ELECTRA

전자공학과 220210031 유현우

1. 기존 대부분의 pretrained language model은 denoising autoencoder 학습 기반이라고 볼 수 있음
    A. 이는 주로 입력 sequence의 token 중 약 15%를 masking하고 이를 복원하는 masked language modeling(MLM)이라는 task를 통해서 학습
    B. 이는 기존 autoregressive language modeling과 비교하여 bi-directional한 information을 고려한다는 점에서 효과적이지만 문제가 있음
        i. 첫 번째는 전체 token중 15%에 대해서만 loss가 발생한다는 것
            1. 그래서 학습할 때 cost가 많이 들어감
        ii. 그리고 학습때는 masking된 token을 모델이 참고하여 prediction하지만 inference 시에는 mask된 token이 없음
2. 논문은 이런 문제를 개선하기위해 replaced token detection(RTD)를 제안
    A. 이는 새로운 pretraining task으로써 generator를 이용해 실제 입력의 일부 token을 replaced token으로 바꾸고 각 token이 실제 입력에 있는 original token인지 generator가 생성해낸 replaced token인지를 discriminator가 맞히는 binary classification task로 설정
    B. 이 pretraining task에서 ELECTRA는 입력의 15%가 아닌 모든 token에 대해 학습하므로 매우 efficient하며 effective 함
        i. ELECTRA가 BERT보다 훨씬 빠르게 학습할 수 있으며 downstreaming task에 이용할 때에도 더 좋은 성능을 보임
3. Method
    A. Model architecture

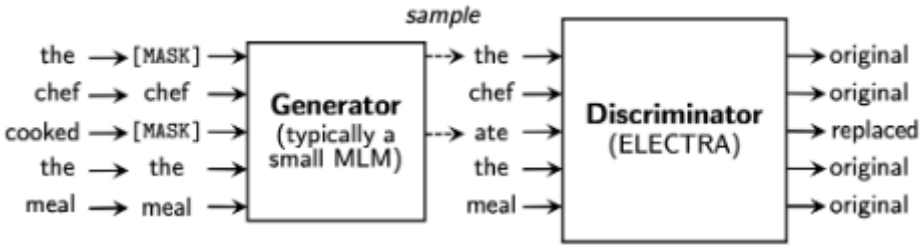

Figure 2: An overview of replaced token detection. The generator can be any model that produces an output distribution over tokens, but we usually use a small masked language model that is trained jointly with the discriminator. Although the models are structured like in a GAN, we train the generator 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.

i. 위의 model architecture에서 볼 수 있듯이 RTD task를 학습하기 위해서는 generator와 discriminator가 필요함

1. 이때 두 네트워크는 transformer encoder block으로 구성됨

ii. Generator

1. Generator는 BERT의 MLM mechanism과 동일

A. Masking 기반의 autoregressive method

2. 이때 학습되는 loss

A. $$\mathcal{L}_{\text{MLM}}(\mathbf{x}, \theta_G) = \mathbb{E}\left(\sum_{i \in \mathbf{m}} -\log p_G(x_i | \mathbf{x}^{masked})\right)$$

iii. Discriminator

1. Discriminator는 앞서 설명된 바와 같이 입력 token에 대해서 각 token이 original인지 replaced인지 binary classfication으로 분류하며 학습

2. $$\mathbf{x}^{corrupt} = \text{REPLACE}(\mathbf{x}, \mathbf{m}, \hat{\mathbf{x}})$$
$$\hat{\mathbf{x}} \sim p_G(x_i | \mathbf{x}^{masked}) \text{ for } i \in \mathbf{m}$$

A. Mechanism을 자세히 설명하자면, generator에서 masking할 위치의 집합 m에 masking된 token이 아닌 generator의 softmax분포 $p_G(x_t|\mathbf{x})$에 대해 sampling한 token으로 corrupt 해줌

3. $$D(\mathbf{x}^{corrupt}, t) = \text{sigmoid}(w^T h_D(\mathbf{x}^{corrupt})_t)$$

A. 치환된 입력 $\mathbf{x}^{corrupt}$에 대해서 discriminator는 softmax를 기반으로 각 token이 원래 input과 동일한 것인지 corrupt된 것인지를 예측

4. 이때 최종적인 loss는 아래와 같음

A. $$\mathcal{L}_{\text{Disc}}(\boldsymbol{x}, \theta_D) = \mathbb{E}\left(\sum_{t=1}^{n} -\mathbb{1}(x_t^{\text{corrupt}} = x_t)\log D(\boldsymbol{x}^{\text{corrupt}}, t) - \mathbb{1}(x_t^{\text{corrupt}} \neq x_t)\log(1 - D(\boldsymbol{x}^{\text{corrupt}}, t))\right)$$