BART - Denoising Sequence-to-Sequence Pre-training for Natural Language Genration, Translation, and Comprehension

문서 유형	Text Summarization
상태	진행 중
작성자	

[가짜연구소 : Agora NLP paper review group] 논문 발재로 공부한 'BART' 내용에 대해 정리 하고자 한다. 이번 포스팅에서는 'BART' 을 제안한 **BART : Denoising Sequence-to-Sequence Pre-training for Natural Language Genration, Translation, and Comprehension** 논문 내용을 바탕으로 참고 자료와 함께 다시 정리 하였다.

1. Introduction

- **BART는 Sequence-to-Sequence model 이면서, denoising task를 pre-train object 하는 model**

▼ **Bi-directional Encoder (BERT)와 Autoregressive Decoder (GPT)를 사용한 구조**

- Input text는 BERT의 Masked Language Model (MLM) 처럼 corrupted text를 input으로 사용
- Correpted input text를 original text로 reconstrut하는 task
- **하지만 기존 Masked Language Model (MLM) 방법들은 특정 End task 형태에 집중**

▼ **Difference BERT, BART**

- **BERT**
  - Transformer Encoder 부분으로만 구성한 model
  - MLM pre-train object와 NSP(Next Sentence Prediction) pre-train object 학습
  - 또한 BERT는 denoising Auto-Encoder base

    $logp_\theta(\bar{x}|\hat{x}) \approx \sum_{t=1}^{T} m_t \, logp_\theta(x_t \mid \hat{x})$

    ($\bar{x}$: reconstruct document, $\hat{x}$: corrupted document, m_t : masked token index)

    즉, Corrupted text가 input일 때 reconstruct text (Original text) 로 되돌릴 확률

    ▼ **XLNet에서는 MLM의 independent assumption problem 지적**

    - '모든 masked token은 독립적으로 재구축 되기 때문에 equal한 것이 아니라 approxi-mation' 하는 한계
    - fine-tuning에서는 masked token index를 사용하지 않기 때문에 pre-train / fine-tuning discrependency
- **BART**

  ▼ Bi-directional encoder + Autoregressive decoder 구조

  - Decoder의 각 layer별 cross-attention을 수행
  - Decoder에서 GeLUs activation function 사용
  - Parameter를 평균 0, 분산이 0.2인 정규 분포로 초기화
  - **Autoregressive Decoder를 가짐으로 BERT의 단점을 개선함**
  - **Pre-train task에 MLM을 사용하는데 이전 모델과 달리 Noise Flexibility를 가짐**

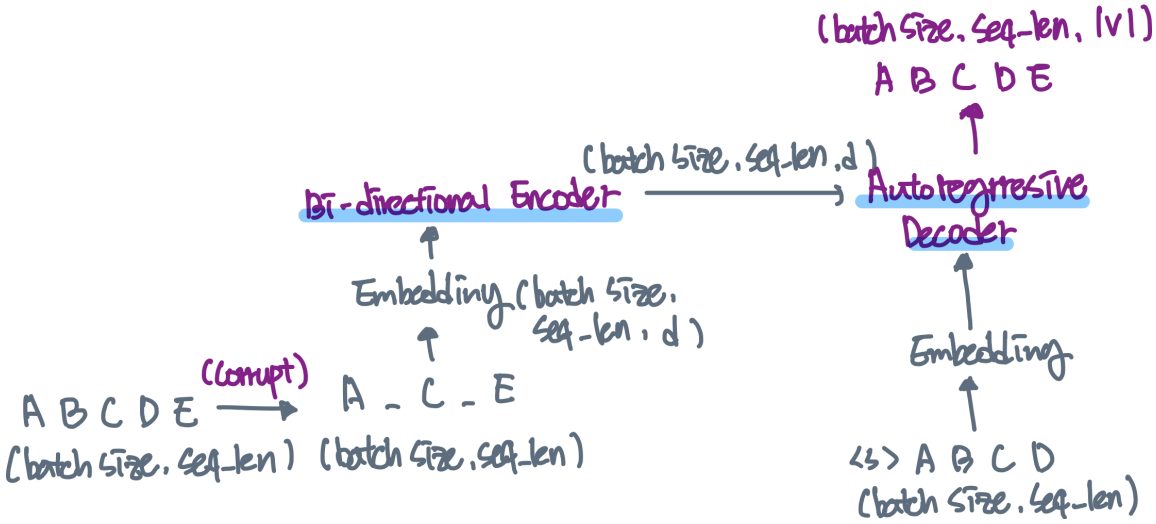
2. Model

▼ **BART Architecture**

- Corrupted text를 input으로 하고, 해당 **Text를 reconstruct한 decoder의 output과 original text의 cross-entropy reconstruct loss로 학습**

  $loss\ function(x, class) = -log(\frac{exp(x[class])}{\sum_j exp(x[j])}) = -x[class] + log(\sum_j exp(x[j]))$

(x : original text(batch size, seq_len) , class : deocder outputs(batch size, seq_len, |v|))



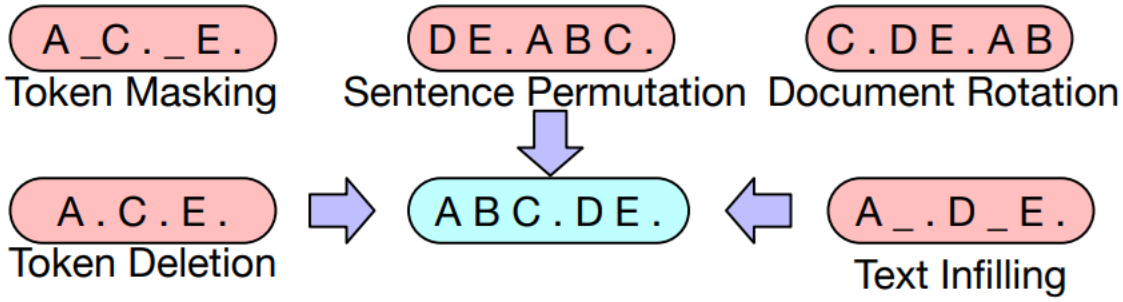
[그림 1] BART process architecture

▼ **Denoising object**

- **Token masking** : BERT와 동일하게 random token을 sampling 하고 이를 masking
- **Token deletion** : random하게 token을 제거
- **Text infilling** : poisson (λ = 3) 분포에서 span length를 추출한 길이만큼 text span을 sampling하고, 각 token을 단일 mask token으로 대체

  (SpanBERT에서 아이디어를 얻었는데 SpanBERT는 span의 길이를 알려주었으나 해당 논문에선 알려주지 않고 모델이 얼마나 많은 토큰이 빠졌는지 예측하게 함)
- **Sentence Permutation** : full stop들을 기준으로 document를 sentence로 나누고, 문장들을 ran-dom하게 섞음
- D**ocument rotation** : random token을 선택하고 document를 해당 token으로 시작하도록 rotate

  → model이 sentent의 시작이 어디인지 학습할수 있도록 함



[그림 2] Type of denoising object

3. Fine-tuning BART

▼ **Sequence Classification Task**

- 같은 input이 encoder와 decoder에 입력으로 사용하며, 디코더의 final hidden state가 새로운 linear classifier로 전달
- BERT의 [CLS] token과 유사하지만 마지막 토큰까지 입력해주면서 전체 입력에 대한 디코더의 attention을 계산할 수 있게 구조화 함

▼ **Token classification tasks**

- document를 encoder와 decoder에 입력으로 사용하며, decoder의 top hidden state를 각 단어에 대한 representation으로 하여 token classification에 사용

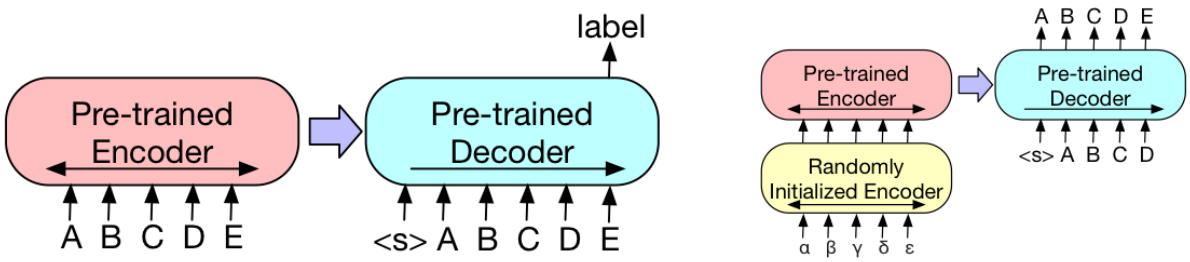
▼ **Sequence Generation Tasks**

- BART는 autoregressive 디코더를 갖고 있으므로 바로 fine-tuning 가능

▼ **Machine Translation Tasks**

- **BART를 machine translation 을 위한 pre-trained 디코더로 사용하고 새로운 인코더를 추가해 encoder-decoder를 fine-tuning**

  새로운 인코더는 외국어를 BART가 학습한 언어로 denoising 할 수 있는 입력으로 mapping 하는 역할



[그림 3] fine-tuning BART for classification and Translation

4. Comparing Pre-training Objectives

- BART는 noising method에 제약을 두지 않음으로 다양한 방법을 비교 실험 해봄
- 일반적인 sequence-to-sequence task처럼 source를 **인코더에 주고 target을 디코더 output으로** 하는 방법과 **source를 디코더 target의 prefix로 주고 target 부분만 loss를 계산하는 방법으로 학습**

  → 전자의 경우 BART가 다른 model 보다 좋은 performance를 보였으나 후자의 경우 그렇지 못함

▼ Comparison Objectives

- Language Model : GPT와 비슷하며, left-to-right transformer model을 train (cross-attention이 빠진 BART 디코더)
- Permuted Language Model : XL-Net을 기반으로, 1/6 토큰을 샘플링하고 랜덤한 순서로 auto-regressive하게 생성
- Masked Language Model : BERT처럼 15% 토큰을 mask 토큰으로 바꾸고 독립적으로 토큰을 예측
- Multitask Masked Language Model : self-attention mask를 추가해서 masked language mo-del을 학습
- Masked Seq-to-Seq : 토큰의 50%를 포함하는 span에 mask를 하고 mask된 토큰을 예측하는 seq-to-seq 모델을 학습

Model	SQuAD 1.1 F1	MNLI Acc	ELI5 PPL	XSum PPL	ConvAI2 PPL	CNN/DM PPL
BERT Base (Devlin et al., 2019)	88.5	**84.3**	-	-	-	-
Masked Language Model	90.0	83.5	24.77	7.87	12.59	7.06
Masked Seq2seq	87.0	82.1	23.40	6.80	11.43	6.19
Language Model	76.7	80.1	**21.40**	7.00	11.51	6.56
Permuted Language Model	89.1	83.7	24.03	7.69	12.23	6.96
Multitask Masked Language Model	89.2	82.4	23.73	7.50	12.39	6.74
BART Base						
w/ Token Masking	90.4	84.1	25.05	7.08	11.73	6.10
w/ Token Deletion	90.4	84.1	24.61	6.90	11.46	5.87
w/ Text Infilling	**90.8**	84.0	24.26	**6.61**	**11.05**	5.83
w/ Document Rotation	77.2	75.3	53.69	17.14	19.87	10.59
w/ Sentence Shuffling	85.4	81.5	41.87	10.93	16.67	7.89
w/ Text Infilling + Sentence Shuffling	**90.8**	83.8	24.17	6.62	11.12	**5.41**

[그림 4] Comparison or pre-training objectives

- **Result summary**

- **Pre-trained 모델의 성능은 task가 큰 영향을 줌**

▼ Token masking이 중요함

- 실험결과 Document rotation이나 sentence shuffling 기법만 사용했을때 성능이 좋지 않음
- Token deletion이나 masking 방법을 사용한 것이 더 좋은 성능을 보였고 generation task에서는 deletion이 masking 보다 더 좋은 성능을 보임

▼ Left-to-right pre-training이 generation 성능을 향상에 영향

- Masked Language Model과 Permuted Language Model은 다른 모델에 비해 generation 성능이 떨어지는데, 그 이유를 논문에서는 left-to-right auto-regressive 모델링이 포함되지 않았기 때문이라고 주장

- **SQuAD에선 문맥의 정보(future context)가 중요하기 때문에 이 정보를 얻지 못하는 left-to-right 디코더는 성능이 좋지 않음**
- **Pre-training 방법만 중요한 것이 아님**

5. Large-scale Pre-training Experiments

- 최근 연구에서 큰 batch size와 corpora를 사용해 pre-training이 성능의 향상 이끌어내고 있음으로 비교 실험 진행

	SQuAD 1.1 EM/F1	SQuAD 2.0 EM/F1	MNLI m/mm	SST Acc	QQP Acc	QNLI Acc	STS-B Acc	RTE Acc	MRPC Acc	CoLA Mcc
BERT	84.1/90.9	79.0/81.8	86.6/-	93.2	91.3	92.3	90.0	70.4	88.0	60.6
UniLM	-/-	80.5/83.4	87.0/85.9	94.5	-	92.7	-	70.9	-	61.1
XLNet	**89.0**/94.5	86.1/88.8	89.8/-	95.6	91.8	93.9	91.8	83.8	89.2	63.6
RoBERTa	88.9/**94.6**	**86.5/89.4**	**90.2/90.2**	96.4	92.2	94.7	**92.4**	86.6	**90.9**	**68.0**
BART	88.8/**94.6**	86.1/89.2	89.9/90.1	**96.6**	**92.5**	**94.9**	91.2	**87.0**	90.4	62.8

	CNN/DailyMail			XSum		
	R1	R2	RL	R1	R2	RL
Lead-3	40.42	17.62	36.67	16.30	1.60	11.95
PTGEN (See et al., 2017)	36.44	15.66	33.42	29.70	9.21	23.24
PTGEN+COV (See et al., 2017)	39.53	17.28	36.38	28.10	8.02	21.72
UniLM	43.33	20.21	40.51	-	-	-
BERTSUMABS (Liu & Lapata, 2019)	41.72	19.39	38.76	38.76	16.33	31.15
BERTSUMEXTABS (Liu & Lapata, 2019)	42.13	19.60	39.18	38.81	16.50	31.27
BART	**44.16**	**21.28**	**40.90**	**45.14**	**22.27**	**37.25**

[그림 5] Comparison Benchmark Dataset result