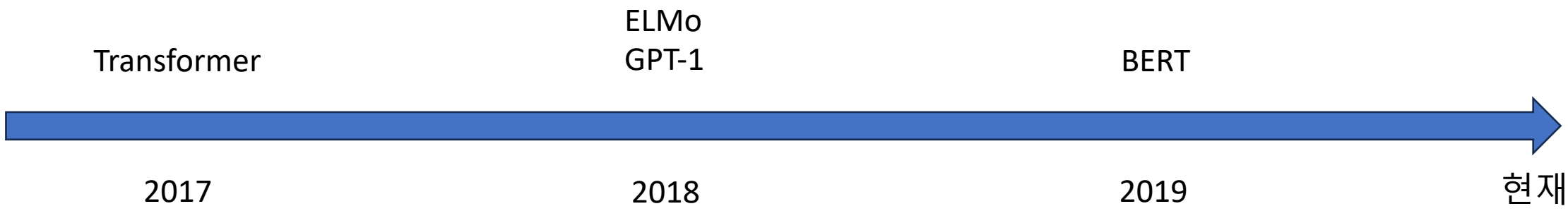
BERT

Samsung Software Developer Community
Korea Vision & Robotics
GiBeom Kim
2023.10.08.

Contents

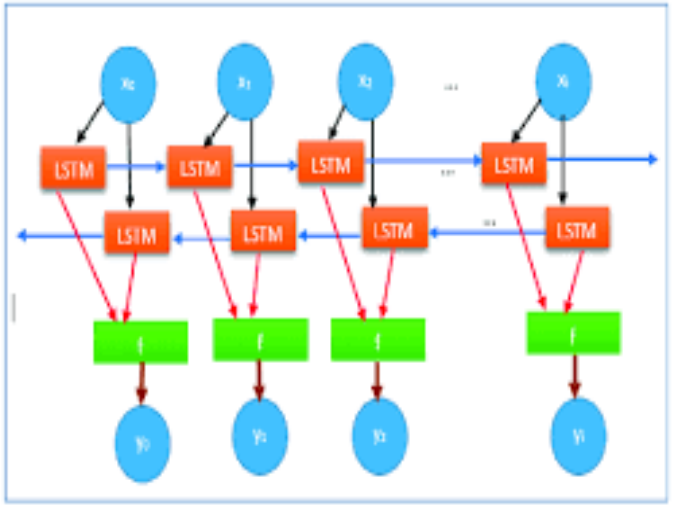
설명

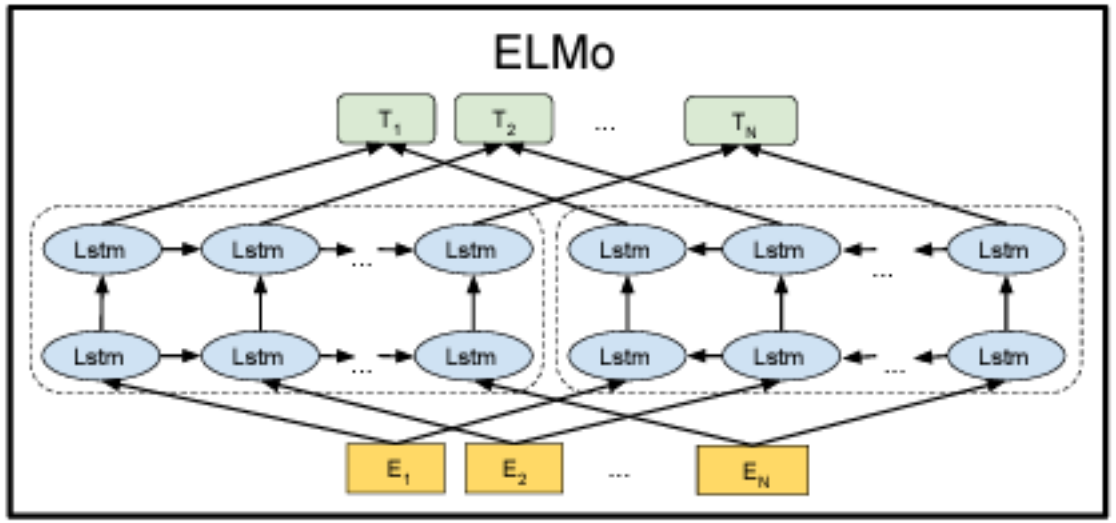
NLP 모델 역사

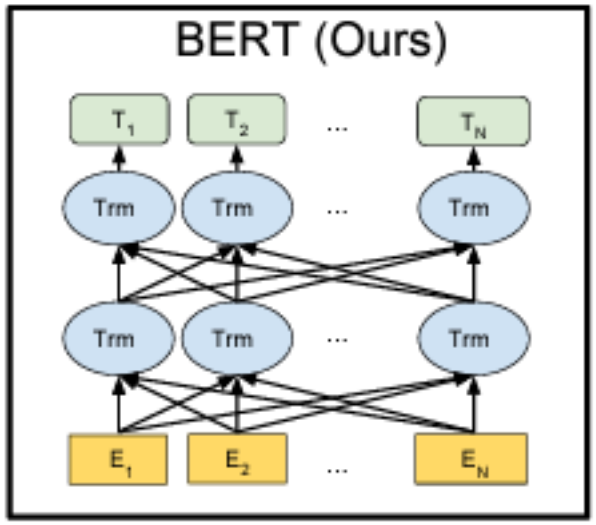


설명

BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding



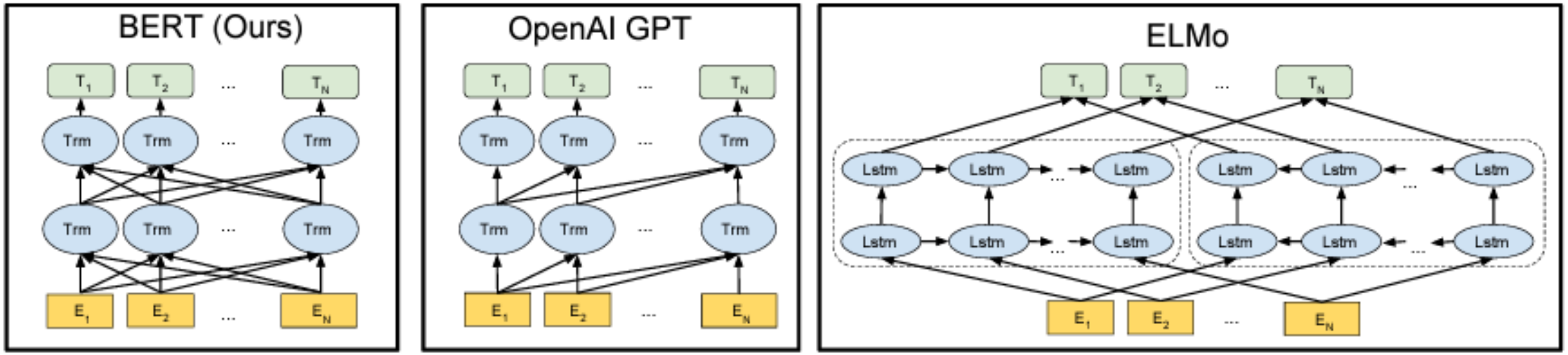




두 개의 단방향 모델을 사용

진짜 양방향

설명



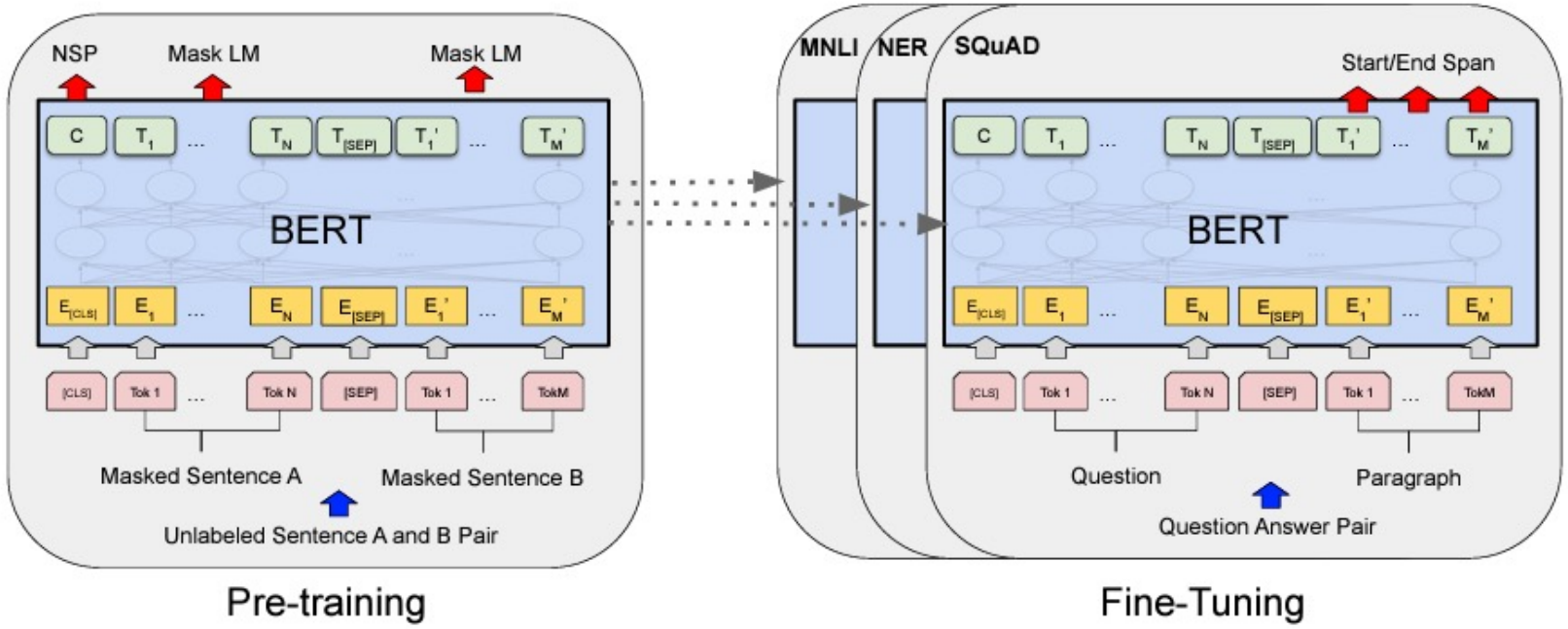
문장의 모든 토큰 사용

현재 토큰 기준 이전 토큰만 사용

두 개의 단방향 LSTM 모델 사용

설명

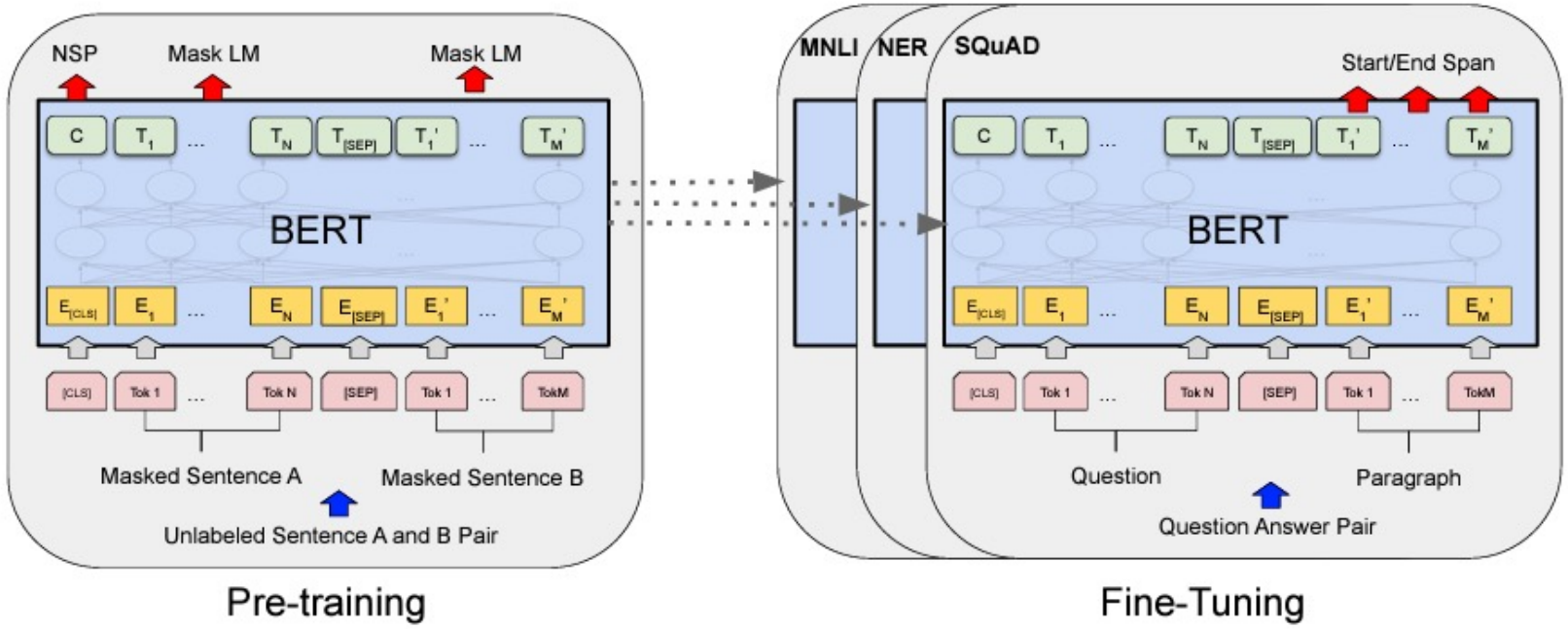
Transfer Learning 용이



- Output Layer만 추가하면 간단하게 Fine-Tuning 가능 (=확장성 좋음)
- 11개의 주요 NLP 태스크 SOTA 달성
- Pre-training 단계에서만 Masked Language Model(이하 MLM) 사용

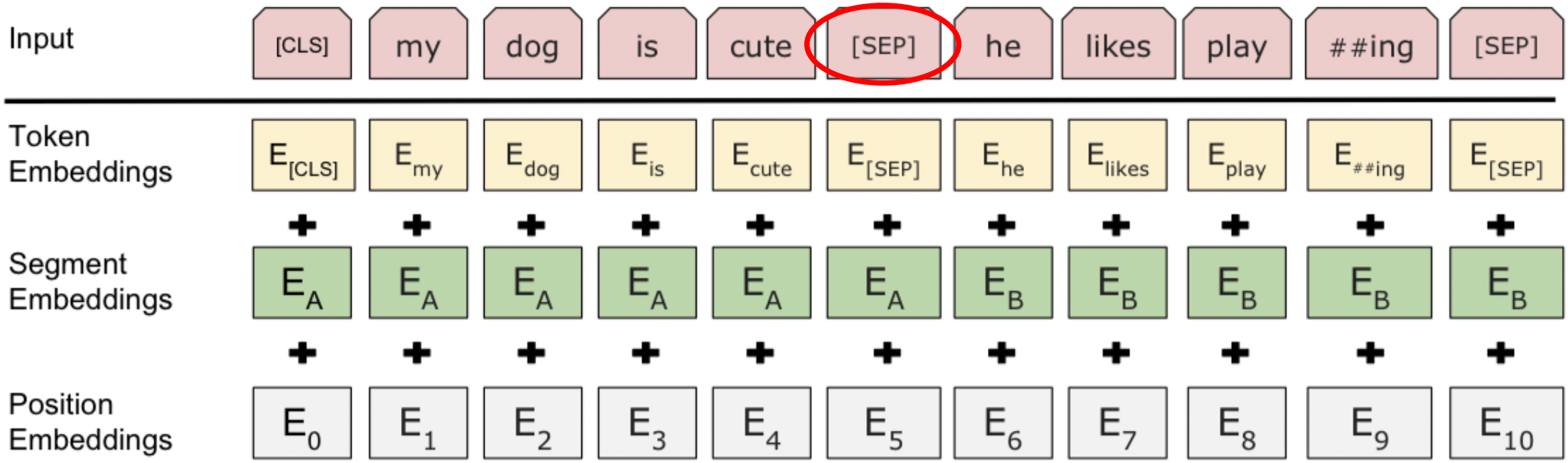
설명

Transfer Learning 용이



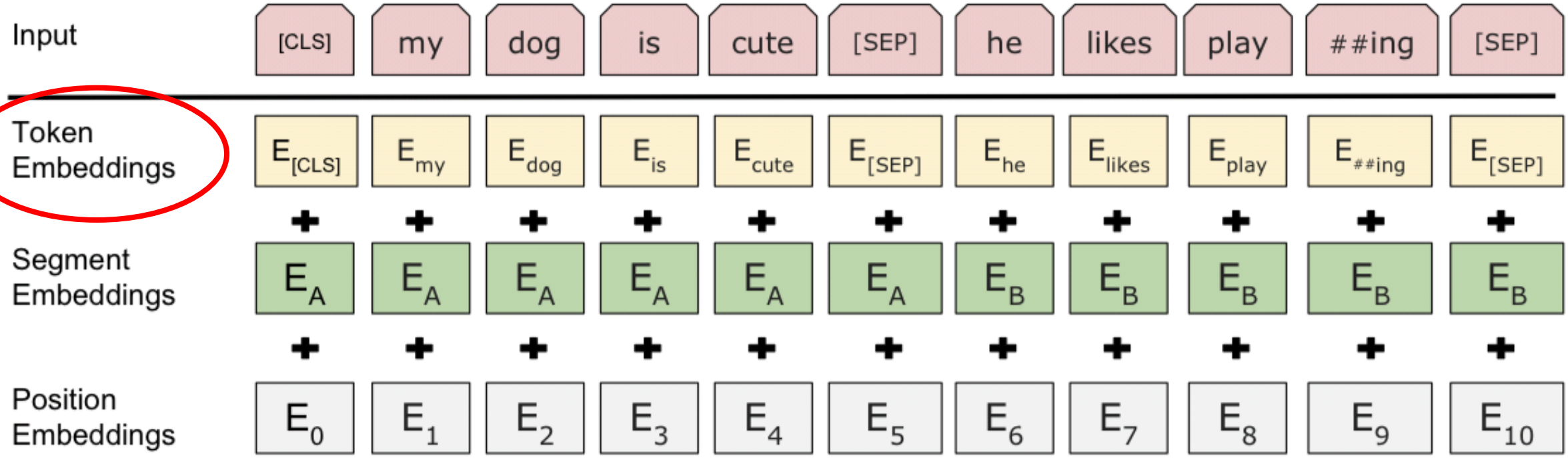
- Output Layer만 추가하면 간단하게 Fine-Tuning 가능 (=확장성 좋음)
- 11개의 주요 NLP 태스크 SOTA 달성
- Pre-training 단계에서만 Masked Language Model(이하 MLM) 사용

설명



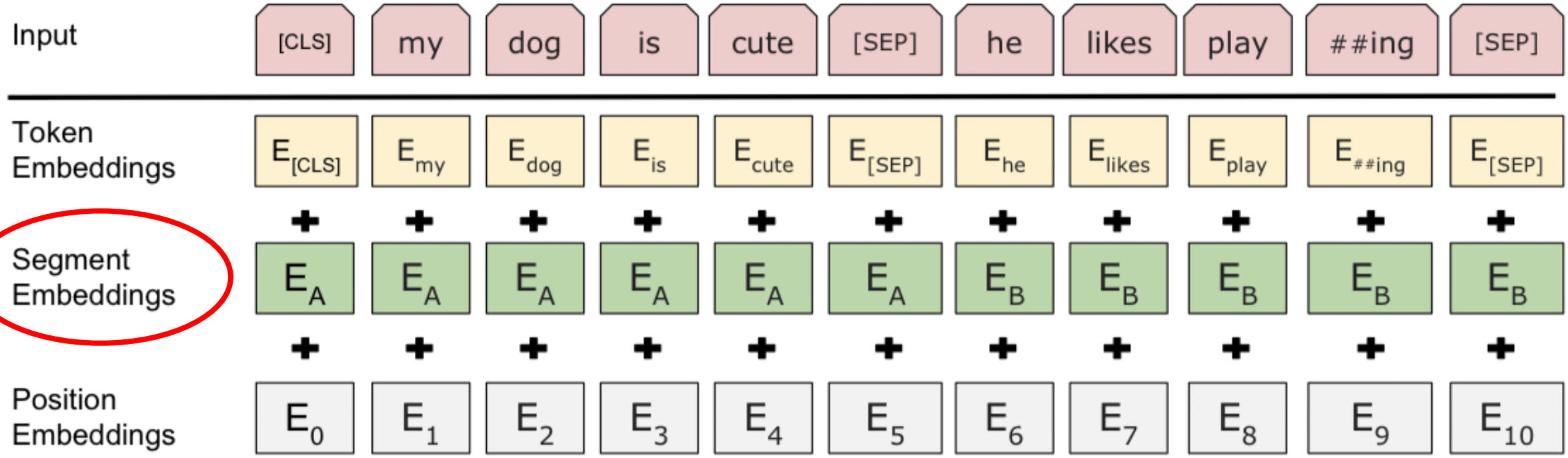
$$INPUT = Token\ Emb. + Segment\ Emb. + Position\ Emb.$$

설명



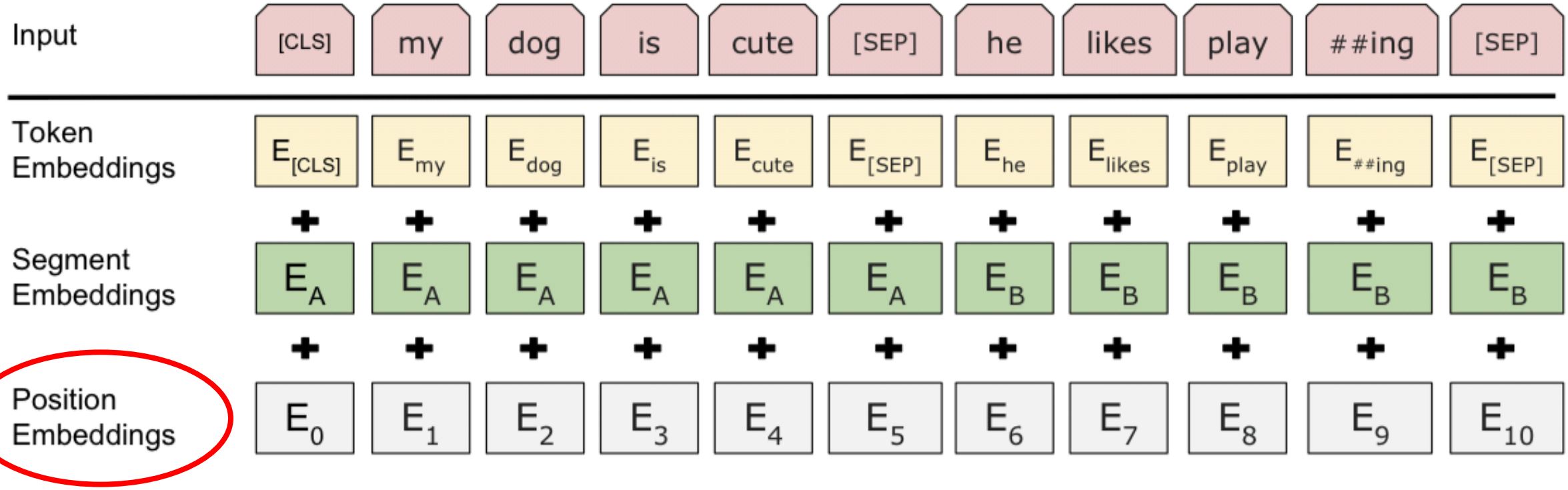
- WordPiece 임베딩 기법으로 토큰을 벡터화

설명



- 여러 문장이 입력으로 들어오기 때문에 문장 고유번호를 지정

설명



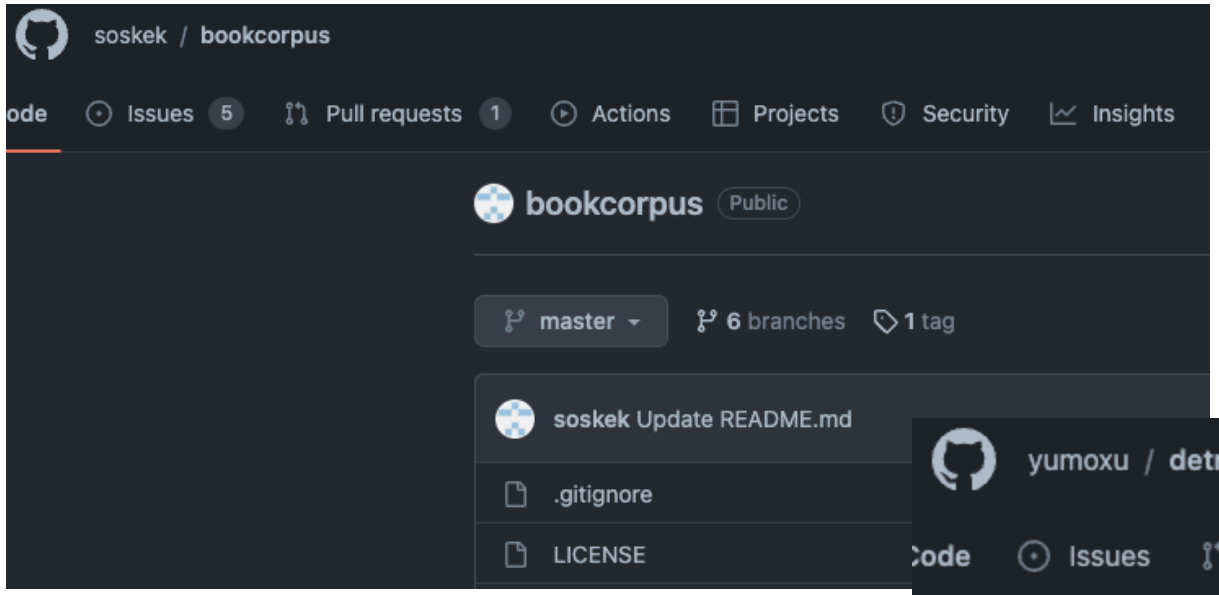
- Tranformer의 Encoder를 활용하기 때문에 거기서 사용한 Position 임베딩임

Pre-trained 단계

두 가지 Unsupervised Learning 진행

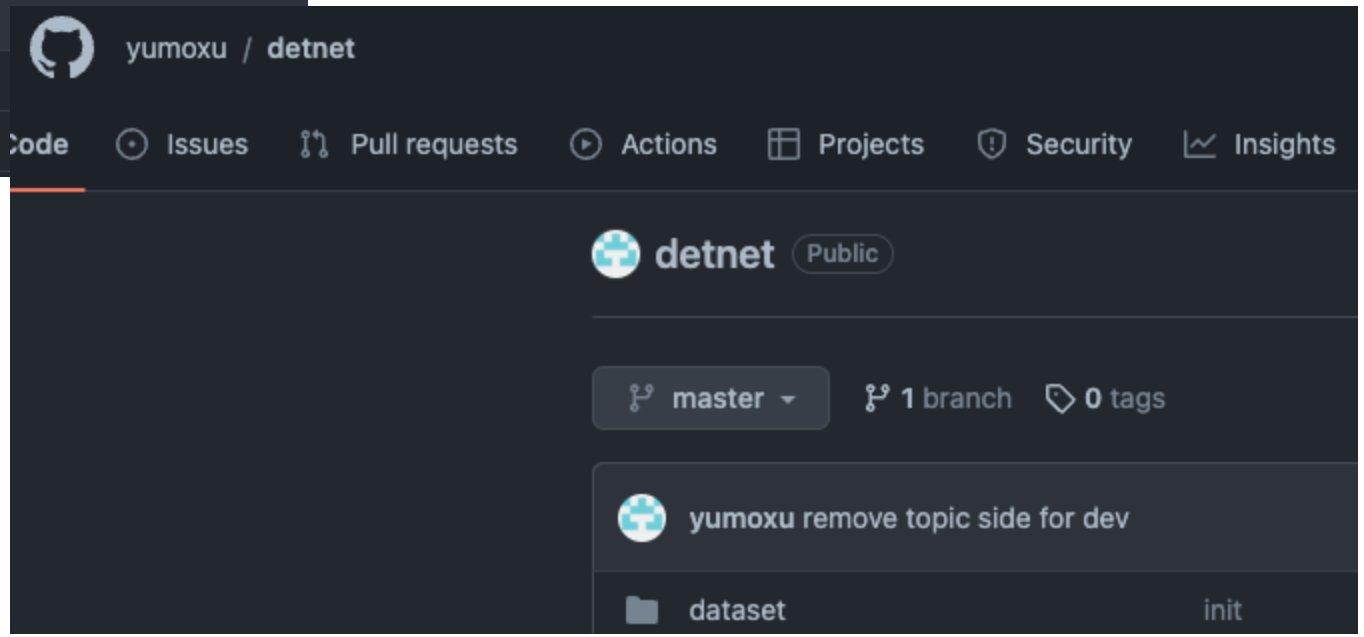
*Next Sentence Prediction

Pre-trained 단계

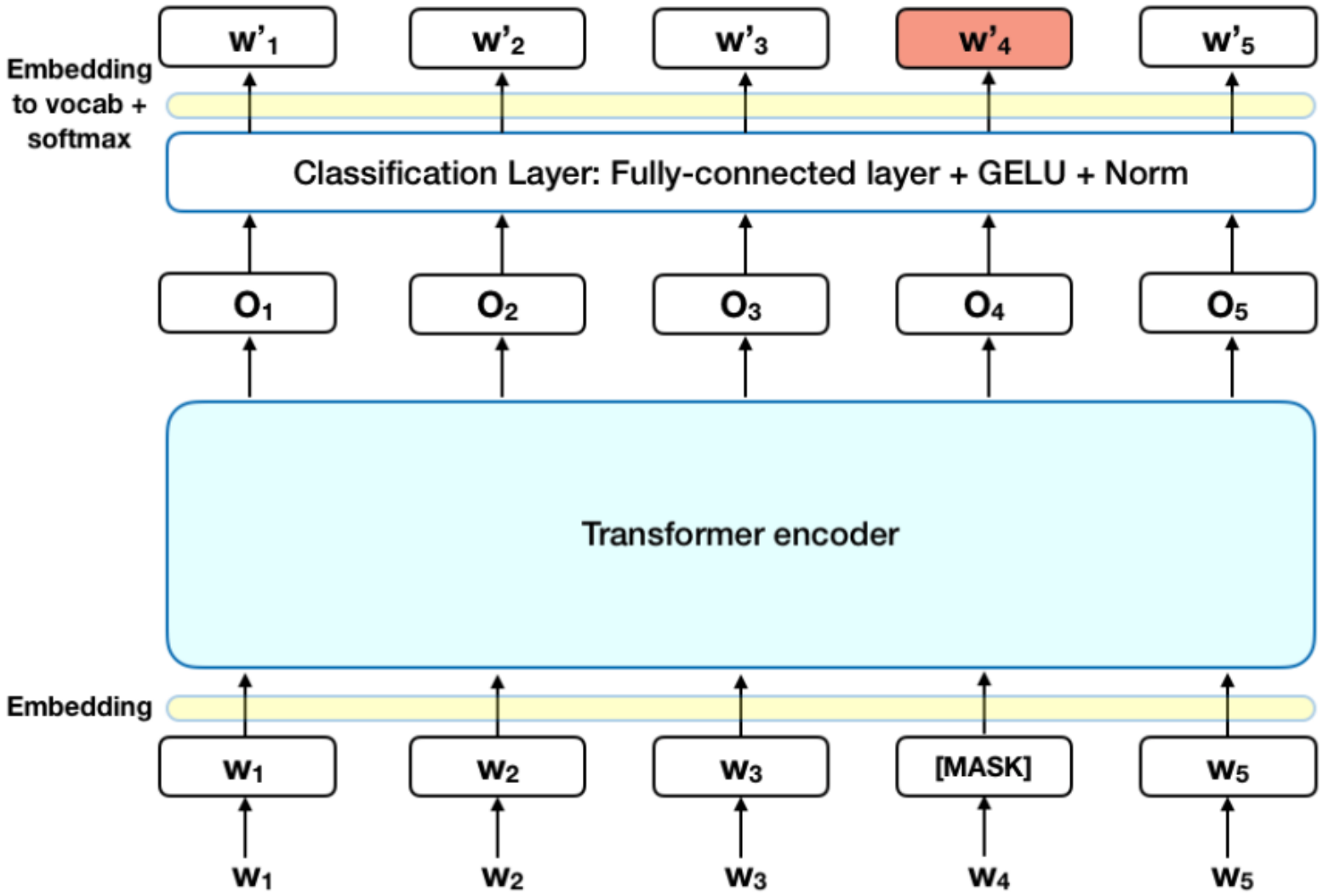


= 약 8억 개의 단어

약 25억 개의 단어 =

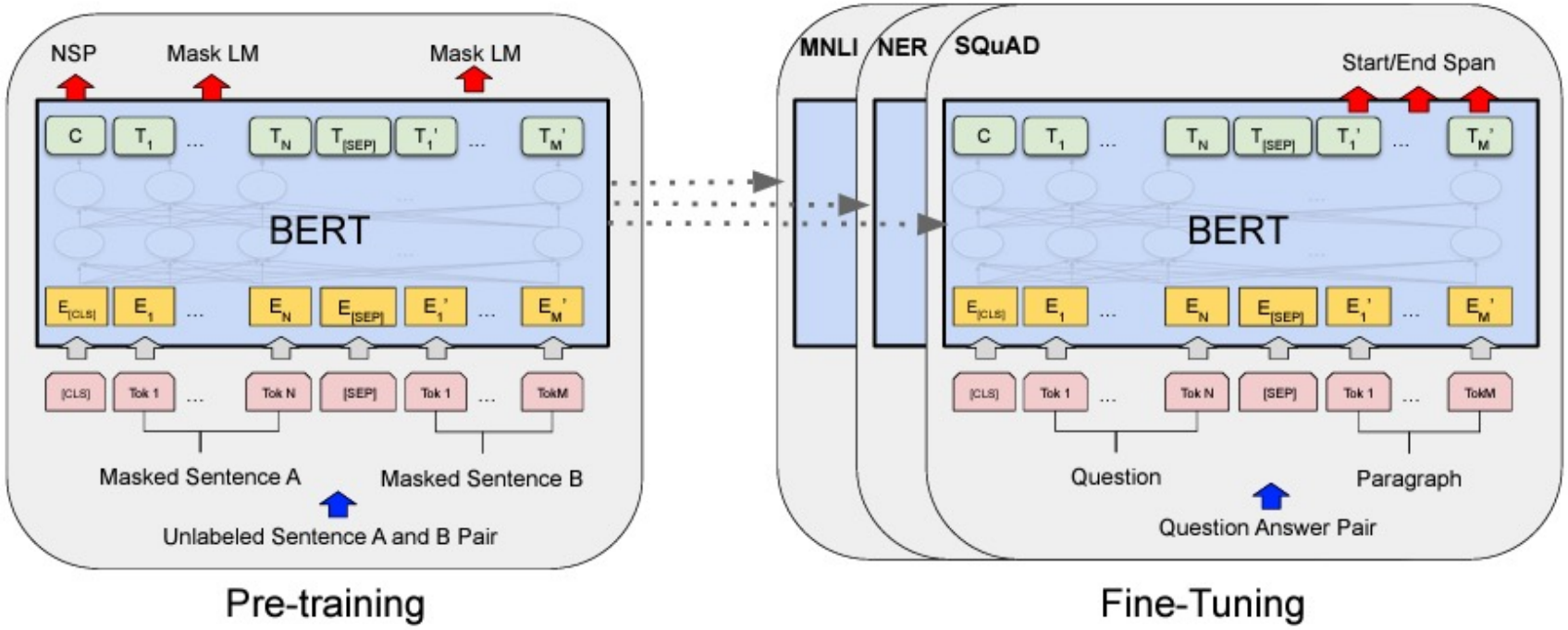


Pre-trained 단계 - MLM



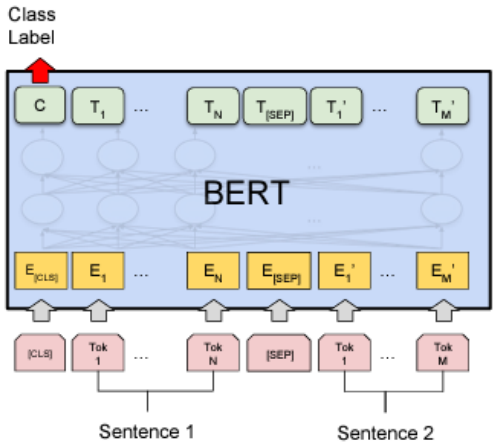
- 전체 데이터 중 15%만 적용
  - 80%: 마스크
  - 10%: 랜덤 변경
  - 10%: 그대로 두기
- 예시) my dog is hairy
  - my dog is [MASK]
  - my dog is apple
  - my dog is hairy
- 랜덤 변경은 10%나 되지만, 전체 데이터셋에서는 1.5% 정도라 데미지 없음

Pre-trained 단계 - NSP

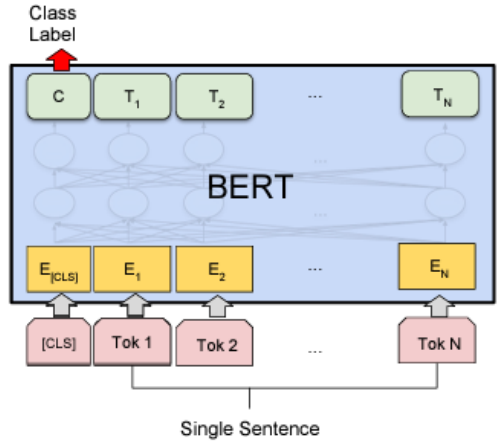


- 문장 A와 B가 입력이라면, 50%는 True Data, 50%는 False Data로 입력 데이터셋 구성
- CLS 토큰(output에서는 C)을 사용해 0 or 1을 예측
  - 0은 다음 문장 아님, 1은 다음 문장 맞음

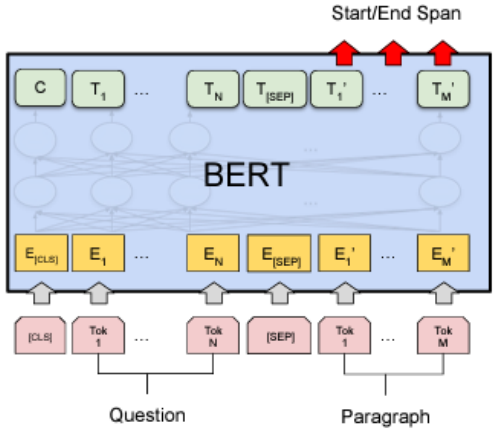
Fine-tuning 단계



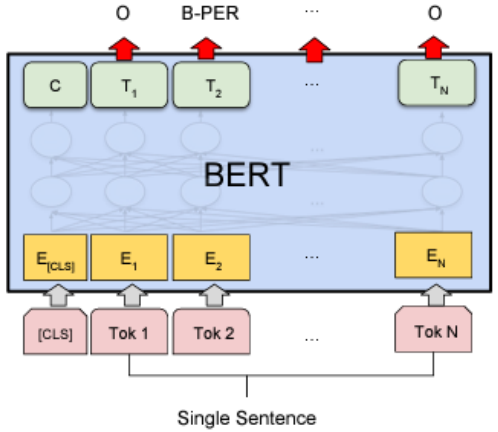
(a) Sentence Pair Classification Tasks: MNLI, QQP, QNLI, STS-B, MRPC, RTE, SWAG



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



(c) Question Answering Tasks: SQuAD v1.1



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

실험 결과

GLUE 데이터셋으로 학습

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**

실험 결과

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

- BERT base: 베이스라인 모델
- No NSP: Next Sentence Prediction을 수행 안 함
- LTR: Left to Right
- BiLSTM: Bidirectional LSTM

실험 결과

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	-

개체명 인식 TASK

하이퍼파라미터에 따른 모델 성능

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	base
12	1024	16	3.54	85.7	86.9	93.3	
24	1024	16	3.23	86.6	87.8	93.7	large

- L: Layer 개수
- H: Hidden size 크기
- A: Attention head 개수

감사합니다.