ELECTRA

이름 / 학번 : 황민규 / V2022117

전공 : 메타버스 테크놀로지

기존 BERT를 비롯한 많은 language model들은 입력을 [mask] token으로 치환하고 이를 치환 전의 original token으로 복원하는 masked language modeling 태스크를 통해 pre-training을 한다. 하지만 이런 모델들은 학습 시 상당히 많은 계산량을 필요로 하다는 단점이 있다.

ELECTRA는 모델의 정확도와 함께 학습의 효율성에도 주목을 하였다. 해당 모델에서는 학습 효율을 향상시키기 위해 Replaced Token Detection (RTD)이라는 새로운 pre-training 태스크를 사용했으며, 이를 통해 ELECTRA는 보다 빠르고 효과적으로 학습이 가능하다. 결과적으로 ELECTRA는 모델 크기, 데이터, 컴퓨팅 리소스가 동일한 조건에서 기존 BERT의 성능을 능가했다.

1. Introduction

현재 state-of-the-art representation learning 기법은 일종의 denoising autoencoder 학습이다. 주로 입력 시퀀스의 토큰 중 15%를 마스킹을 하고, 이들을 복원하는 masked language modeling (MLM) 방법을 통해 학습을 진행한다. MLM은 양방향 정보를 고려할 수 있다는 장점이 있지만 몇 가지 단점도 가지고있다.

1. 하나의 샘플에 대해 15%만 학습을 진행하기에 비용과 시간이 많이 든다

2. [maske] token 자체가 fine-tuning 과정에서 등장하지 않기에, pre-training과 fine-tuning 사이에 불일치가 발생한다.

ELECTRA모델은 이러한 문제를 해결하기 위해 Replaced Token Detection (RTD)라는 새로운 방법을 만들었는데, 이는 실제 입력 토큰 중 일부를 가짜 토큰으로 만들고, 각 토큰이 실제 입력에 있는 진짜 토큰인지 아닌지를 맞히는 이진 분류 문제로 진행을 한다. 이런 방법을 통해 15%가 아닌 전체 토큰에 대해 학습이 진행이 가능하며, 그렇기에 학습시간이 빠르면서 좋은 성능을 얻을 수 있다.

2. METHOD

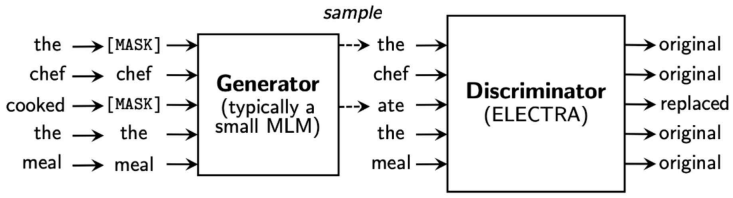

Figure 2: An overview of replaced token detection. The generator can be any model that produces an output distribution over tokens, but we usually use a small masked language model that is trained jointly with the discriminator. Although the models are structured like in a GAN, we train the generator with maximum likelihood rather than adversarially due to the difficulty of applying GANs to text. After pre-training, we throw out the generator and only fine-tune the discriminator (the ELECTRA model) on downstream tasks.

전체적인 구조는 하나의 Generator, 그리고 Discriminator를 가지고 있으며, 이 두 네트워크는 공통적으로 Transformer의 인코더 구조로 되어있다.

Generator :

Generator G는 사실 BERT의 MLM과 같다고 볼 수 있다.

먼저 입력 x=[x1,x2,...,xn]을 받아 입력에 대해 마스킹을 진행할 위치를 결정한다.

결정한 위치에 있는 토큰들을 [mask]로 바꿔준다.

마스킹 된 입력 토큰에 대해 generator는 원래 토큰이 무엇인지 예측한다.

Discriminator :

Discriminator D는 입력 토큰 시퀀스에 대해 generator가 예측한 토큰이 원래 있던 토큰인지, 아니면 가짜로 치환된 토큰인지를 이진 분류로 학습한다.

1. Generator G를 이용해서 마스킹 된 입력 토큰에 대해 예측을 진행

2. Genertor G에서 예측된 토큰들로 마스킹 된 토큰들을 치환하고, Discriminator D의 입력으로 넣어준다.

3. Discriminator D에서 입력 토큰 시퀀스에 대해 generator가 예측한 토큰이 원래 있던 토큰인지, 아니면 가짜로 치환된 토큰인지를 이진 분류로 학습한다.

GAN과의 차이점 :

Generator와 Discriminator가 있다는 것이 GAN과의 공통점이지만, ELECTRA의 training objective는 GAN과 몇가지 차이점이 있다.

1. Generator가 원래 토큰과 동일한 토큰을 생성했을 때, GAN은 negative sample(fake)로 간주하지만 ELECTRA는 positive sample로 간주하는 점.

2. Generator가 discriminator를 속이기 위해 adversarial 하게 학습하는 게 아니고 maximum likelihood로 학습한다는 점.

3. generator에서 샘플링하는 과정 때문에 역전파가 불가능하고, 따라서 adversarial 하게 generator를 학습하는게 어려움. 그래서 강화 학습으로 이를 구현해보았지만 maximum likelihood로 학습시키는 것보다 성능이 좋지 않았음(논문의 Appendix F 참조)

4. Generator의 입력으로 노이즈 벡터를 넣어주지 않는 점.

최종적으로 ELECTRA는 대용량 코퍼스에 대해서 generator loss와 discriminator loss의 합을 최소화하도록 학습한다. 앞에서 설명했듯이 샘플링 과정이 있기 때문에 discriminator loss는 generator로 역전파되지 않으며, 위의 구조로 pre-training을 마친 뒤에 generator는 버리고 discriminator만 취해서 downstream task으로 fine-tuning을 진행한다.