Week8_예습과제_김도희

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

💡 **Abstract**

BERT is designed to pretrain deep bidirectional representations from unlabeled text by jointly conditioning on both left and right context in all layers. As a result, the pre-trained BERT model can be finetuned with just one additional output layer to create state-of-the-art models for a wide range of tasks, such as question answering and language inference, without substantial taskspecific architecture modifications.

⇒ 사전 학습된 BERT 모델은 질문 응답이나 언어 추론과 같은 다양한 작업에 대해 추가적인 작업별 아키텍처 수정 없이 하나의 출력 층만 추가하여 SOTA 모델을 만들 수 있다.

1. Introduction

- **feature-based(e.g. ELMo)**

  additional feature 같은 pre-trained representations를 포함하는 task-specific architecture를 이용

- **fine-tuning(e.g. Generative Pre-trained Transformer)**

  minimal task-specific parameters, downstream tasks

⇒ 위의 두 방식 모두 한방향 언어모델에서 일반적인 언어표현을 학습하기 위해 쓰인다. 이러한 방식은 모델의 구조 선택에서 제한이 된다. 이 논문에서는 BERT를 제안하여 파인튜닝 기반 접근법을 개선한다. BERT는 마스킹 언어 모델이라는 사전 학습 목표를 사용하여 단방향성 제약

을 해결한다. 이 방법은 일부 입력 토큰을 무작위로 마스킹하고, 마스킹된 단어의 원래 어휘 ID를 문맥만을 기반으로 예측하도록 모델을 훈련한다.

2. Related Work

2.1 Unsupervised Feature-based Approaches

- 사전 학습된 단어 임베딩은 현대 자연어 처리 시스템의 중요한 구성 요소
- 학습 초기부터 임베딩을 새로 학습하는 것보다 성능 향상
- ELMo: 좌에서 우로와 우에서 좌로의 언어 모델로부터 문맥에 민감한 특징을 추출

2.2 Unsupervised Fine-tuning Approaches

- 문맥적 토큰 표현을 생성하는 문장 또는 문서 인코더가 비지도 데이터로부터 사전 학습된 후, 감독된 다운스트림 작업에 맞춰 파인튜닝되는 방식 연구
- 이러한 접근법의 장점은 처음부터 학습해야 하는 매개변수가 적은 것

2.3 Transfer Learning from Supervised Data

자연어 추론 이나 기계 번역과 같이 대규모 데이터셋을 사용하는 지도학습으로부터 전이학습

3. BERT

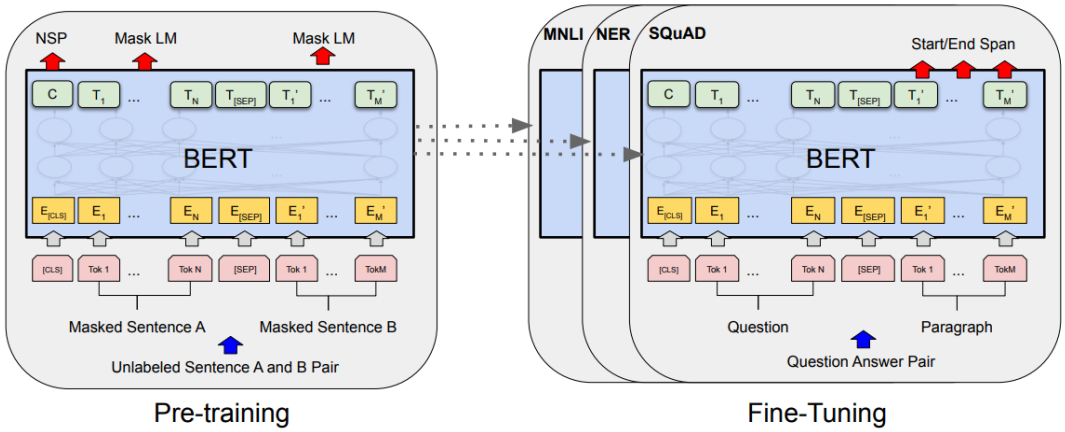
Two steps in framework

- **pre-training**

  다양한 비지도 학습 과제를 통해 unlabel 데이터에서 모델을 학습

- **fine-tuning**

  BERT 모델을 사전 학습된 매개변수로 초기화한 후, 다운스트림 작업의 레이블이 있는 데이터를 사용해 모든 매개변수를 파인튜닝



Model architecture

multi-layer bidirectional Transformer encoder

L : the number of layers

H: the hidden size

A: the number of self-attention heads

- $BERT_{BASE}$ (L=12, H=768, A=12, Total Parameters=110M)
- $BERT_{LARGE}$(L=24, H=1024, A=16, Total Parameters=340M)

Input/Output Representations

Input : 다양한 다운스트림 작업을 처리할 수 있도록 설계했다. 이 표현은 단일 문장과 문장 쌍(예: 질문과 답변)을 하나의 토큰 시퀀스로 명확하게 나타낼 수 있다

BERT는 30,000개의 토큰으로 구성된 WordPiece 임베딩을 사용한다. 각 시퀀스의 첫 번째 토큰은 항상[CLS]이다. 특수 분리 토큰 [SEP]으로 문장을 분리하고 각 토큰에 해당 문장이 A 또는 B에 속하는지를 나타내는 학습된 임베딩을 추가한다. 이 표현은 토큰 임베딩, 세그먼트 임베딩, 위치 임베딩의 합으로 구성된다.

3.1 Pre-training BERT

- **Task #1: Masked LM**

  입력 토큰의 일부를 무작위로 마스킹하고, 마스킹된 단어를 예측하는 방법을 사용(MLM)

이때 모든 다음 토큰을 예측하는 것이 아닌 입력에서 무작위 하게 몇 개의 토큰을 마스킹하고 이를 Transformer 구조에 넣어 주변 단어의 맥락으로 마스킹된 토큰만 예측한다.

BERT에서 MLM이 수행되는 과정은 우선 토큰 중 15%는 무작위로 [MASK]토큰으로 바꾸고 10%는 토큰을 무작위 단어로 바꾼다. 이 [MASK] 토큰은 pre-training에만 사용되고, fine-tuning시에는 사용되지 않는다.

- **Task #2: Next Sentence Prediction (NSP)**

  두 문장의 관계를 이해하기 위해 BERT의 학습 과정에서 두 번째 문장이 첫 번째 문장의 바로 다음에 오는 문장인지 예측하는 방식. BERT는 [SEP] 특수 토큰으로 문장을 분리하고 [CLS] 토큰의 출력은 간단한 분류 계층을 사용하여 2x1 모양의 벡터로 변환한다.

3.2 Fine-tuning BERT

the BooksCorpus (800M words) and English Wikipedia (2,500M words)을 이용하여 모델을 학습

BERT는 두 단계를 통합하여, self attetion mechanism을 통해 두 문장 간의 양방향 cross attetion를 효과적으로 포함합니다. 작업별로 입력과 출력을 BERT에 연결하고, 모든 매개변수를 end-to-end로 fine-tuning.

4. Experiments

4.1 GLUE

GLUE datasets

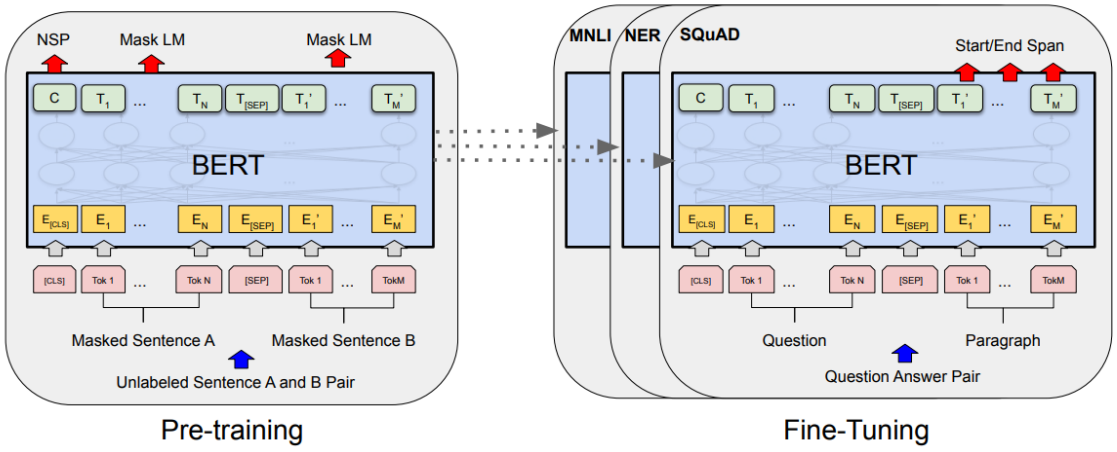
System	MNLI-(m/mm) 392k	QQP 363k	QNLI 108k	SST-2 67k	CoLA 8.5k	STS-B 5.7k	MRPC 3.5k	RTE 2.5k	**Average** -
Pre-OpenAI SOTA	80.6/80.1	66.1	82.3	93.2	35.0	81.0	86.0	61.7	74.0
BiLSTM+ELMo+Attn	76.4/76.1	64.8	79.8	90.4	36.0	73.3	84.9	56.8	71.0
OpenAI GPT	82.1/81.4	70.3	87.4	91.3	45.4	80.0	82.3	56.0	75.1
$\text{BERT}_{\text{BASE}}$	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**

Table 1: GLUE Test results, scored by the evaluation server (https://gluebenchmark.com/leaderboard). The number below each task denotes the number of training examples. The "Average" column is slightly different than the official GLUE score, since we exclude the problematic WNLI set.[8] BERT and OpenAI GPT are single-model, single task. F1 scores are reported for QQP and MRPC, Spearman correlations are reported for STS-B, and accuracy scores are reported for the other tasks. We exclude entries that use BERT as one of their components.

BERTBASE와 BERTLARGE는 모든 작업에서 모든 시스템을 능가하는 성능을 보였다.

4.2 SQuADv1.1

The Stanford Question Answering Dataset



입력 질문과 단락을 하나의 시퀀스로 결합하여 표현하며, 질문은 A 임베딩, 단락은 B 임베딩을 사용한다. 파인튜닝 시 시작 벡터 $S \in \mathbb{R}^H$와 끝 벡터 $E \in \mathbb{R}^H$를 도입하고 단락의 각 단어가 정답 범위의 시작 또는 끝일 확률은 소프트맥스 함수를 사용하여 계산한다.

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**

Table 2: SQuAD 1.1 results. The BERT ensemble is 7x systems which use different pre-training checkpoints and fine-tuning seeds.

→ 데이터의 추가적인 학습이 없이도 성능이 좋다는 점을 알 수 있다.

4.3 SQuADv2.0

The Stanford Question Answering Dataset, SQuADv1.1의 확장

답변이 없는 질문을 [CLS] 토큰 위치에 시작과 끝이 존재하는 것으로 처리하며, 정답 범위 위치 공간에 [CLS] 토큰의 위치를 포함하도록 확장했다. null 답변 점수와 최고 점수를 비교하여 답변 여부를 결정한다.

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

Table 3: SQuAD 2.0 results. We exclude entries that use BERT as one of their components.

→ +5.1의 F1 점수 향상

4.4 SWAG

The Situations With Adversarial Generations dataset

주어진 문장(A)과 가능한 연속 문장(B)을 포함하는 네 개의 입력 시퀀스를 구성했다. [CLS] 토큰의 표현과 점수를 계산해 소프트맥스 레이어로 정규화했다.

System	Dev	Test
ESIM+GloVe	51.9	52.7
ESIM+ELMo	59.1	59.2
OpenAI GPT	-	78.0
$\text{BERT}_{\text{BASE}}$	81.6	-
$\text{BERT}_{\text{LARGE}}$	**86.6**	**86.3**
Human (expert)†	-	85.0
Human (5 annotations)†	-	88.0

Table 4: SWAG Dev and Test accuracies. †Human performance is measured with 100 samples, as reported in the SWAG paper.

5. Ablation Studies

5.1 Effect of Pre-training Tasks

> demonstrate the **importance of the deep bidirectionality of BERT** by evaluating two pretraining objectives using exactly the same pretraining data, fine-tuning scheme, and hyperparameters as BERTBASE

⇒ BERT의 다양한 측면을 분석하여 성능에 미치는 영향

- NSP 제거

  QNLI, MNLI, SQuAD 1.1에서 성능이 크게 떨어진다.

- LTR & NSP 제거: 단방향으로 학습

  좌에서 우 단방향 모델은 모든 작업에서 MLM 모델보다 성능이 낮았다. 좌에서 우와 우에서 좌 모델을 개별적으로 학습하고 두 모델을 결합하는 방식도 고려할 수 있지만 비용이 두 배로 소요되며, QA에서 비직관적이다.

5.2 Effect of Model Size

레이어 수, 은닉 크기, self attetion head 수를 다르게 설정하여 여러 BERT 모델을 학습

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

Table 6: Ablation over BERT model size. #L = the number of layers; #H = hidden size; #A = number of attention heads. "LM (ppl)" is the masked LM perplexity of held-out training data.

⇒ 더 큰 모델은 모든 데이터셋에서 명확하게 정확도가 향상된다.

5.3 Feature-based Approach with BERT

특징 기반 접근법은 특정 작업에 최적화된 모델 구조가 필요하거나, 미리 계산된 표현을 활용해 실험을 더 빠르게 진행할 수 있다

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	-

Table 7: CoNLL-2003 Named Entity Recognition results. Hyperparameters were selected using the Dev set. The reported Dev and Test scores are averaged over 5 random restarts using those hyperparameters.

BERT를 CoNLL-2003 개체명 인식(NER) 작업에 적용했다. BERT에서 특징을 추출하여 고정된 임베딩으로 사용한 뒤, 무작위로 초기화된 BiLSTM을 추가해 결과를 비교했다. 특징기반의 접근법에서도 좋은 성능을 보여줌을 알 수 있다.

6. Conclusion

최근 언어 모델을 활용한 전이 학습의 실험적 성과들은 풍부한 비지도 사전 학습이 많은 언어 이해 시스템의 중요한 부분이 될 수 있음을 입증했다. 특히, 이러한 성과들은 리소스가 부족한 작업에서도 깊이 있는 단방향 아키텍처가 효과적일 수 있음을 보여준다.

이 논문의 주요한 점은 깊이 있는 양방향 아키텍처로 일반화하여, 동일한 사전 학습된 모델이 광범위한 자연어 처리 작업을 성공적으로 해결할 수 있음을 증명한 것이다. BERT는 복잡한 작업별 아키텍처의 필요성을 줄이며, 여러 문장 및 토큰 수준 작업에서 SOTA를 달성할 수 있는 강력한 언어 표현 모델이다.