[Week 7] 2022.08.18. **KUBIG 15기 수학과 이승은**

BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension [ACL 2020, Facebook AI]

BART [Bidirectional Auto-Regressive Transformer]

BART: BERT와 GPT를 하나로 합친 형태.

pretraining objective는 Denoising task로 두며 seq2seq architecture를 기본 (text를 corrupting하여 임의의 noise를 주어 훼손시키고, 다시 original text로 복구하는 식으로 학습을 진행)

cf. encoder - BERT [Bidirectional Encoder Representation from Transformer]: auto encoder같은 특성

cf. decoder - Auto-Regressive: GPT (generation)

-> BERT와 GPT를 합침 (Bidirectional한 Transformer의 encoder 그리고 Auto-Regressive한 Transformer의 decoder를 합친 seq2seq 모델을 학습시킨 모델)

-> 다양한 방식으로 text에 noising을 주어서 실험을 진행 (특히 text generation과 comprehension task에서 성능이 우수함)

1. Introduction

Self-supervised Learning (Word2vec, ELMO, BERT, SpanBERT, XLNET, Roberta)

autoencoder를 denoising하는 MLM (masked language model)의 성능이 좋음

BUT 특정 end task에만 집중해서 범용적으로 활용하는 데에는 어려움이 있음

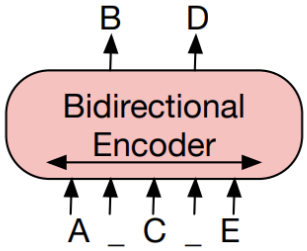
-> BART: Noising Flexibility (Arbitrary Transformations이 원본 text에 적용되어서 토큰 혹은 text의 길이 등을 자유롭게 변형할 수 있다, 즉 원본 text에 noise를 유연하게 적용할 수 있다)

2. Model

seq2seq는 RNN계열의 GRU, LSTM, Simple RNN 등을 Encoder와 Decoder의 모델로 활용

BART는 Transformer의 Encoder와 Decoder를 활용

cf. BERT



BERT는 pretraining objective로 NSP (Next Sentence Prediction)과 MLM (Masked Language Model)을 두어 학습 / Encoder로만 구성

BERT의 MLM:

$$\max_{\theta} \quad \log p_\theta(\bar{\mathbf{x}} \mid \hat{\mathbf{x}}) \approx \sum_{t=1}^{T} m_t \log p_\theta(x_t \mid \hat{\mathbf{x}})$$

$\bar{\mathbf{x}}$: original text

$\hat{\mathbf{x}}$: corrupted text

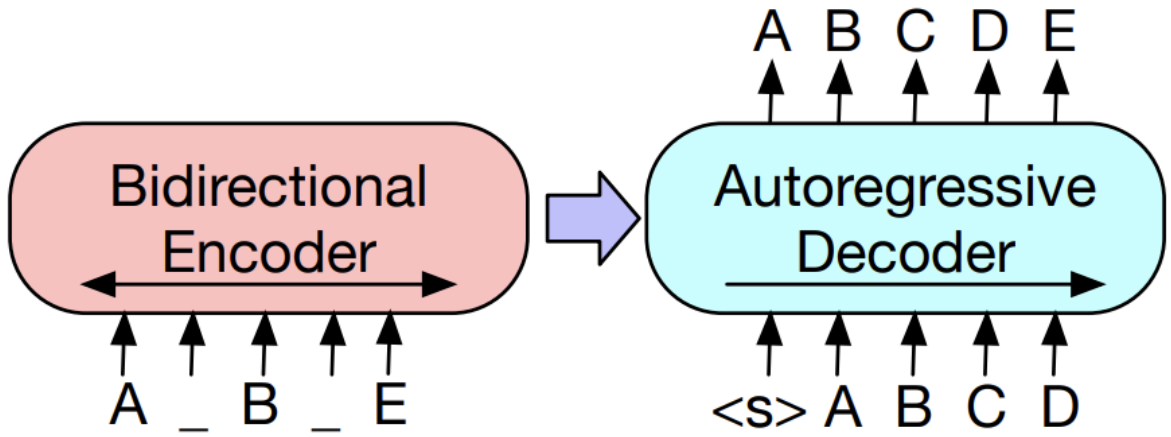
m_t : masked index

-> corrupted text가 original text로 복구되는 확률에 대한 MLE로 학습

(masked token을 복구할 때, 각각의 masked token들이 독립적으로 구축이 되어있어서 log 값의 summation으로 따로 계산가능)

* BERT는 pretrain의 경우 [MASK] token이 있지만, fine tuning시에는 [MASK] token이 없어서 두 가지 학습 방식 사이에 따른 discrepancy 문제

BART



activation function

GPT: ReLU / BART, BERT: GeLU

$$\mathrm{GELU}(x) = 0.5x\left(1+\tanh\left(\sqrt{2/\pi}(x+0.044715x^3)\right)\right)$$

(네트워크가 깊어질수록 adaptive dropout 형태로, 입력치에 가중치를 부여하여 zone out)

parameter들의 초기 값: 평균이 0이고, 표준편차가 0.02인 정규분포로 설정

1) Auto-Regressive (이전 모든 토큰들에 대해 다음 토큰을 예측하는 방식, masked token들이 이전 시점의 masked token에 영향을 받으므로 독립적으로 구축되어있는 문제 해결)

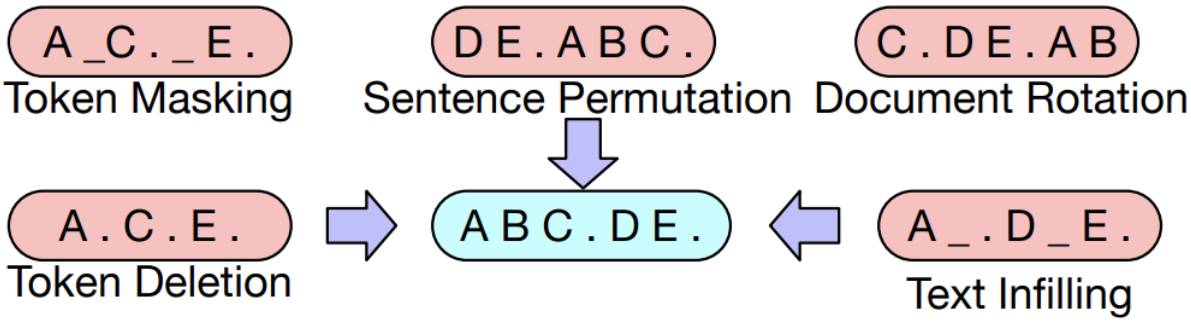
2) BERT와 달리 word prediction 직전에 FFNN 추가 X

3) BERT보다 BART가 10%정도의 parameter를 더 가지고 있음

"-> BART: Noising Flexibility (Arbitrary Transformations이 원본 text에 적용되어서 토큰 혹은 text의 길이 등을 자유롭게 변형할 수 있다, 즉 원본 text에 noise를 유연하게 적용할 수 있다)"

original text에 noising을 다양하게 줄 수 있음

Token Masking, Sentence Permutation, Document Rotation, Token Deletion, Text Infilling



1) Token Masking

BERT의 MLM과 동일한 방법. random tokens들이 추출되어 [MASK] token으로 대체되는 것.

2) Token Deletion

Input으로 들어가는 text에서 random tokens들이 삭제됨. [MASK] token을 맞추는 Token Masking과 다르게 어느 위치에서 token이 삭제가 되었는지 맞추는 것이 목적.

3) Text Infilling (가장 성능이 좋았음)

lambda가 3인 Poisson 분포에서 span length를 추출한 길이만큼의 text spans을 샘플링하고, 이를 단일 [MASK] token으로 대체

ABC.DE. -> A_. D_E.

첫번째 문장은 2만큼의 길이의 span에서 'BC'라는 text span이 샘플링이 되어 단일 [MASK] token으로 대체.

두번째 문장은 '0'만큼의 길이의 span에서 'empty'라는 text span이 샘플링이 되어 [MASK] token으로 대체

(span이란 단순히 text의 토큰들. span은 labmda가 3인 Poisson 분포 (평균=분산=lambda)를 따르기에 0-6 사이의 값이 span length로 선정될 확률이 큼)

Text Infilling 방법은 모델이 span에서 얼마만큼의 토큰들이 없어졌는지 예측하는 것을 학습

(SKT에서 발표한 KoBART는 'Text Infilling'만을 사용했다고 알려져 있음)

4) Sentence Permutation

단순히 문장 간의 순서를 바꿈. ABC.DE. -> DE.ABC.

5) Document Rotation

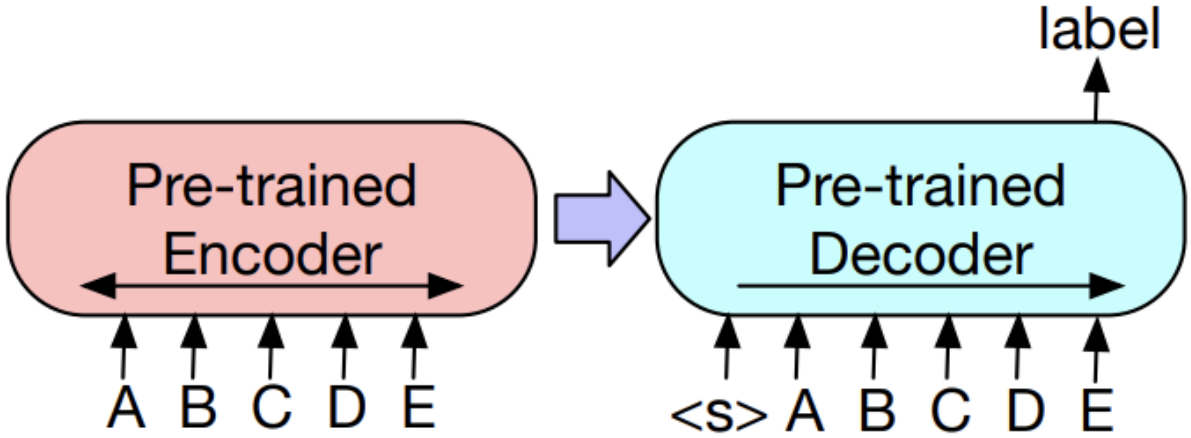
토큰들의 나열로 이루어진 text에서 동일한 확률로 랜덤하게 하나의 토큰을 골라서, 이를 시작점으로 두고 배열하는 것

ABC.DE.

랜덤하게 C라는 토큰을 임의로 뽑아 이를 시작 점에 배치해두고 C앞에 있던 토큰들은 뒤로 가서 바뀐 문장은 C.DE.AB

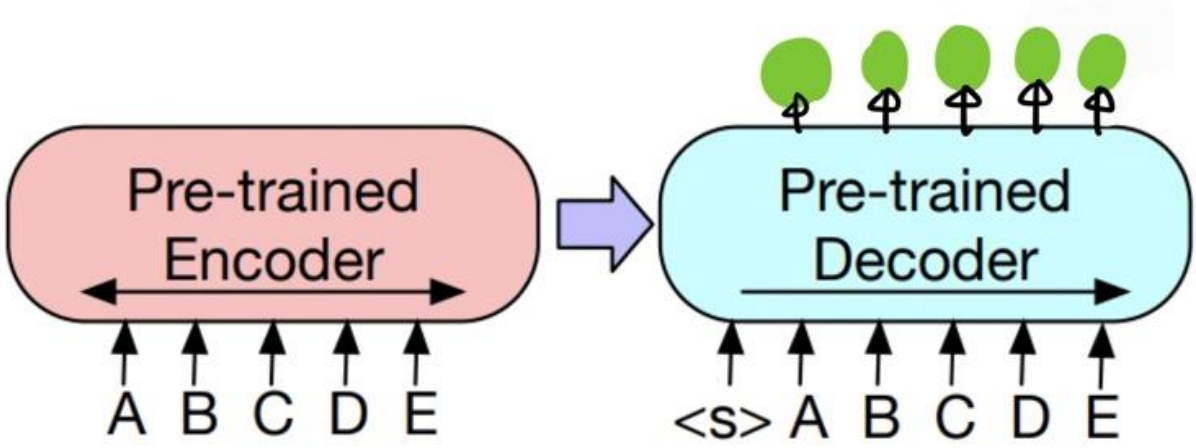
3. Fine-tuning

1) Sequence Classification Task



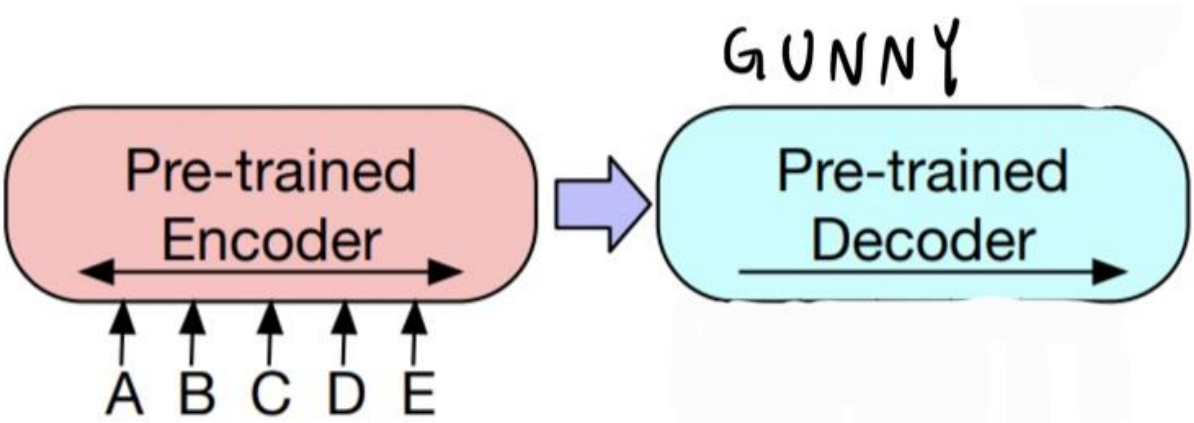
Decoder 토큰의 마지막 hidden state 값이 multi-class linear classifier로 들어가서 문장 분류

2) Token Classification Task



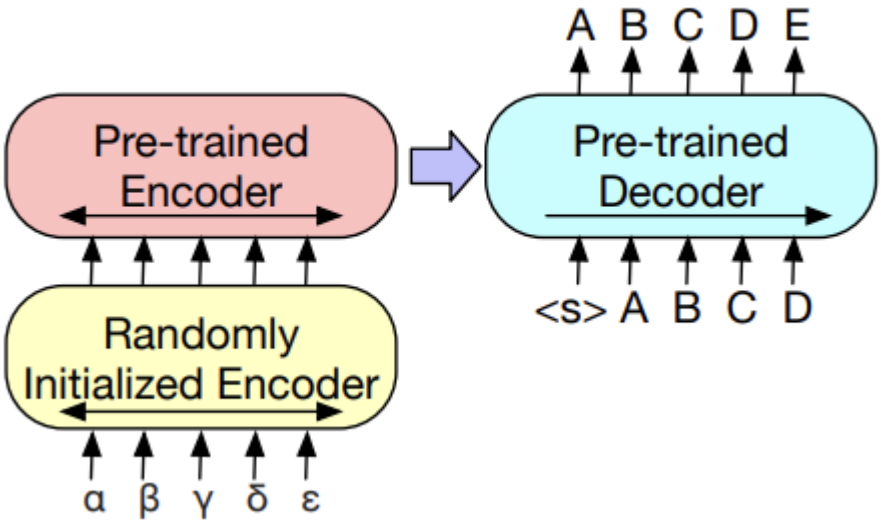
Decoder의 마지막 hidden state의 token 각각의 representation으로 분류

3) Sequence Generation Task



Auto-Regressive한 Decoder를 이용해서 생성

4) Machine Translation Task



대부분 (language1)과 (language2)간의 Translation은 noise를 주어 학습하는 경우에서는 학습이 안 된다고 함 -> additional encoder를 추가해서 다른 언어의 text 정보를 학습된 언어의 text 정보로 넘겨주어서 해결

BART라는 모델이 영어에만 학습된 모델이라고 할 때,

'가,나,다,라,마' 한글 text가 additional Encoder의 input으로 들어옴

이 한글에 대한 정보를 영어로 바꿔주어 기존의 영어로 학습된 pre-trained Encoder에 넣어주고

이는 pre-trained된 Decoder로 넘어가서 영어에 대한 output을 출력할 수 있습니다.

구체적으로, additional encoder에서 두 단계의 학습이 진행됨

- BART의 대부분의 parameters를 freeze하고 randomly initialized한 additional Encoder의 parameters들 학습.

- 적은 iteration 만큼의 횟수로 모델의 모든 parameter를 학습.

(Base) 6개의 Encoder와 Decoder, 그리고 hidden state를 768차원으로 설정한 모델

(Large) 12개 Encoder와 Decoder, 그리고 hidden state를 768차원으로 설정한 모델

4. Results

특히 요약문을 작성하는데 큰 성능 향상이 있었고

나머지는 RoBERTa와 비슷했음

Reference.

https://velog.io/@tobigs-nlp/BART-Denoising-Sequence-to-Sequence-Pre-training-for-Natural-Language-Generation-Translation-and-Comprehension

https://arxiv.org/abs/1910.13461

https://velog.io/@tajan_boy/Computer-Vision-GELU