



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

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

We introduce a new language representation model called BERT, which stands for Bidirectional Encoder Representations from Transformers. Unlike recent language representation models, BERT is designed to pre-train deep bidirectional representations from unlabeled text by jointly conditioning on both left and right context in all layers.

 <https://arxiv.org/abs/1810.04805>

Abstract

1. Introduction

2. Related Work

2.1 Unsupervised **Feature-based** Approaches

2.2 Unsupervised **Fine-tuning** Approaches

2.3 Transfer Learning from Supervised Data(전이 학습)

3. BERT

Model Architecture

Input/Output Representations

3.1 Pre-training BERT

Task #1: Masked LM(MLM)

Task #2: Next Sentence Prediction (NSP)

Pre-training data

3.2 Fine-tuning BERT

5. Ablation Studies

5.1 Effect of Pre-training Tasks

5.2 Effect of Model Size

6. Conclusion

Appendix

A. Additional Details for BERT

A.3 Fine-tuning Procedure

B. Detailed Experimental Setup

B.1 Detailed Descriptions for the GLUE Benchmark Experiments

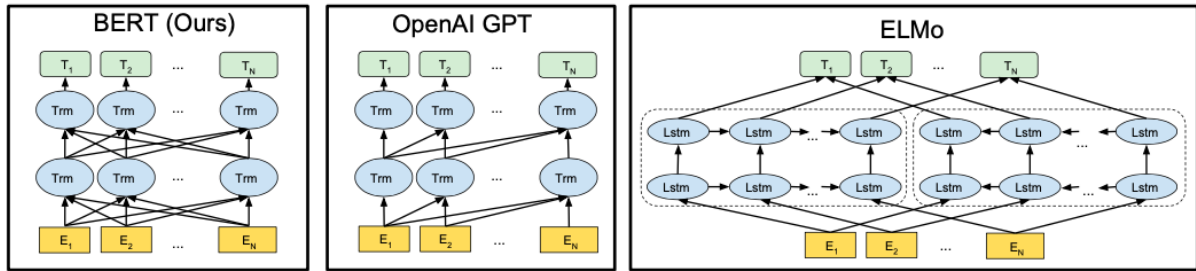
Abstract

BERT : **B**idirectional **E**ncoder **R**epresentations from **T**ransformers(트랜스포머의 양방향 인코더 표현)

→ transformer의 구조 중 인코더만을 사용

라벨이 없는 텍스트로 **모든** 레이어의 **좌우 문맥(left & right context)**을 따져가며 깊은 양방향 표현(deep bidirectional representations)을 사전학습 하도록 설계됨

사전학습된 BERT는 QA나 언어추론 등 대부분의 과제에 특화된 구조로 수정할 필요 없이 하나의 추가 출력 layer만 붙여도 SOTA 가능



- **BERT** : 양방향 Transformer 사용
- **OpenAI GPT** : left-to-right Transformer 사용
- **ELMo** : 독립적으로 훈련된 LTR, RTL LSTM 모델을 합침(downstream task의 특징을 생성하기 위해)

→ 3가지 모델 중 **BERT만 모든 층에서 좌, 우 문맥 모두에서** 같이 조건 설정

→ BERT & OpenAI GPT : **fine-tuning** approach, ELMo : **feature-based** approach

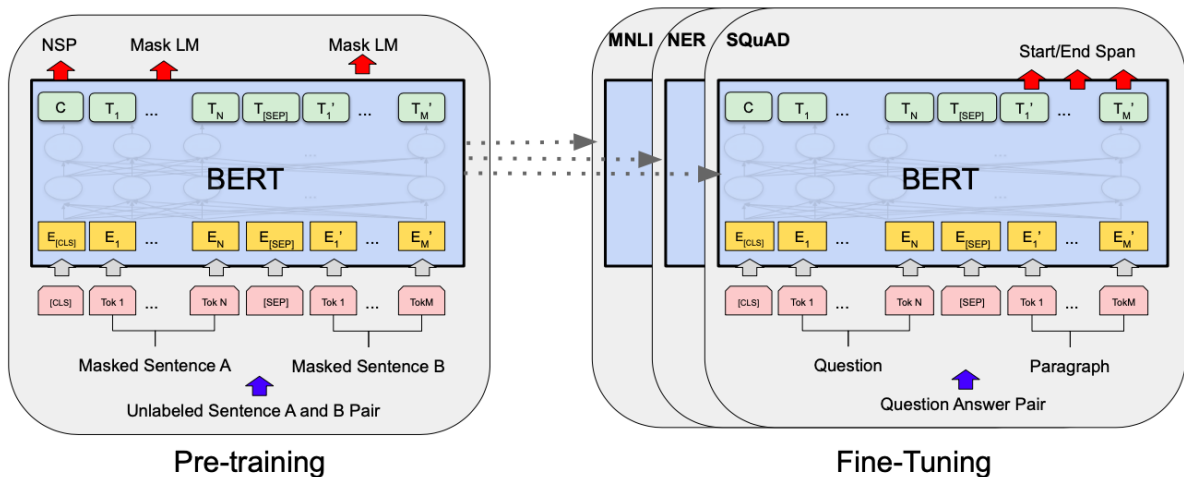
1. Introduction

언어 모델 사전훈련은 많은 NLP 과제에서 결과를 개선시키는데 효과적이라는 것을 보여줌

- 문장 수준 : natural language inference(NLI, 자연어 추론), paraphrasing
- 토큰 수준 : entity recognition(개체명 인식), question answering(QA)

사전훈련된 언어 표현을 downstream task에 적용하는 2가지의 기존 전략

- **feature-based**(특성 기반): 사전훈련된 표현을 포함한 **과제에 특화된 구조**를 사용(ex. ELMo)
 - 특정 과제를 수행하는 네트워크에 pre-trained language representation(사전 훈련된 언어 표현)을 추가적인 feature로 제공 → 두 개의 네트워크를 붙여서 사용한다고 보면 됨
- **fine-tuning**(미세 조정): 과제에 특화된 파라미터는 최소한으로 사용, 범용적인 사전 학습된 모델 사용
 - 특정 downstream task를 수행할 때, 학습을 통해 사전훈련된 파라미터를 **모두** 미세 조정 (ex. OpenAI GPT)
 - 범용적으로 pre-training 수행 후, task에 맞게 fine-tuning 해줌



→ 두 가지의 전략 모두 단방향 언어 모델을 사용하기 때문에, 사전학습하는 동안 같은 목적함수를 사용함

⇒ 주요한 한계는 표준 언어 모델이 단방향이고, 이것이 사전훈련동안 사용될 수 있는 구조의 **선택폭을 제한함**

ex. OpenAI GPT에서 left → right 구조를 사용해서 Transformer의 self-attention layer에서 모든 토큰이 **전의 토큰에 의존**할 수 밖에 없게 됨

⇒ 문맥을 양방향에서 통합하는게 중요한 QA와 같은 토큰 수준의 task를 위한 fine-tuning approach에서 매우 위험



BERT : masked language model(MLM) 사전훈련을 이용해 **단방향의 제약을 완화**함

→ MLM : 입력층의 **토큰 중 일부를 랜덤하게 mask** 설정 ⇒ 마스크 처리된 원래 단어의 id를 유추하는 것이 목적
⇒ left-to-right 언어 모델 사전훈련과 달리 좌, 우 문맥을 결합해 deep bidirectional Transformer를 사전훈련하게 함
추가로 **next sentence prediction(NSP)** task를 통해 문자-쌍 표현을 결합하는 사전훈련(jointly pre-train)함

본 논문이 기여하는 바

- 언어 표현에서 **양방향 사전훈련의 중요성 입증**
- 사전훈련된 표현은 **과제에 특화된 구조의 필요성을 줄여줌**을 입증
- 11개 NLP task에서 SOTA(the state of the art) 달성

2. Related Work

일반 언어 표현(general language representation)의 **사전훈련의 역사**

2.1 Unsupervised Feature-based Approaches

단어 임베딩은 현대의 NLP 시스템의 상당한 발전을 가져온 없어서는 안될 파트

→ **단어 임베딩** 벡터를 사전훈련하기 위해

- **left-to-right 언어 모델**을 사용하거나
- 좌우 문맥에서 **부정확한 단어로부터 정확한 단어를 구별**하는 방법 사용

이후, **문장/문단 임베딩**과 같은 세부적인 분야로 일반화됨

- 문장 표현을 훈련하기 위해 **다음에 올 문장의 후보 순위**를 매김
- 이전의 문장 표현이 주어졌을 때 **다음 문장의 단어를 left-to-right 방향으로 생성**
- **오토인코더의 노이즈를 제거**

ELMo와 이전 모델들은 기존의 단어 임베딩을 다른 차원으로 일반화

- left-to-right 언어 모델과 right-to-left 모델로부터 **문맥에 민감한(context-sensitive)** 특성을 추출
 - 각 토큰의 문맥적 표현은 **left-to-right 언어 모델과 right-to-left 모델의 연결**
 - QA, 감정 분석, 개체명 인식 등의 NLP task에서 SOTA 달성

2.2 Unsupervised Fine-tuning Approaches

unlabeled text로 사전훈련된 **단어 임베딩 파라미터 사용**

최근, 문맥상의 토큰 표현을 생성하는 **문장/문서 인코더**가 unlabeled text로 사전훈련

supervised downstream task에 맞춰 fine-tune됨

→ 장점 : **처음부터 훈련해야하는 파라미터가 거의 없음**

2.3 Transfer Learning from Supervised Data(전이 학습)

자연어 추론, 기계 번역과 같은 대규모 데이터셋이 있는 지도 학습에서 효율적인 변환 작업
컴퓨터 비전 연구 또한 대규모 사전훈련된 모델로부터 변환 훈련의 중요성을 입증

3. BERT

- **pre-training**

다른 사전 훈련 task에서 unlabeled data로 부터 훈련됨

- **fine-tuning**

처음에는 사전훈련된 파라미터로 시작

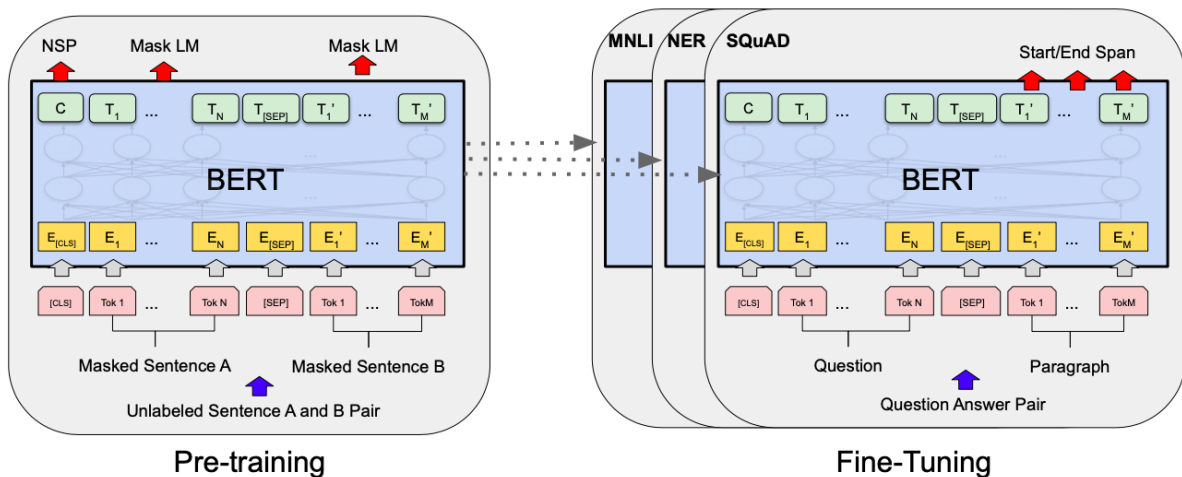
모든 파라미터는 downstream task의 labeled data를 사용해 fine-tune(미세 조정)

각각의 downstream task는 fine-tuned model을 분리(but 처음엔 사전훈련된 동일한 파라미터로 초기화)

→ BERT의 독특한 특징 : 여러 다른 task에도 통합된 구조 사용

즉, 사전 훈련된 구조와 downstream 구조 사이의 차이가 매우 적음

→ 출력층을 제외하고는 같은 구조를 사용



Model Architecture

논문 'Attention Is All You Need'의 Transformer를 기반으로 한 다층 양방향 Transformer 인코더

- BERT_base : L = 12, H = 768, A = 12, Total = 110M → OpenAI GPT와 비교하기 위해 같은 모델 사이즈
- BERT_large : L = 24, H = 1024, A = 16, Total = 340M
- (L : layer 수, H : 은닉층의 수, A : self-attention head의 수)

Input/Output Representations

BERT가 다양한 downstream task를 다루기 위해, 입력층 표현이 단일 sentence인지 한 쌍의 sentence인지를 토큰 sequence에서 분명하게 표현해야함

+) **sentence** : 연속적인 text의 임의의 범위 - 문장의 일부일 수도 있음(실제 언어학적인 문장 X)

sequence : BERT의 입력 토큰 시퀀스, 단일 문장이나 두 문장이 함께 패킹될 수 있음

3만개의 토큰 단어가 있는 WordPiece 임베딩 사용

- 모든 시퀀스의 첫 토큰은 항상 **CLS** 토큰
- **분류 task(감성 분석, 문장 관계 파악)** : CLS와 대응되는 최종 hidden state에서 집합 시퀀스 표현(aggregate sequence representation)으로 사용됨
 - sentence 쌍은 한 시퀀스로 패킹됨
 - sentence를 구분하기 위한 2가지 방법
 - **SEP 토큰을 사용**
 - 모든 토큰에 sentence A인지 B인지 나타내는 훈련된 임베딩을 추가(Segment Embedding)

주어진 토큰에 대해 입력층 표현은 token, segment(구획), position(위치) embedding의 합으로 구성됨

Input	[CLS]	my	dog	is	cute	[SEP]	he	likes	play	###ing	[SEP]
Token Embeddings	$E_{[CLS]}$	E_{my}	E_{dog}	E_{is}	E_{cute}	$E_{[SEP]}$	E_{he}	E_{likes}	E_{play}	$E_{\# \# \# ing}$	$E_{[SEP]}$
	+	+	+	+	+	+	+	+	+	+	+
Segment Embeddings	E_A	E_A	E_A	E_A	E_A	E_A	E_B	E_B	E_B	E_B	E_B
	+	+	+	+	+	+	+	+	+	+	+
Position Embeddings	E_0	E_1	E_2	E_3	E_4	E_5	E_6	E_7	E_8	E_9	E_{10}

3.1 Pre-training BERT

이 섹션에서 설명하는 두 가지 비지도 task를 통해 BERT를 사전 훈련

Task #1: Masked LM(MLM)

표준 조건부 언어 모델은 단방향으로만 훈련됨

양방향으로 훈련하기 위해 → 입력 토큰의 **일부분을 마스크 처리**하고, 그 **마스크를 예측**하게 함 → **Masked LM(MLM)**

전체 토큰 중 15%를 랜덤하게 마스크 처리

이것으로 양방향 사전 훈련 모델 획득 가능

but, downside에서는 **MASK** 토큰이 **fine-tuning 동안 나타나지 않기 때문에**, pre-training과 fine-tuning 사이에는 mismatch가 발생

→ 이를 줄이기 위해, 실제 **MASK** 토큰을 항상 **MASK** 토큰으로 대체하지 않음

1. 훈련 데이터 생성자는 토큰 위치의 15% 랜덤하게 선택
2. i번째의 토큰을 선택하면, i번째의 토큰을

- **80%** 확률로 **MASK** 토큰으로 변경

ex. my dog is hairy → my dog is **MASK**

- **10%** 확률로 **random** 토큰으로 변경

ex. my dog is hairy → my dog is **apple**

- **10%** 확률로 원래의 토큰으로 설정

ex. my dog is hairy → my dog is hairy

Masking Rates			Dev Set Results		
MASK	SAME	RND	MNLI Fine-tune	NER Fine-tune	NER 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

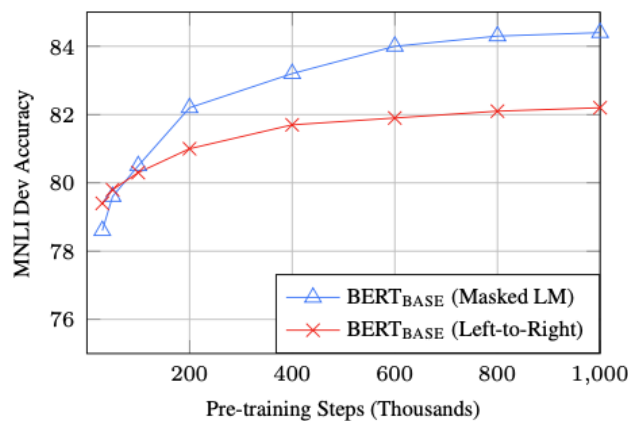
3. 그러면 T_i 는 cross entropy loss로 원본 토큰을 예측

→ 장점

- Transformer 인코더는 어느 단어를 예측해야하는지, 이 단어가 랜덤하게 대체되었는지 모르기 때문에, 모든 입력 토큰의 분포상의 문맥 표현을 유지하게 함
- 모든 토큰 중 1.5%(15%중 10% 설정)만 대체되기 때문에, 모델의 언어 이해 능력에 거의 손상을 입히지 않음

+) 표준 언어 모델 학습과 비교해, MLM은 각 batch에서 토큰의 15%에서만 예측을 하게 하므로, 모델이 수렴하는 데 **더 많은 사전 학습 단계**가 필요할 수 있음

하지만, 증가된 훈련 비용(리소스, 시간..)보다 MLM모델의 **경험적 개선**이 더 큼



Task #2: Next Sentence Prediction (NSP)

QA(Question Answering) & **NLI**(Natural Language Inference)와 같은 downstream task에서 가장 중요한 것은 언어 모델링으로 직접 포착되지 않은 **두 문장의 관계를 이해하는 것**

→ 문장 단위의 언어 모델링은 문장 사이의 관계를 이해하기 어려움

→ 문장 관계를 이해하게 하기 위해, 어느 단일 말뭉치에서 생성될 수 있는 다음 문장 예측(**Next Sentence Prediction, NSP**) task를 사전학습 시킴

사전 훈련 예시로 문장 A, B를 고를 때,

- **50%**의 확률로 B는 실제로 A 다음의 문장(**IsNext** 라벨 붙음)

Input : [CLS] the man went to [MASK] store [SEP] he bought a gallon [MASK] milk [SEP]

Label : IsNext

- 50%의 확률로 B는 말뭉치에서 랜덤한 문장(NotNext 라벨 붙음)

Input : CLS the man MASK to the store SEP penguin MASK are flight ##less birds SEP

Label : NotNext

→ 처리 간단해도 QA와 NLI에 매우 유용한 사전 훈련

→ NSP(next sentence prediction) task는 표현 학습과 연관되어 있음

하지만 이전 연구에서는 오직 문장 임베딩이 downstream task에 전달된 반면, BERT는 모든 파라미터를 전달해 최종 작업 모델 파라미터를 초기화

Pre-training data

사전 훈련 과정은 대부분 기존의 문헌을 따름

→ 말뭉치 사전 훈련을 위해 BooksCorpus (800M 단어), English Wikipedia (2,500M 단어) 사용

→ Wikipedia에서는 리스트, 테이블, 헤더는 무시하고, text 단락만 추출함

→ 순서가 섞인 문장 수준 말뭉치를 사용하지 않고, 문서 수준의 말뭉치를 사용(긴 연속적인 시퀀스를 추출하기 위해)

3.2 Fine-tuning BERT

Transformer의 self-attention mechanism이 BERT가 많은 downstream task를 모델링할 수 있게 하기 때문에(single text나 text pair 모두 가능), Fine-tuning은 간단함

text pair를 포함하는 문제에 대해, 일반적인 패턴은 양방향 교차 어텐션(bidirectional cross attention)을 적용하기 전에 text pair를 독립적으로 인코딩

→ BERT는 두 단계를 통합하기 위해 self-attention mechanism을 사용

→ 즉, self-attention으로 결합된 text pair를 인코딩하는 것은 두 문장 간의 양방향 교차 어텐션(bidirectional cross attention)을 효과적으로 포함

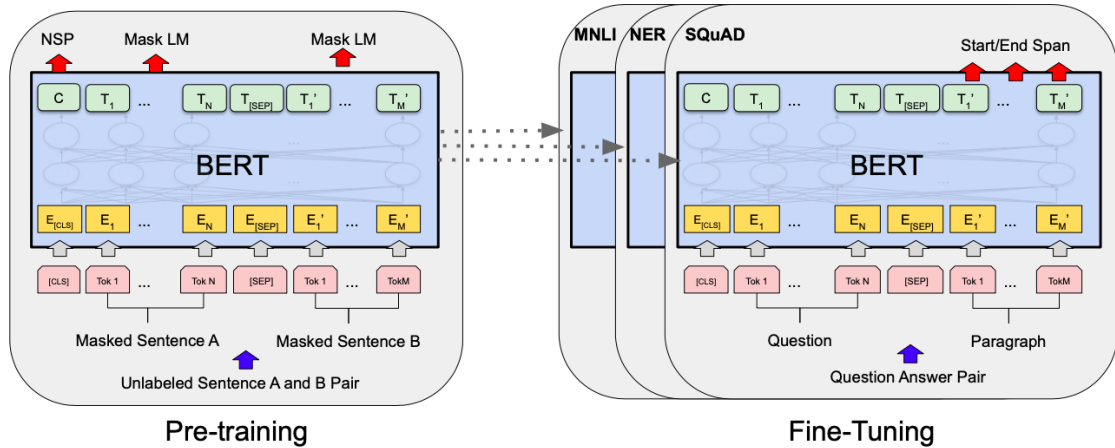
⇒ 각 task에서 task에 특화된 입력, 출력을 BERT에 간단하게 연결 + 모든 파라미터를 end-to-end로 fine-tune(미세 조정)

→ 입력에서 사전훈련된 문장 A, B는 아래와 유사

- paraphrasing된 문장 쌍
- 가설-전제 쌍
- 질문-답변 쌍
- 문서분류나 시퀀스 태깅에서의 퇴색된 문장-공집합 쌍

→ 출력에서

- 토큰 표현(nth token)은 시퀀스 태깅이나 QA와 같은 토큰 수준 task를 위해 출력층으로 넘어감
- CLS 표현은 감정 분석처럼 분류를 위해 출력층으로 넘어감



pre-training에 비해 fine-tuning은 연산량이 적음

→ 본 논문의 모든 결과는 single Cloud TPU에서 최대 1시간안에 재현될 수 있음(같은 사전훈련 모델로 실행시)

5. Ablation Studies

Tasks	Dev Set				
	MNLI-m (Acc)	QNLI (Acc)	MRPC (Acc)	SST-2 (Acc)	SQuAD (F1)
BERT _{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

No NSP : masked LM(MLM)을 사용한 양방향 모델을 next sentence prediction task 없이 학습

LTR & No NSP : next sentence prediction task 없이 left-to-right LM로 학습(OpenAI GPT 같이)

+ BiLSTM : fine-tuning시 LTR + No NSP의 위에 랜덤하게 초기화된 BiLSTM을 추가

5.1 Effect of Pre-training Tasks

→ ELMo처럼 LTR & RTL 모델을 합쳐서 훈련할 수도 있지 않을까?

- 단일 양방향 모델보다 연산 수가 2배 많음
- RTL모델은 질문에 대한 답을 설정할 수 없어서, QA와 같은 task에서는 직관적이지 않음
- 깊은 양방향 모델(deep bidirectional model)은 모든 층에서 좌,우 문맥을 사용할 수 있기 때문에 덜 효과적임

5.2 Effect 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

6. Conclusion

언어 모델을 사용한 전이 학습으로 비지도 사전 학습이 언어 이해 시스템의 필수적인 부분임을 입증

특히, 이러한 결과는 낮은 리소스 task에서도 deep unidirectional architecture의 이점을 얻게 함

본 논문의 주요한 기여점은 사전 학습 모델을 **다양한 범위의 NLP task에 적용**시킬 수 있도록, deep *bidirectional* architecture(깊은 양방향 구조)를 일반화한 것

Appendix

A. Additional Details for BERT

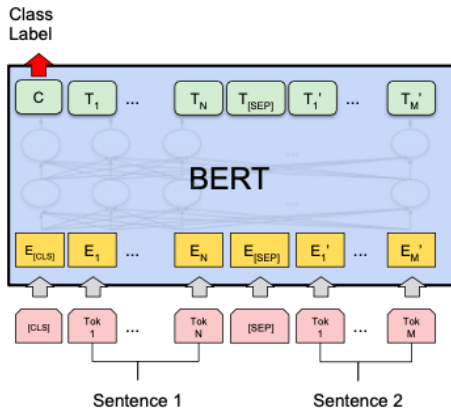
A.3 Fine-tuning Procedure

모든 task에서 잘 작동하는 값의 범위

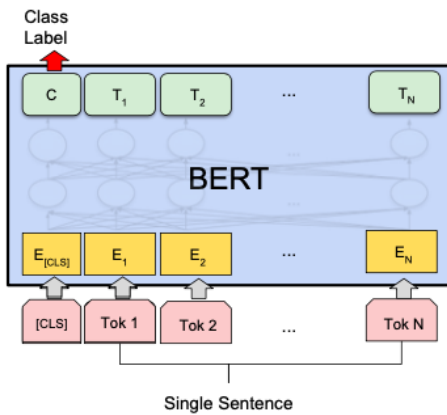
- Batch size: 16, 32
- Learning rate (Adam): 5e-5, 3e-5, 2e-5
- Number of epochs: 2, 3, 4

B. Detailed Experimental Setup

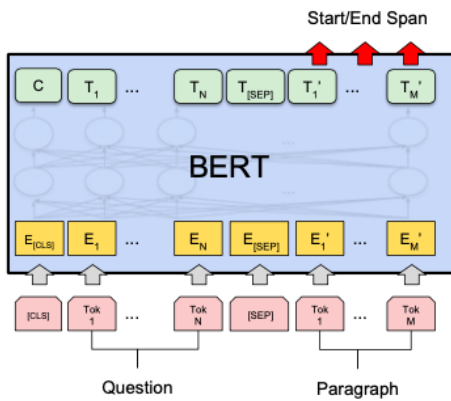
B.1 Detailed Descriptions for the GLUE Benchmark Experiments



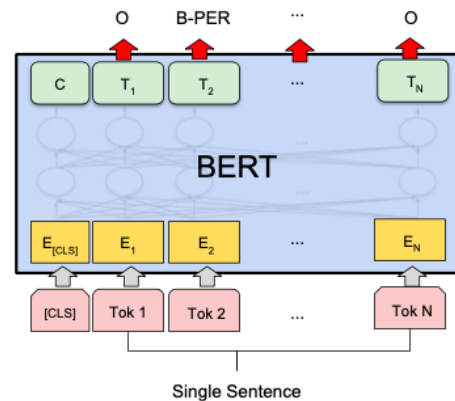
(a) Sentence Pair Classification Tasks:
MNLI, QQP, QNLI, STS-B, MRPC,
RTE, SWAG



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



(c) Question Answering Tasks:
SQuAD v1.1



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

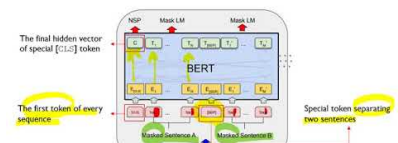
1. 한 쌍의 문장 분류 - ex. **문장 관계 파악** (A다음에 B가 자연스러운지, 두가지 문장의 유사도 파악 등)
2. 한 문장의 분류 - ex. **감성 분석** (긍정, 부정)
3. 질문에 대한 답변을 포함해 해당 **질문에 맞는 답** 추출
4. 각 토큰의 **개체명 인식**

• 참고 사이트

08-5: BERT

고려대학교 산업경영공학과 일반대학원 Unstructured Data Analysis 08-5: GPTBERT: Bidirectional Encoder Representation from Transformer <https://github.com/pilsung-kang/text-analytics>

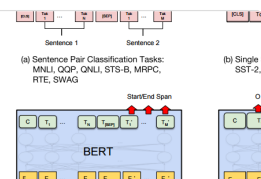
https://www.youtube.com/watch?v=IwtexRHoWG0



[바람돌이/딥러닝] BERT 논문 리뷰(Pre-training of Deep Bidirectional Transformers for Language Understanding)

안녕하세요. 오늘은 저번 Transformer, Attention is all you need 논문 리뷰 이후 나온 BERT 논문 리뷰를 하려고 합니다. BERT는 기존에 나온 transformer를 활용한 모델이며 논문의 내용을 간단하게 정리했습니다. Feature-based approach는 대표적으로 ELMO를 생각할 수 있습니다. Feature-based의 핵심은 어떠한 특정 task를 해결하기 위한 architecture를 구성하며 pre-trained

<https://m.blog.naver.com/PostView.nhn?blogId=winddori2002&logNo=222022178447&categoryNo=32&proxyReferer=https:%2F%2Fwww.google.com%2F>



Python, Machine & Deep Learning


이 글에서는 2018년 10월 Jacob Devlin 등이 발표한 BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding를 살펴보고자 한다. 어찌지 ELMo 를 매우 의식한 듯한 모델명이다. 코드와 사전학습(기학습)된 모델은 여기 에서 볼 수 있다. 중요한 부분만 적을 예정이므로 전체가 궁금하면 원 논문을 찾아 읽어보

🌸 [https://greeksharifa.github.io/nlp\(natural%20language%20processing\)%20/%20mns/2019/08/23/BERT-Pre-training-of-Deep-Bidirectional-Transformers-for-Language-Understanding/](https://greeksharifa.github.io/nlp(natural%20language%20processing)%20/%20mns/2019/08/23/BERT-Pre-training-of-Deep-Bidirectional-Transformers-for-Language-Understanding/)



BERT 논문정리

최근에 NLP 연구분야에서 핫한 모델인 BERT 논문을 읽고 정리하는 포스트입니다. 구성은 논문을 쪽 읽어다가며 정리한 포스트기 때문에 논문과 같은 순서로 정리하였습니다. Tmax Data AI 연구소에서 제가 진행한 세미나 동영상도 첨부합니다. BERT : Bidirectional Encoder Representations form Transformer 논문의 제목에서 볼 수 있듯

 <https://mino-park7.github.io/nlp/2018/12/12/bert-%EB%85%BC%EB%AC%B8%EC%A0%95%EB%A6%AC/?fbclid=IwAR3S-8iLWEVG6FGUVxoYdwQyA-zG0GpOUzVEsFBd0ARFg4eFXqCyGLznu7w>

