Parameter-Efficient Transfer Learning for NLP 리뷰

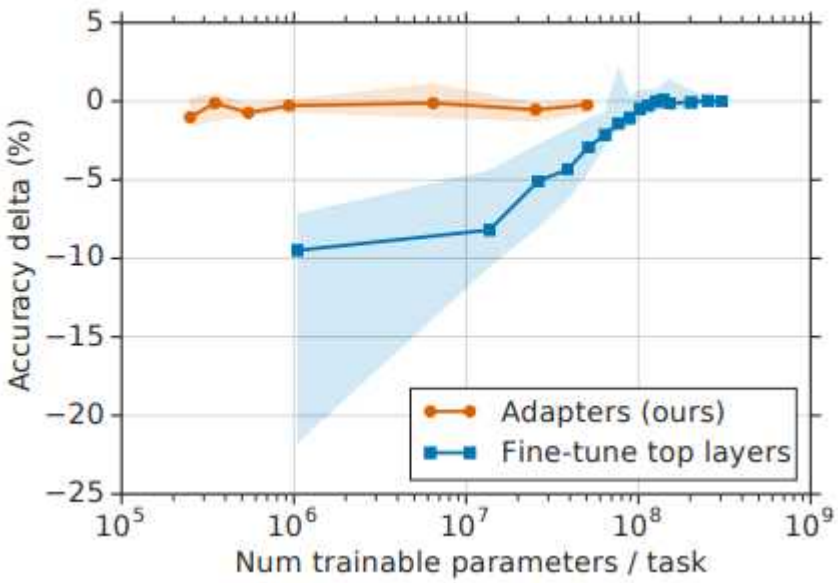
20기 김재훈

1. introduction

매번 새로운 태스크가 생길 때마다 모델을 학습하고 훈련하는 것은 매우 번거롭다. 따라서 새로운 작업에도 잘 작동하는 모델을 만들자는 것이 목적.

새로운 태스크가 생겨도 기존의 작업물을 잊지않고 조금만 파라미터를 학습시키고 추가해서 잘 작동하게 만들자!

파인튜닝도 하나의 방법이 될 수는 있지만 어댑터 튜닝 방법이 더 효율적이다. 태스크당 더 적은 파라미터로도 파인튜닝의 성능을 따라잡는다.



어댑터 튜닝은 멀티 태스크와 연속적인 학습과 관련있다.

1)멀티 태스크

멀티 태스크는 모든 태스크에 대해 동시에 접근 가능해야 하나 어댑터 튜닝은 그런 조건을 요구하지 않는다.

2) 연속적인 학습
 계속되는 작업 스트림을 통해 이전의 정보를 잊지 않고 완벽히 기억한다.

2. Adapter tuning for NLP

3가지의 key properties가 존재.

1) 좋은 성능
2) 순서대로 작업이 가능하나 데이터에 동시접근은 요구하지 않음.
3) 새 태스크가 생길 때 작은 파라미터만 추가

이러한 properties를 충족시키기 위해 bottleneck adaper module을 제안.

어댑터 모듈을 튜닝하는 것은 이전 태스크에 대한 파라미터에 새로운 파라미터를 추가하는 것임.

vanila 파인튜닝은 미래의 태스크와 이전의 태스크에 대한 차이로 인해 로스가 발생. 어댑터 모듈은 이전의 태스크를 이용해 좀 더 일반적인 구조로 수정

또한 파인튜닝은 새로운 파라미터와 기존의 가중치가 함께 학습되지만 어댑터 튜닝은 기존의 파라미터는 고정되기 때문에 다양한 태스크에 대해 파라미터 공유가 가능하다.

어댑터 모듈의 두 가지 특징이 있는데 적은 파라미터와 near identity initialization 이다.

어댑터 모듈은 작은 개수의 레이어를 가지고 있는데 점점 크기를 키워나간다.

2.1 Instantiation for transformer networks

어댑터 모듈은 bottle neck 구조를 사용해 파라미터의 수에 제약을 둔다.

d차원을 m차원으로 만들고 다시 d차원으로 사영시키려면 2md + m + d개의 파라미터가 필요하다. m << d 라는 전제를 각 태스크마다 걸어둠으로써 파라미터의 효율성과 성능을 조절했다. 논문에서는 0.5~8%를 사용

3. Experiments

사전 훈련된 bert transformer모델을 사용. 분류 문제를 수행

GLUE란 무엇인가?

GLUE는 분류 추론 유사도 평가 등 다양한 작업을 통해 NLP모델의 성능을 테스트 하는 것.

다양한 태스크에 맞게 일반적인 모델을 만드는 것이 이 논문의 목적이므로 이게 성능이 좋아야 한다.

Parameter-Efficient Transfer Learning for NLP

	Total num params	Trained params / task	CoLA	SST	MRPC	STS-B	QQP	MNLI_{m}	MNLI_{mm}	QNLI	RTE	Total
$\text{BERT}_{\text{LARGE}}$	9.0×	100%	60.5	94.9	89.3	87.6	72.1	86.7	85.9	91.1	70.1	80.4
Adapters (8-256)	1.3×	3.6%	59.5	94.0	89.5	86.9	71.8	84.9	85.1	90.7	71.5	80.0
Adapters (64)	1.2×	2.1%	56.9	94.2	89.6	87.3	71.8	85.3	84.6	91.4	68.8	79.6

점수를 비교해 보았을 때 기존 모델보다 어댑터가 성능이 조금 안 좋았다. 뭔가 더 보강이 필요해 보인다.

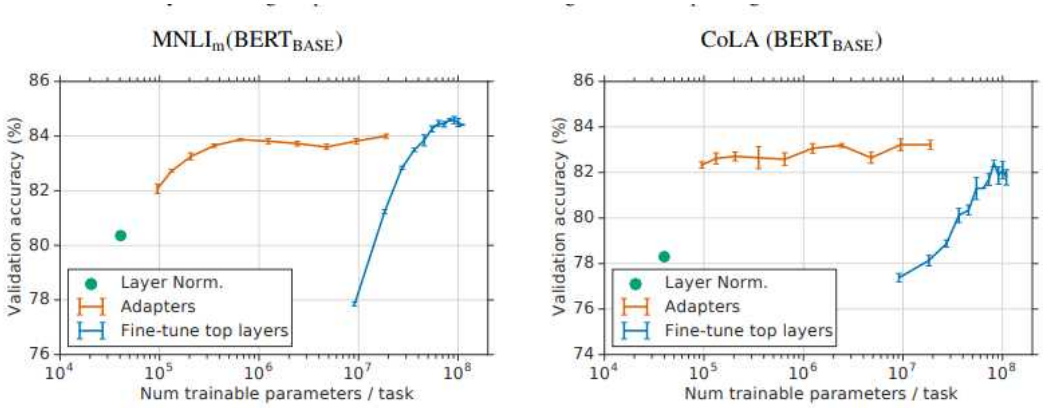
그 이유는 데이터 셋마다 필요한 어댑터의 크기가 달랐기 때문

Dataset	No BERT baseline	$\text{BERT}_{\text{BASE}}$ Fine-tune	$\text{BERT}_{\text{BASE}}$ Variable FT	$\text{BERT}_{\text{BASE}}$ Adapters
20 newsgroups	91.1	92.8 ± 0.1	92.8 ± 0.1	91.7 ± 0.2
Crowdflower airline	84.5	83.6 ± 0.3	84.0 ± 0.1	84.5 ± 0.2
Crowdflower corporate messaging	91.9	92.5 ± 0.5	92.4 ± 0.6	92.9 ± 0.3
Crowdflower disasters	84.9	85.3 ± 0.4	85.3 ± 0.4	84.1 ± 0.2
Crowdflower economic news relevance	81.1	82.1 ± 0.0	78.9 ± 2.8	82.5 ± 0.3
Crowdflower emotion	36.3	38.4 ± 0.1	37.6 ± 0.2	38.7 ± 0.1
Crowdflower global warming	82.7	84.2 ± 0.4	81.9 ± 0.2	82.7 ± 0.3
Crowdflower political audience	81.0	80.9 ± 0.3	80.7 ± 0.8	79.0 ± 0.5
Crowdflower political bias	76.8	75.2 ± 0.9	76.5 ± 0.4	75.9 ± 0.3
Crowdflower political message	43.8	38.9 ± 0.6	44.9 ± 0.6	44.1 ± 0.2
Crowdflower primary emotions	33.5	36.9 ± 1.6	38.2 ± 1.0	33.9 ± 1.4
Crowdflower progressive opinion	70.6	71.6 ± 0.5	75.9 ± 1.3	71.7 ± 1.1
Crowdflower progressive stance	54.3	63.8 ± 1.0	61.5 ± 1.3	60.6 ± 1.4
Crowdflower US economic performance	75.6	75.3 ± 0.1	76.5 ± 0.4	77.3 ± 0.1
Customer complaint database	54.5	55.9 ± 0.1	56.4 ± 0.1	55.4 ± 0.1
News aggregator dataset	95.2	96.3 ± 0.0	96.5 ± 0.0	96.2 ± 0.0
SMS spam collection	98.5	99.3 ± 0.2	99.3 ± 0.2	95.1 ± 2.2
Average	72.7	73.7	74.0	73.3
Total number of params	—	17×	9.9×	1.19×
Trained params/task	—	100%	52.9%	1.14%

Table 2. Test accuracy for additional classification tasks. In these experiments we transfer from the $\text{BERT}_{\text{BASE}}$ model. For each

추가적인 실험을 진행했다. 파인튜닝된 것이 성능이 조금 더 좋긴하지만 파라미터의 수가 어댑터 모델보다 월등히 많다.

성능이 아주아주 조금 나아진다하더라도 모델이 많이 무겁다면 가볍지만 성능은 차이가 별로 안나는 어댑터 모듈이 더 좋은 평가를 받을 수 있다.



MNLI라는 추론 문제와 CoLA라는 문법관련 문제를 해결하기위해 필요한 파라미터 수는 어댑터가 훨씬 적지만 성능은 아주아주 유사하다.