Project 1
영어 문장 이진 분류기

Team 5
박석훈, 고명준, 손기훈, 서경원

목차

1. 프로젝트 개요

[활용 라이브러리 및 프레임워크]
영어 문장 이진분류

[프로젝트 목표]
1. 제한된 자원 내에서 효율적인 실험 및 기록
2. 재사용 가능성이 높은 코드 스니펫 확보

[성과 달성 지표]
Accuracy

[개발 환경]
Google Colab Pro + (언어: Python)

2. 프로젝트 진행 프로세스

구분	기간	활동	비고
팀 빌딩, 업무 분담	2/7(월)	착수 회의, 역할 분담	
베이스라인 코드 리뷰	2/7(월)~2/8(화)	베이스라인 코드 확인	각 역할 전원 참여
디버깅	2/8(화)~2/11(금)	파라미터 조정, 모델 변경	각 역할 전원 참여
논문 리뷰	2/8(화)~2/11(금)	BERT, RoBERTa 논문 리뷰	각 역할 전원 참여
결과 정리, 보고서 작성	2/10(목)~2/11(금)	결과 정리, 보고서 작성	각 역할 전원 참여

3. 프로젝트 팀 구성 및 역할

훈련생	담당 업무
박석훈(팀장)	- 프로젝트 총괄, 결과 보고서 작성
고명준	- 코드 작성, 디버깅
서경원	- 논문 리뷰, 자료 조사
손기훈	- 디버깅, 자료 조사

* 담당 업무는 주요 담당 업무이며, 전원 각 업무에 참여

BERT(Bidirectional Encoder Representations from Transformers)

- Google에서 개발한 '문맥을 고려한 임베딩 모델'
- 입력 데이터를 임베딩하는 임베딩 레이어(토큰, 세그먼트, 위치)
- - 토큰 임베딩: 각 토큰을 임베딩으로 변환
- - 세그먼트 임베딩: 문장을 구분하는 임베딩
- - 위치 임베딩: 문장 각 토큰에 대한 위치를 제공
- 마스크 언어 모델링(MLM), 다음 문장 예측(NSP)로 학습
- - MLM: 다음 단어 예측하도록 학습, 마스킹된 단어를 읽기 위해 양방향 독해
- - NSP: 두 문장을 제시하고, 두 번째 문장이 첫 문장 다음 문장인지를 예측

BERT(Bidirectional Encoder Representations from Transformers)

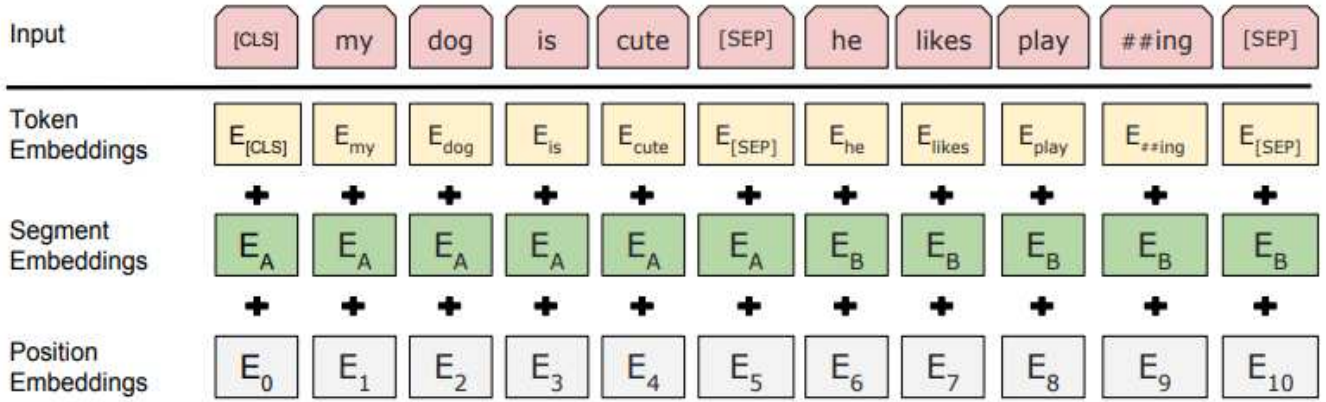

Figure 2: BERT input representation. The input embeddings are the sum of the token embeddings, the segmentation embeddings and the position embeddings.

RoBERTa(A Roubstly Optimized BERT Approach)

- 기존 BERT 모델에서 hyper-parameter 실험이 제대로 진행되지 않았다는 문제의식
- Replication study로서 BERT를 활용해 더 좋은 성능을 내고자 함
- BERT보다 훨씬 더 큰 양의 데이터셋을 학습(160GB: BookCorpus, CC-News, OpenWebText, Stories)
- 평가: GLUE, SQuAD, RACE(각 task 모두 BERT에 비해 더 좋은 성과를 냄)
- Pre-train을 오래할 수록 성능이 더 좋아지며, 학습 데이터의 양 역시 정확도를 높이는데 대단히 중요(다른 복잡한 요소에 충실하는 것보다)
- https://github.com/pytorch/fairseq (데이터셋)
- https://arxiv.org/pdf/1907.11692.pdf (논문)

RoBERTa(A Roubstly Optimized BERT Approach)

- Batch Size
- BERT 모델에서 batch size가 달라졌을 때의 성능을 비교함
- Batch size * step의 값이 같도록 설정.
- NSP 태스크를 하지 않고, MLM만 사용
- Large batch로 학습하는 것이 모델의 복잡도를 개선시키며 정확도에도 긍정적 영향을 미침

bsz	steps	lr	ppl	MNLI-m	SST-2
256	1M	1e-4	3.99	84.7	92.7
2K	125K	7e-4	**3.68**	**85.2**	**92.9**
8K	31K	1e-3	3.77	84.6	92.8

Table 3: Perplexity on held-out training data (*ppl*) and development set accuracy for base models trained over BOOKCORPUS and WIKIPEDIA with varying batch sizes (*bsz*). We tune the learning rate (*lr*) for each setting. Models make the same number of passes over the data (epochs) and have the same computational cost.

BERT vs RoBERTa

Model	data	bsz	steps	SQuAD (v1.1/2.0)	MNLI-m	SST-2
RoBERTa						
with BOOKS + WIKI	16GB	8K	100K	93.6/87.3	89.0	95.3
+ additional data (§3.2)	160GB	8K	100K	94.0/87.7	89.3	95.6
+ pretrain longer	160GB	8K	300K	94.4/88.7	90.0	96.1
+ pretrain even longer	160GB	8K	500K	**94.6/89.4**	**90.2**	**96.4**
$\text{BERT}_{\text{LARGE}}$						
with BOOKS + WIKI	13GB	256	1M	90.9/81.8	86.6	93.7
$\text{XLNet}_{\text{LARGE}}$						
with BOOKS + WIKI	13GB	256	1M	94.0/87.8	88.4	94.4
+ additional data	126GB	2K	500K	94.5/88.8	89.8	95.6

Table 4: Development set results for RoBERTa as we pretrain over more data (16GB → 160GB of text) and pretrain for longer (100K → 300K → 500K steps). Each row accumulates improvements from the rows above. RoBERTa matches the architecture and training objective of $\text{BERT}_{\text{LARGE}}$. Results for $\text{BERT}_{\text{LARGE}}$ and $\text{XLNet}_{\text{LARGE}}$ are from Devlin et al. (2019) and Yang et al. (2019), respectively. Complete results on all GLUE tasks can be found in the Appendix.

BERT vs RoBERTa

	MNLI	QNLI	QQP	RTE	SST	MRPC	CoLA	STS	WNLI	Avg
Single-task single models on dev										
$\text{BERT}_{\text{LARGE}}$	86.6/-	92.3	91.3	70.4	93.2	88.0	60.6	90.0	-	-
$\text{XLNet}_{\text{LARGE}}$	89.8/-	93.9	91.8	83.8	95.6	89.2	63.6	91.8	-	-
RoBERTa	**90.2/90.2**	**94.7**	**92.2**	**86.6**	**96.4**	**90.9**	**68.0**	**92.4**	**91.3**	-
Ensembles on test (from leaderboard as of July 25, 2019)										
ALICE	88.2/87.9	95.7	**90.7**	83.5	95.2	92.6	**68.6**	91.1	80.8	86.3
MT-DNN	87.9/87.4	96.0	89.9	86.3	96.5	92.7	68.4	91.1	89.0	87.6
XLNet	90.2/89.8	98.6	90.3	86.3	**96.8**	**93.0**	67.8	91.6	**90.4**	88.4
RoBERTa	**90.8/90.2**	**98.9**	90.2	**88.2**	96.7	92.3	67.8	**92.2**	89.0	**88.5**

Table 5: Results on GLUE. All results are based on a 24-layer architecture. $\text{BERT}_{\text{LARGE}}$ and $\text{XLNet}_{\text{LARGE}}$ results are from Devlin et al. (2019) and Yang et al. (2019), respectively. RoBERTa results on the development set are a median over five runs. RoBERTa results on the test set are ensembles of *single-task* models. For RTE, STS and MRPC we finetune starting from the MNLI model instead of the baseline pretrained model. Averages are obtained from the GLUE leaderboard.

BERT와 RoBERTa를 여러 조건에서 동시 수행하기

[Task]

1. BERT와 RoBERTa를 같은 조건에서 실행하여, 성과를 비교
2. 두 모델에 적용되는 조건에 변주를 주어, 조건이 달라졌을 때 성능 비교
3. 성능의 차이를 해석

4. 프로젝트 결과

Epoch = 1	Batch size			
	16	64	128	200
BERT	0.981	0.980	0.975	0.978
RoBERTa	0.977	0.981	0.978	0.979

실습: Google Colab Pro +
Score: Accuracy

4. 프로젝트 결과

Lr = 1e-05, Batch = 16	BERT	RoBERTa
Epoch = 1	0.981	0.984
Epoch = 3	0.977	0.986

Epoch = 1, Batch = 64	BERT	RoBERTa
Lr = 1e-05	0.980	0.981
Lr = 1e-06	0.973	0.975

Lr = 1e-05, Epoch = 5	BERT	RoBERTa
Batch = 64	0.983	0.983
Batch = 128	0.988	0.982
Batch = 200	0.982	0.982

실습: Google Colab Pro +
Score: Accuracy

4. 실험결과 정리

- BERT 모델의 경우 Batch Size(lr:1e-05)가 늘어날수록 1 Epoch 당 수행 시간이 감소(16: 35분, 128: 17분, 200: 15분)

- RoBERTa가 반드시 더 좋은 성능을 보장해주지는 않으며(실험결과 참고), 평균 3배 수행시간을 기록(Epoch 당 소요되는 시간 측정)

- '일반적으로' RoBERTa가 BERT에 비해 더 좋은 성능을 기록(Accuracy 기준, 11번 중 9번)

- 가장 좋은 지표는 BERT 모델(lr:1e-05, Batch:128, Epoch =5)일 때 기록(0.988)

5. 자체평가 및 보완

- Epoch, Batch, Learning rate에 따른 실험을 더 많이 수행했으면 BERT 모델과 RoBERTa 비교를 보다 정확히 할 수 있었을 것
- 본 Project와 논문에서 score를 측정하는 실험이 서로 다르기에, accuracy를 비교하는 것이 어떤 의미를 갖는지 고민 필요
- Best Score가 RoBERTa가 아닌 BERT에서 나왔다는 점
- RoBERTa의 경우 Epoch가 1이나 5가 아닌 3으로 했다면, 과적합 없이 더 좋은 성과를 거둘 수 있었을 것이라 판단

References

- Goorm NLP Course Material(강의, Notebook)
- 수다르산 라비찬디란(2021)『구글 BERT의 정석』한빛미디어
- arxiv.org/pdf/1810.04805.pdf
- arxiv.org/pdf/1907.11692.pdf
- https://stats.stackexchange.com/questions/153531/what-is-batch-size-in-neural-network
- https://sooftware.io/roberta/
- https://medium.com/dataseries/roberta-robustly-optimized-bert-pretraining-approach-d033464bd946