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

2021.7.21

채형주

목차

0. Abstract

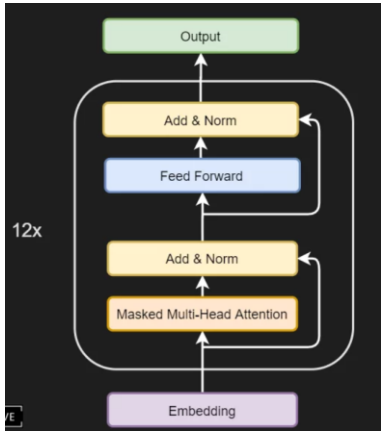
- BART는 seq2seq모델을 pretraining하기위한 denoising autoencoder이다
- 원래 문장을 변형하고 모델이 이를 다시 복원하는 과정에서 model의 성능이 올라감
- Transformer-based
- language generation에서 특히 좋은 성능을 보이고 comprehension task에서도 준수한 성능을 보여준다.
- abstractive dialogue, question answering, summarization task 에서 SOTA달성

1. Introduction / 최근 동향

- 최근에 다양한 NLP task에서 우수한 성과를 보여주는 것은 masked language model들이다.
- 무작위로 선택된 sub sequence들이 mask처리되고 이를 복원하도록 학습된 모델들
- 최근의 성과들은 masked token의 distribution을 향상시킴으로써 얻어졌는데 이는 한정된 end task에만 적용되는 것이었다.

1. Introduction /BART

BART : Bidirectional and Auto-Regressive Transformers



Auto Regressive vs Auto Encoder model

- Autoregressive model : 이전 token을 기반으로 다음 token을 predict

A statistical model is autoregressive if it predicts future values based on past values. For example, an autoregressive model might seek to predict a stock's future prices based on its past performance.

- *Autoencoding model :* 차원이 줄어든 encoding representation을 학습하는 모델

The aim of an autoencoder is to learn a representation (encoding) for a set of data, typically for dimensionality reduction, by training the network to ignore signal "noise". Along with the reduction side, a reconstructing side is learnt, where the autoencoder tries to generate from the reduced encoding a representation as close as possible to its original input, hence its name.

1. Introduction /BART

- 그러면 왜 BART는 Autoregressive model인가?

⇒Seq2Seq모델인지, autoregressive 모델인지, autoencoding모델인지는 모델 아키텍쳐로 정해지는 것이 아니다.

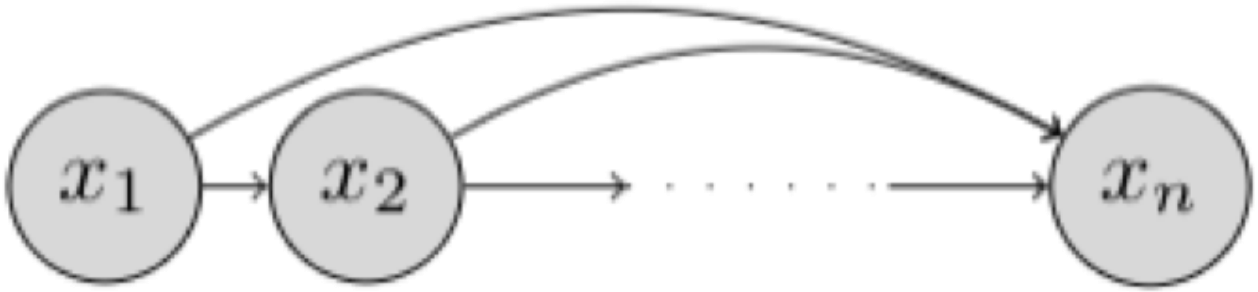
https://www.machinecurve.com/index.php/2020/12/29/differences-between-autoregressive-autoencoding-and-sequence-to-sequence-models-in-machine-learning/

어떤 task를 수행하는지, 어떤 방식으로 학습하는지에 따라서 결정됨.

Transformer의 decoder부분같은 경우에는 autoregressive task에 사용되지만, autoencoding에도 사용될 수는 있음. encoder도 마찬가지

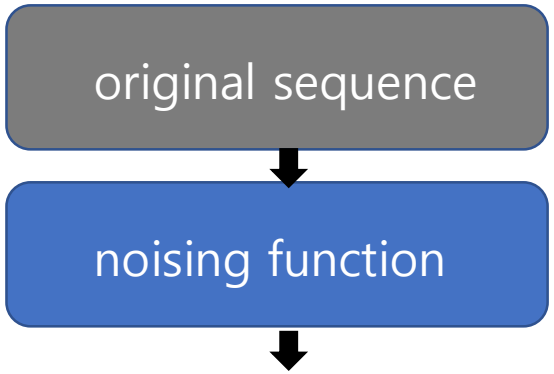
1. Introduction /BART

- model이 하나의 전체 sequenc를 다른 sequence를 변환하는 작업을 수행한다면 **Seq2Seq model**이다.
- model이 encoded representation을 생성하고 이를 기반으로 original sequence를 생성한다면 **autoencoding model**이다.
- model이 이전 token을 기반으로 다음 token 을 생성한다면 **autoregressive model**이다.



1. Introduction /BART - pretraining

- 1. original text를 임의의 noising function을 이용하여 변형한 뒤
- 2. sequence-to-sequence model이 original text를 생성한다.



[BART, models, a, autoencoder, denoising, for, pretraining, sequence-to-sequence]

[BART, is, a, denoising, autoencoder, for, pretraining, sequence-to-sequence models]

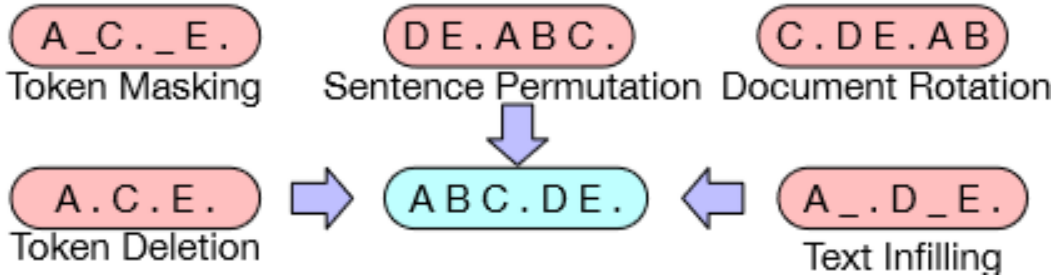
2. Model / 1.Architecture

- sequence-to-sequence model + bidirectional decoder + left-to-right autoregressive decoder
- standard sequence-to-sequence model Transformer architecture사용
- GPT에따라 ReLu대신에 GeLu사용함(initial parameter N(0, 0.02))
- Base model은 6 encoder/ 6 decoder, Large model은 12 / 12
- BERT와 유사하지만 decoder layer에서 마지막 encoder의 hidden layer에 대해 cross-attention을 수행하고, word prediction 이전에 additional feed-forward network를 사용하지 않는다는 점이 다름

2. Model / 2. Pre-training BART

- reconstruction loss를 optimize하는 방향으로 train(cross-entropy between decoder output and original sequence)
- 기존의 denoising autoencoder와는 다르게 BART는 어떠한 noising 방법도 사용할 수가 있다. (source에 대한 정보가 전부 없어졌을 때는 language model과 동일함)

2. Model / 2. Pre-training BART



여러 noising 기법들

- Token Masking

BERT에 따라 random token을 [MASK]로 replace

- Token Deletion

random token을 삭제한다. 하지만 삭제한 위치는 알려주지 않음

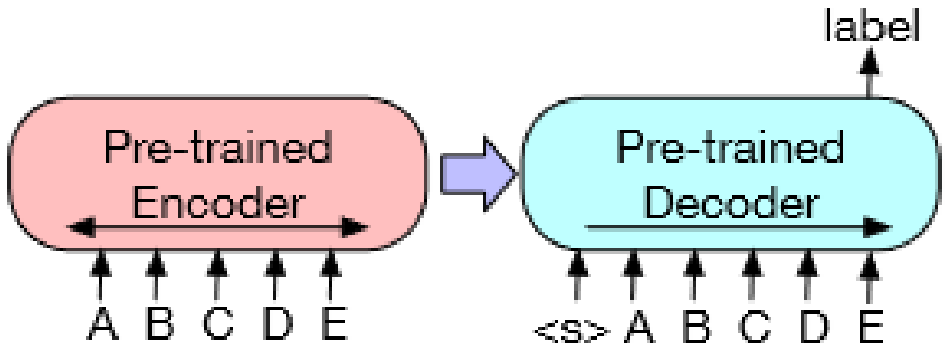
- Text Infilling

포아송 분포(λ = 3)에서 얻은 길이에 맞는 text span을 하나의 [MASK] token으로 replace. SpanBERT와 다른 점은 포아송 분포를 사용했다는 점, text span의 길이를 알려주지 않는다는 점이다.

- Sentence Permutation, Document Rotation

3. Fine-tuning BART /1. Sequence Classification

- decoder의 마지막 hidden state가 추가된 multi-class linear classifier로 들어가서 분류된다.



(a) To use BART for classification problems, the same input is fed into the encoder and decoder, and the representation from the final output is used.

3. Fine-tuning BART /2. Token Classification Tasks

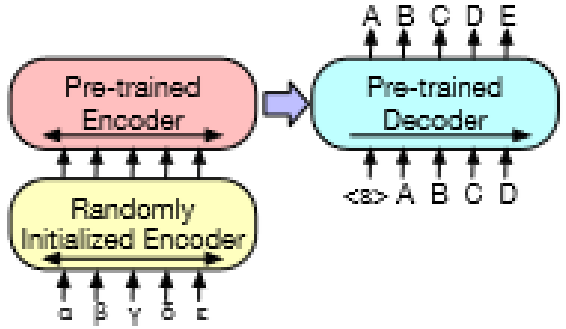
- NER, POS, Chunk …
- SQuAD dataset사용
- 자세한 내용은 Task 부분에서 설명

3. Fine-tuning BART /3. Sequence Generation Tasks

- question answering and summarization…
- decode에서 outpu을 autoregressive 하게 generate
- 자세한 내용은 Task 부분에서 설명

3. Fine-tuning BART /4. Machine Translation

- randomly initialize encoder layer를 추가함.
- 먼저 BART의 대부분의 paramete를 freez시키고, randomly initialize된 encoder만 update
- 그 다음 전체 parameter를 약간의 iteration동안 update



(b) For machine translation, we learn a small additional encoder that replaces the word embeddings in BART. The new encoder can use a disjoint vocabulary.

4. Comparing Pre-training Objectives /1. Comparison Objectives

- BERT의 변형된 모델로 테스트함. 1M steps, on a combination of books and Wikipedia data
- Language Model

GPT와 비슷함. cross-attention빼고는 BART의 decoder와 동일함.

- Permuted Language Model

XLNet 기반

다른 모델과의 일관성을 위해서 relative positional embedding, attention across segments는 구현하지 않음

- Masked Lanugage Model

BERT를 따라 15%를 MASK

- Multitask Masked Language Model

UniLM처럼 additional self-attention mask를 이용하여 학습.

1/6 right-to-left, 1/6 left-to-right, 1/3 unmasked, 1/3 처음 50%는 unmasked, 나머지에는 left-to-right

- Masked Seq-to-Seq

50%를 포함하는 span에 mask함.

4. Comparing Pre-training Objectives /2. Tasks

- SQuAD

BERT와 비슷하게 본문과 질문을 합쳐서 encoder의 input으로 넣어줌.

classifier가 각 토큰이 질문의 답에 대한 start/end 인지 분류

In meteorology, precipitation is any product of the condensation of atmospheric water vapor that falls under **gravity**. The main forms of precipitation include drizzle, rain, sleet, snow, **graupel** and hail... Precipitation forms as smaller droplets coalesce via collision with other rain drops or ice crystals **within a cloud**. Short, intense periods of rain in scattered locations are called "showers".

What causes precipitation to fall?
gravity

What is another main form of precipitation besides drizzle, rain, snow, sleet and hail?
graupel

Where do water droplets collide with ice crystals to form precipitation?
within a cloud

---

4. Comparing Pre-training Objectives /2. Tasks

- MNLI

sentence1 + <EOS> + sentence 2
<EOS> 토큰의 분류로 문장의 관계를 예측

- ELI5

abstractive QnA

- XSum

news summarization

- ConvAI2

dialogue response generation

- CNN/DM

news summarization

4. Comparing Pre-training Objectives /3. Results

Token masking 은 결정적인 역할을 한다. / rotating, permuting sucks

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**

4. Comparing Pre-training Objectives /3. Results

left-to-right pre-training 은 좋다/ Masked Language Model, Permuted – 는 적용안됨

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**

4. Comparing Pre-training Objectives /3. Results

pre-training object가 성능에 영향을 미치는 유일한 결정적인 요소는 아니다. Permuted LM같은 경우에는 XLNet보다 성능이 나빠짐. relative-position embedding, segment-level reccurrence같은 것들이 빠져서 그런듯

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**

5. Large-scale Pre-training Experiments

12 encoder/ 12 decoder

RoBERTa를 참고하여 batch size 8000, 50만 steps, BPE사용

4장에서 text infilling이 성능이 잘 나왔기 때문에 text infilling과 sentence permutatio을 함께 적용함. 30%의 토큰을 mask하고, 모든 문장은 permute함

Sentence Permutation이 그리 성능이 좋지 않았는데 왜 사용함?

⇒ larger pre-trained model에서는 이 task로부터 더 잘 배울것이다 라고 생각했기 때문. data에 더 잘 fit 하기 위해서 마지막 10%의 step에서는 dropout을 삭제

5. Large-scale Pre-training Experiments/ 2.Discriminative Tasks

SQuAD, GLUE dataset 사용

RoBERTa랑 거의 비슷함.

	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

Table 2: Results for large models on SQuAD and GLUE tasks. BART performs comparably to RoBERTa and XLNet, suggesting that BART's uni-directional decoder layers do not reduce performance on discriminative tasks.

5. Large-scale Pre-training Experiments /3. Generation Tasks

summarization

	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**

Table 3: Results on two standard summarization datasets. BART outperforms previous work on summarization on two tasks and all metrics, with gains of roughly 6 points on the more abstractive dataset.

dialogue

	ConvAI2	
	Valid F1	Valid PPL
Seq2Seq + Attention	16.02	35.07
Best System	19.09	17.51
BART	**20.72**	**11.85**

Table 4: BART outperforms previous work on conversational response generation. Perplexities are renormalized based on official tokenizer for ConvAI2.

Abstractive QA

	ELI5		
	R1	R2	RL
Best Extractive	23.5	3.1	17.5
Language Model	27.8	4.7	23.1
Seq2Seq	28.3	5.1	22.8
Seq2Seq Multitask	28.9	5.4	23.1
BART	**30.6**	**6.2**	**24.3**

Table 5: BART achieves state-of-the-art results on the challenging ELI5 abstractive question answering dataset. Comparison models are from Fan et al. (2019).

5. Large-scale Pre-training Experiments /4. Translation

Baselin : Transformer

beam width : 5, length penalty a = 1

	RO-EN
Baseline	36.80
Fixed BART	36.29
Tuned BART	**37.96**

Table 6: The performance (BLEU) of baseline and BART on WMT'16 RO-EN augmented with back-translation data. BART improves over a strong back-translation (BT) baseline by using monolingual English pre-training.