Parameter-Efficient Transfer Learning for NLP 리뷰

review type	논문
Status	Not started
keywords	

Abstract

- Downstream task를 할 때 매번 task에 대해 train해야 한다는 비효율성 문제
- Ardapter **module**을 사용하여 위 문제를 해결한다.
- BERT transformer에서 파라미터를 100% 훈련한 fine-tuning model vs. Adapter module을 사용해서 파라미터를 3.6%사용한 모델
- 성능은 차이 오직 0.4%차이

Introduction

목표

- 새로운 작업에 대해 매번 새로운 모델을 훈련하는 대신, 새로운 작업 모두에 대해 general하게 잘 작동하는 시스템을 개발
- **compact** 하고 **extensible**한 downstream model을 만드는 것
- 즉, 작업당 조금의 파라미터만을 추가해서 문제를 해결하는 모델 + 이전 작업을 잊지 않고 점진적으로 새로운 문제를 풀 수 있도록 훈련되는 모델이.

Transfer learning

Feature-based와 **Fine-tuning 기법 2개**

Feature-based

- 모델의 출력을 특징(feature)로 사용하여, 다른 모델의 입력으로 사용
- 사전 훈련된 모델을 수정 x, 추가 학습 동안 파라미터가 변경 x

Fine-Tuning

- 사전 훈련된 모델 전체 or 일부를 새로운 작업에 맞게 추가학습
- 사전 훈련된 모델의 파라미터가 새로운 데이터셋에 맞게 조정됨
- 사전 훈련된 모델이 이미 갖고 있는 지식을 기반으로 새로운 작업에 대한 성능을 최대화할 때 사용
    - 특히 작은 데이터셋으로 작업할 때 유용
    - 최근 연구에 따르면 Fine-tuning이 좀 더 **parameter efficient**.

Adapter tuning for NLP

Adapter tuning의 3가지 주요 특징

1. 좋은 성능
2. 순차적으로 작업에 적용가능, 하지만 동시 엑세스는 요구되지 않음
3. 작업당 소수의 매개변수만 추가함

이를 만족하기 위해 새로운 **bottleneck adapter module**을 제안

Adapter module은 downstream 작업에 대해 사전 훈련된 네트워크의 용도를 변경하기 위해서 **좀 더 general한 아키텍쳐 수정**을 진행

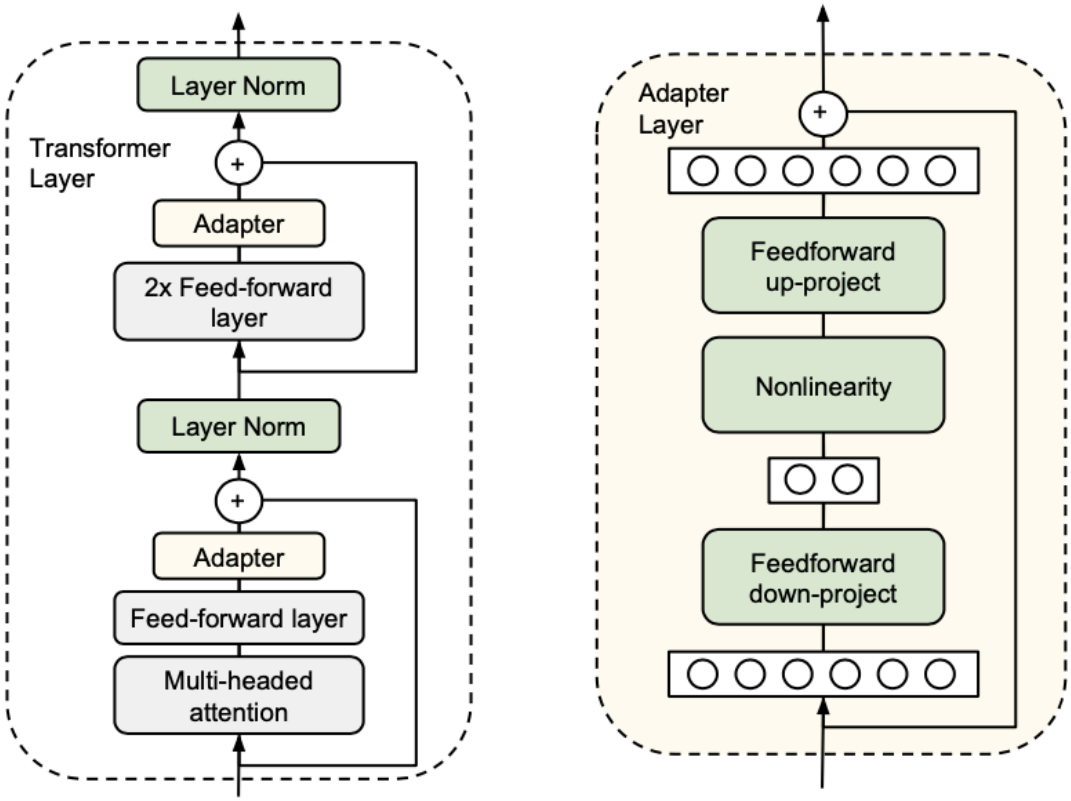
Adapter Module은 두가지 main feature

1. 적은 파라미터2. Near-identity 초기화

- Near-identity: 입력을 거의 그대로 출력하도록 설정하는 것
- 훈련 시작시, 원래의 네트워크가 거의 영향받지 x

Instantiation for Transformer Networks

Transformers 아키텍처 기반으로 Adapter tuning을 진행했다.



Adapter Layer

- 파라미터의 수에 제약을 두기 위해서 **bottle-neck**구조를 제안
- 원래의 d 차원을 m차원으로 비선형을 적용하여 투영(Projection)한 다음 다시 d차원으로 project
- $m << d$로 설정한다. m은 bottle-neck dimension
- m이 작아지면 작아질 수록 전체 파라미터들도 작아짐. 논문 저자는 0.5%~8%로 파라미터의 수를 조절.
- **skip-connection 진행**

Experiments

Experimental Settings

- 사용한 네트워크 : pre-trained BERT Transformer network
- 최적화 : Adam (10%씩 학습률 증가한 다음 어느순간 선형적으로 0으로 감소)
- 4 Google Cloud TPU, batch size는 32
- 다양한 하이퍼파라미터 조합을 사용 후 모델 선택

GLUE benchmark

- 훈련 모델은 pre-trained BERT-large model
- **Hyperparameter sweep**
  - learning rate : {3·10−5,3·10−4,3·10−3}
  - epoch : {3,20}
  - fixed adapter size(number of units in the bottleneck) : {8, 64, 256}

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
BERT_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

Table 1. Results on GLUE test sets scored using the GLUE evaluation server. MRPC and QQP are evaluated using F1 score. STS-B is evaluated using Spearman's correlation coefficient. CoLA is evaluated using Matthew's Correlation. The other tasks are evaluated using accuracy. Adapter tuning achieves comparable overall score (80.0) to full fine-tuning (80.4) using $1.3\times$ parameters in total, compared to $9\times$. Fixing the adapter size to 64 leads to a slightly decreased overall score of 79.6 and slightly smaller model.

- 어댑터는 평균적으로 80.0점이라는 GLUE점수를 얻었다.
- Full fine-tuning은 80.4점

실험 결과 요약

- 데이터 셋 마다 최적 어댑터의 크기가 달랐다.
- MNLI에는 256, RTE에서는 8이 선택되었다.
- 항상 크기를 64로 선택하면 정확도가 79.6으로 줄어들었다.

- BERT total number of parameter 와 비교해서

Fine-tuning : 9X

Adapter : 1.3X

의 파라미터 수의 차이

Additional Classification Tasks

- 훈련 모델 : *BERTBASE* (12 layer로 구성)
- 훈련 예제의 숫자 : 900 ~ 330k
- 클래스 범위 : 2 ~ 157
- 평균 텍스트 길이 : 57 ~ 1.9k

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 task and algorithm, the model with the best validation set accuracy is chosen. We report the mean test accuracy and s.e.m. across runs with different random seeds.