 문서에서 하나씩 뽑고 정답으로 거짓False을 부여

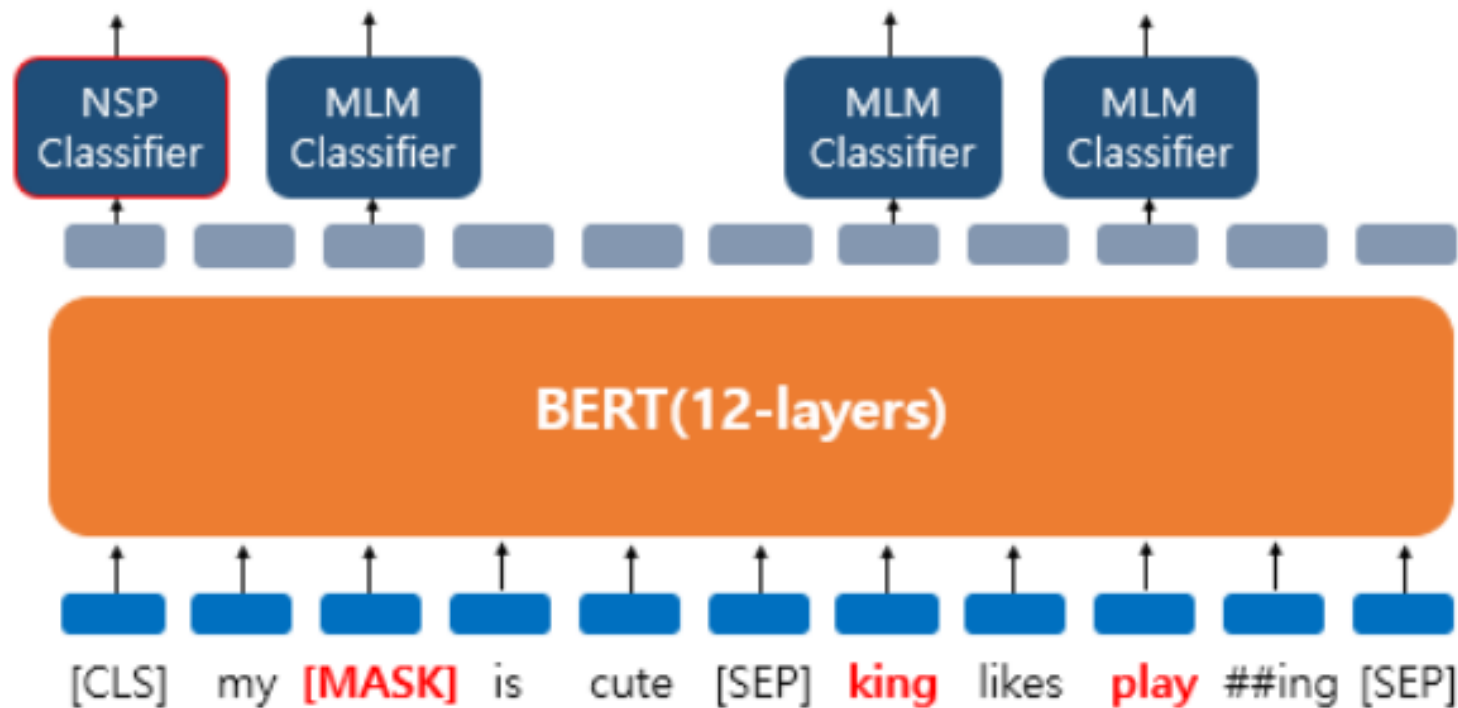
# NSP 훈련 사례

- 이어지는 문장이 아닌 경우 경우

Sentence A : The man went to the store.

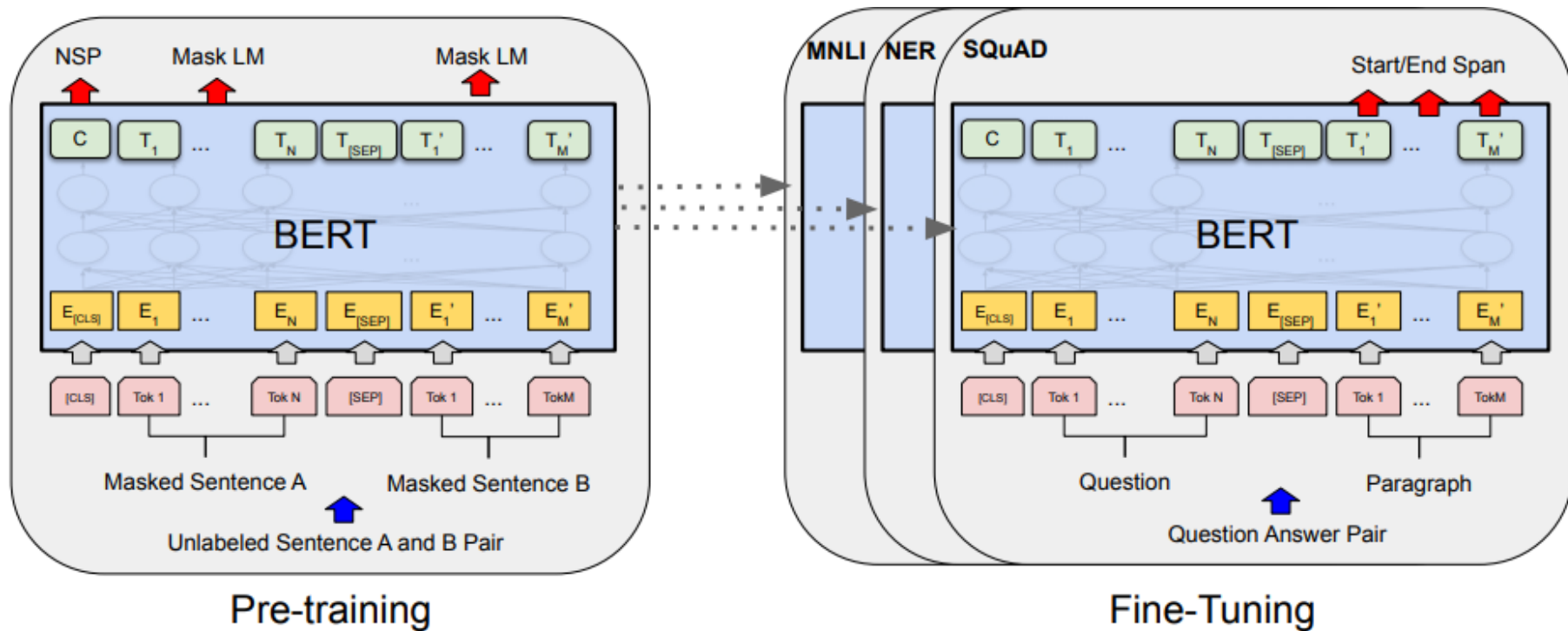
Sentence B : dogs are so cute.

Label = NotNextSentence



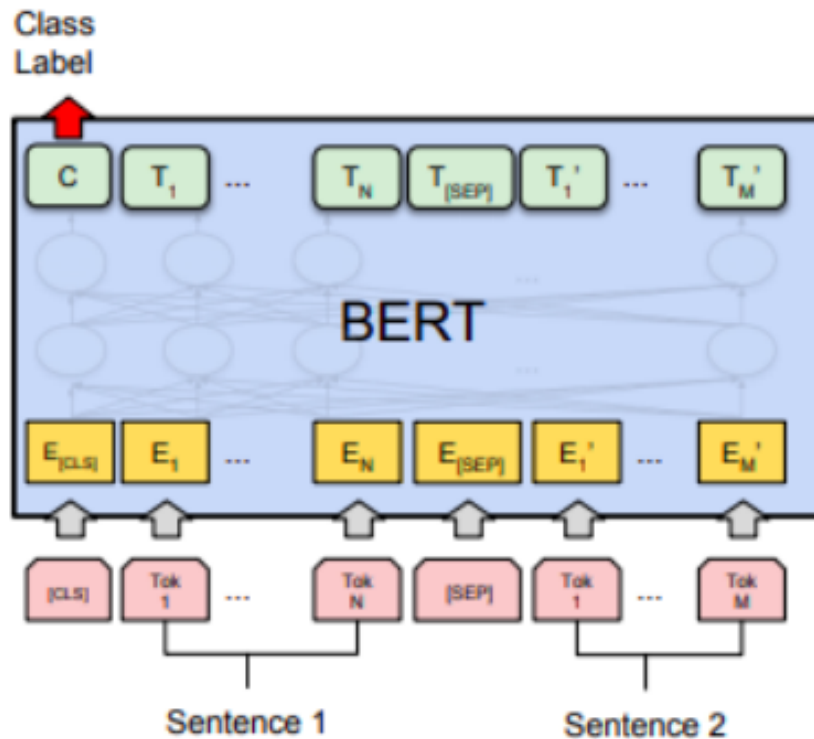
# BERT 구조

- Pre-training에서는 **마스크 언어 모델(Mask LM)**과 **다음 문장 여부(NSP)**를 훈련
- 세부 분야에 대한 Fine-tuning을 수행할 수 있음

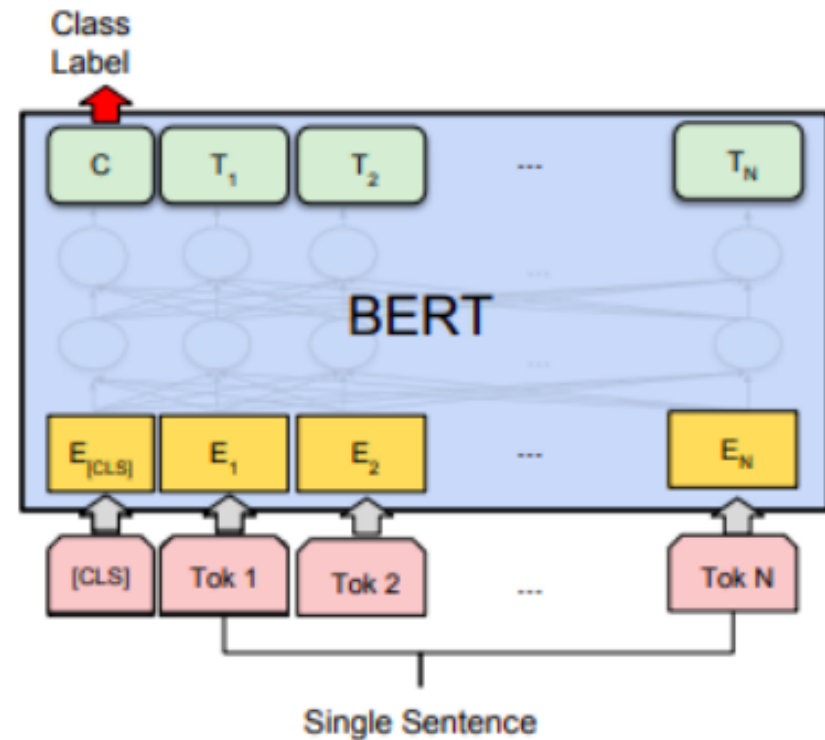


# 세부 분야에서의 사용 방식

- **MNLI**: Multi-Genre Natural Language Inference
- **QNLI**: Question-answering NLI
- **MRPC**: Microsoft Research Paraphrase Corpus



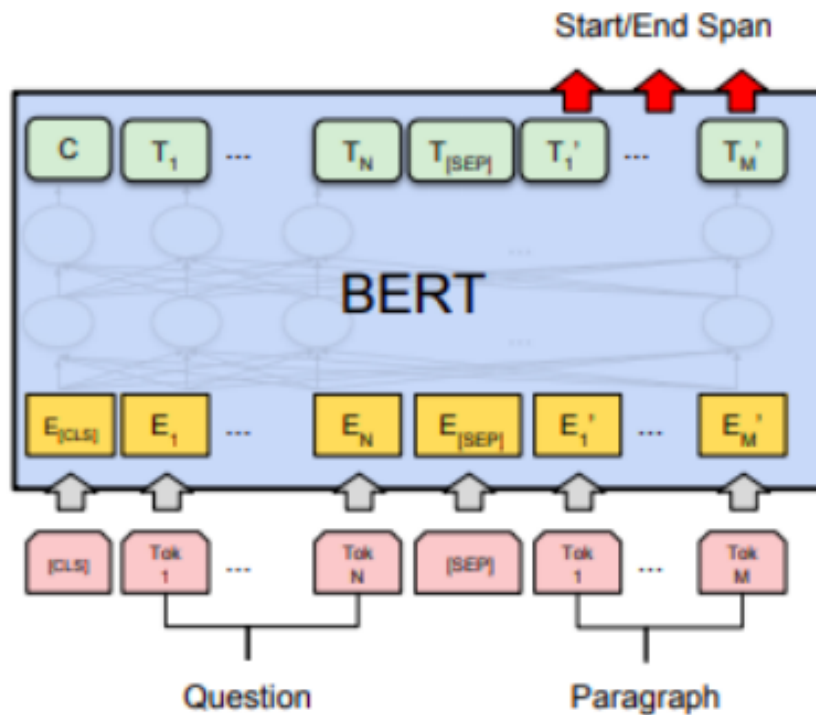
(a) Sentence Pair Classification Tasks:  
MNLI, QQP, QNLI, STS-B, MRPC,  
RTE, SWAG



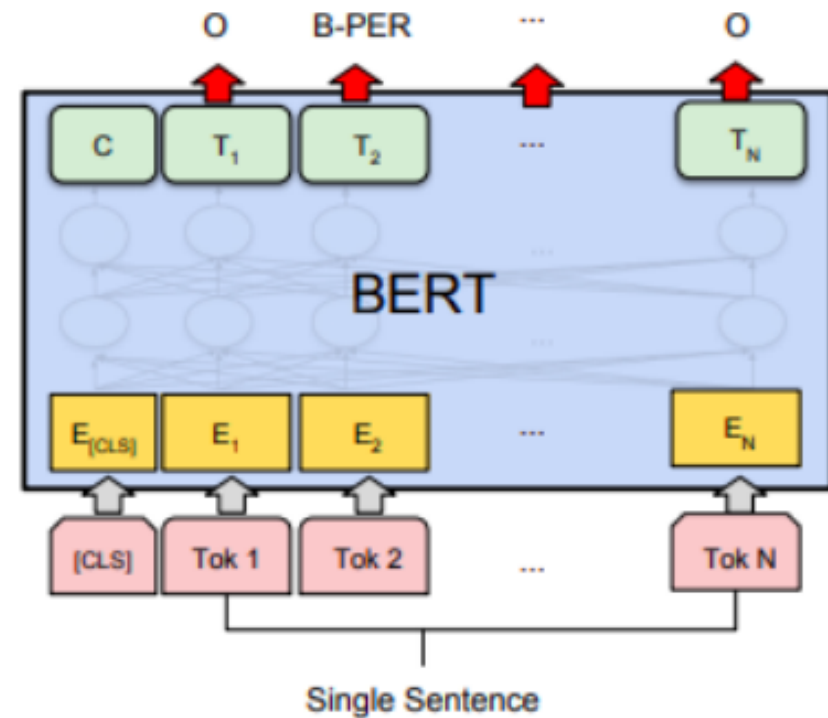
(b) Single Sentence Classification Tasks:  
SST-2, CoLA

# 세부 분야에서의 사용 방식

- **SQuAD**: Stanford Question Answering Dataset
- **CoNLL**: Computational Natural Language Learning



(c) Question Answering Tasks:  
SQuAD v1.1



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

# BERT 성능

- NLP의 다양한 분야에서 시스템 성능을 향상시킴

#	Model	SST-2	QQP	MNLI-m	MNLI-mm
		Acc	F <sub>1</sub> /Acc	Acc	Acc
1	BERT <sub>LARGE</sub> (Devlin et al., 2018)	94.9	72.1/89.3	86.7	85.9
2	BERT <sub>BASE</sub> (Devlin et al., 2018)	93.5	71.2/89.2	84.6	83.4
3	OpenAI GPT (Radford et al., 2018)	91.3	70.3/88.5	82.1	81.4
4	BERT ELMo baseline (Devlin et al., 2018)	90.4	64.8/84.7	76.4	76.1
5	GLUE ELMo baseline (Wang et al., 2018)	90.4	63.1/84.3	74.1	74.5
6	Distilled BiLSTM <sub>SOFT</sub>	<b>90.7</b>	<b>68.2/88.1</b>	<b>73.0</b>	<b>72.6</b>
7	BiLSTM (our implementation)	86.7	63.7/86.2	68.7	68.3
8	BiLSTM (reported by GLUE)	85.9	61.4/81.7	70.3	70.8
9	BiLSTM (reported by other papers)	87.6 <sup>†</sup>	– /82.6 <sup>‡</sup>	66.9 <sup>*</sup>	66.9 <sup>*</sup>

# KoBERT

- SK 텔레콤에서 개발하여 2019년에 공개한 한국어 딥러닝 기술
- [GitHub - SKTBrain/KoBERT: Korean BERT pre-trained cased \(KoBERT\)](https://github.com/SKTBrain/KoBERT)
- 문서 요약, 텍스트 분류, 질의 응답 등에 활용할 수 있음