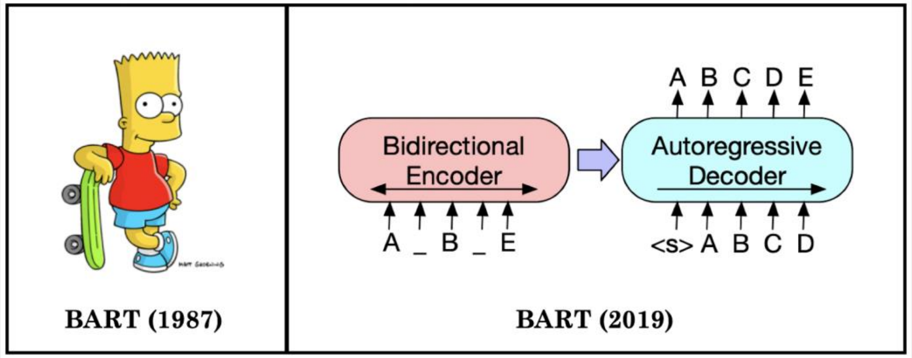
BART :
Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension



BART (1987)

BART (2019)

김사무엘

집현전 Paper-Review

1. Introduction

2. Model

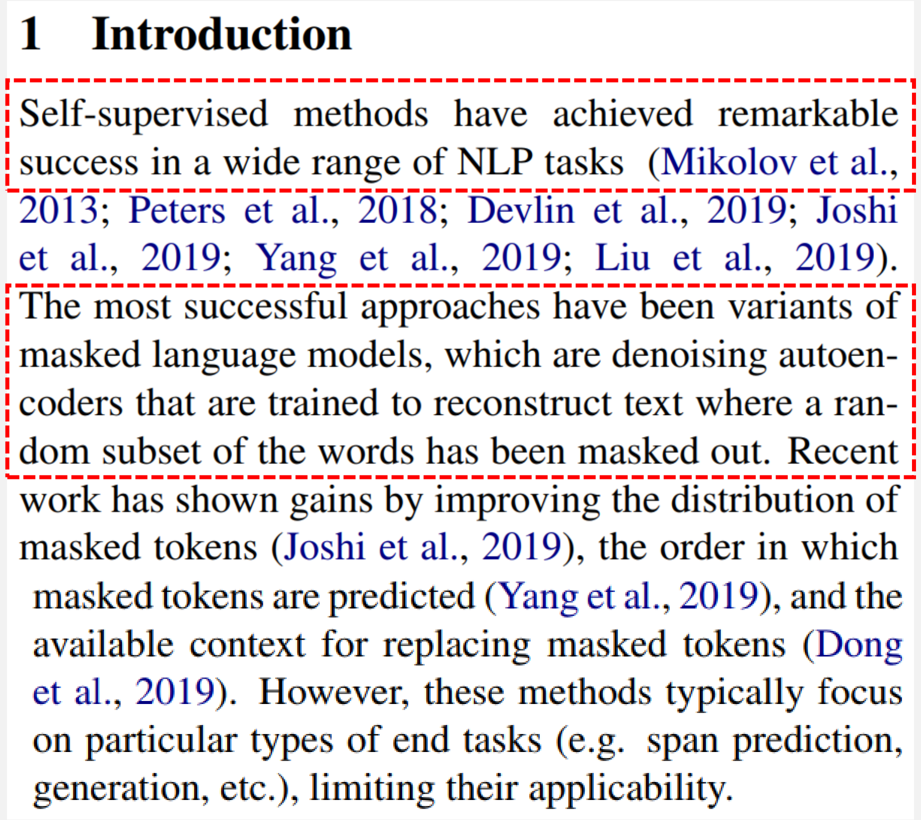
3. Fine-tuning BART

4. Comparing Pre-training Objectives

5. Large-scale Pre-training Experiments

6. Qualitative Analysis

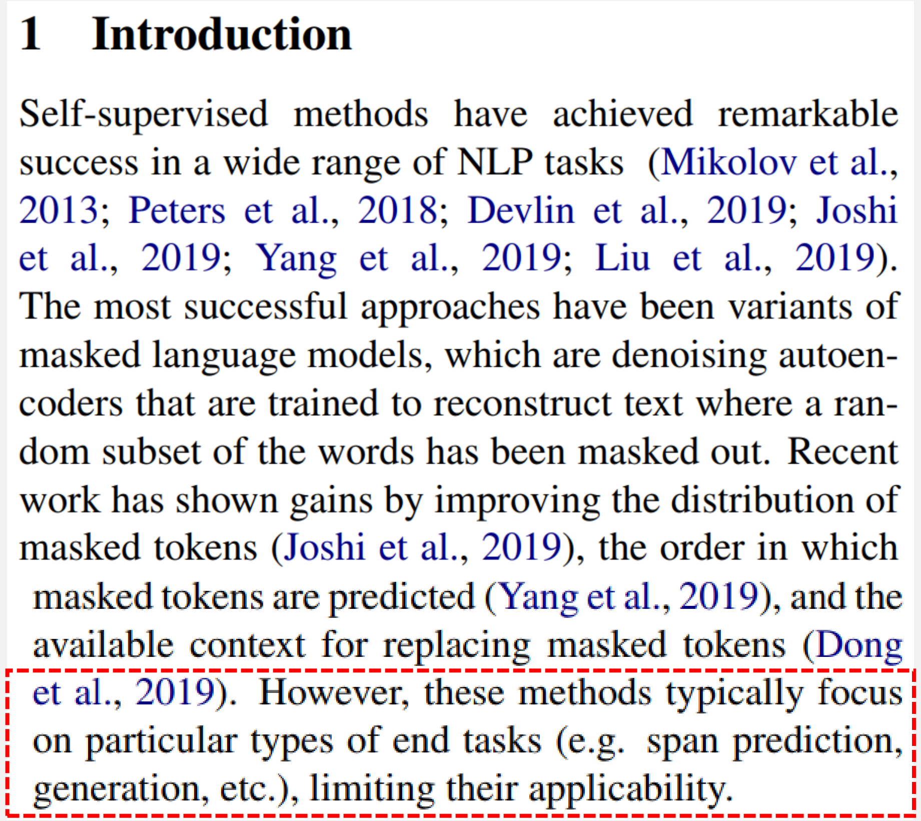
7. Related Work



1 Introduction

Self-supervised methods have achieved remarkable success in a wide range of NLP tasks (Mikolov et al., 2013; Peters et al., 2018; Devlin et al., 2019; Joshi et al., 2019; Yang et al., 2019; Liu et al., 2019). The most successful approaches have been variants of masked language models, which are denoising autoencoders that are trained to reconstruct text where a random subset of the words has been masked out. Recent work has shown gains by improving the distribution of masked tokens (Joshi et al., 2019), the order in which masked tokens are predicted (Yang et al., 2019), and the available context for replacing masked tokens (Dong et al., 2019). However, these methods typically focus on particular types of end tasks (e.g. span prediction, generation, etc.), limiting their applicability.

자기지도학습(self-supervised methods) 중 **MLM**(Masked Language Models) 방식이 최근 훌륭한 성과를 올림 (ex. **BERT**)



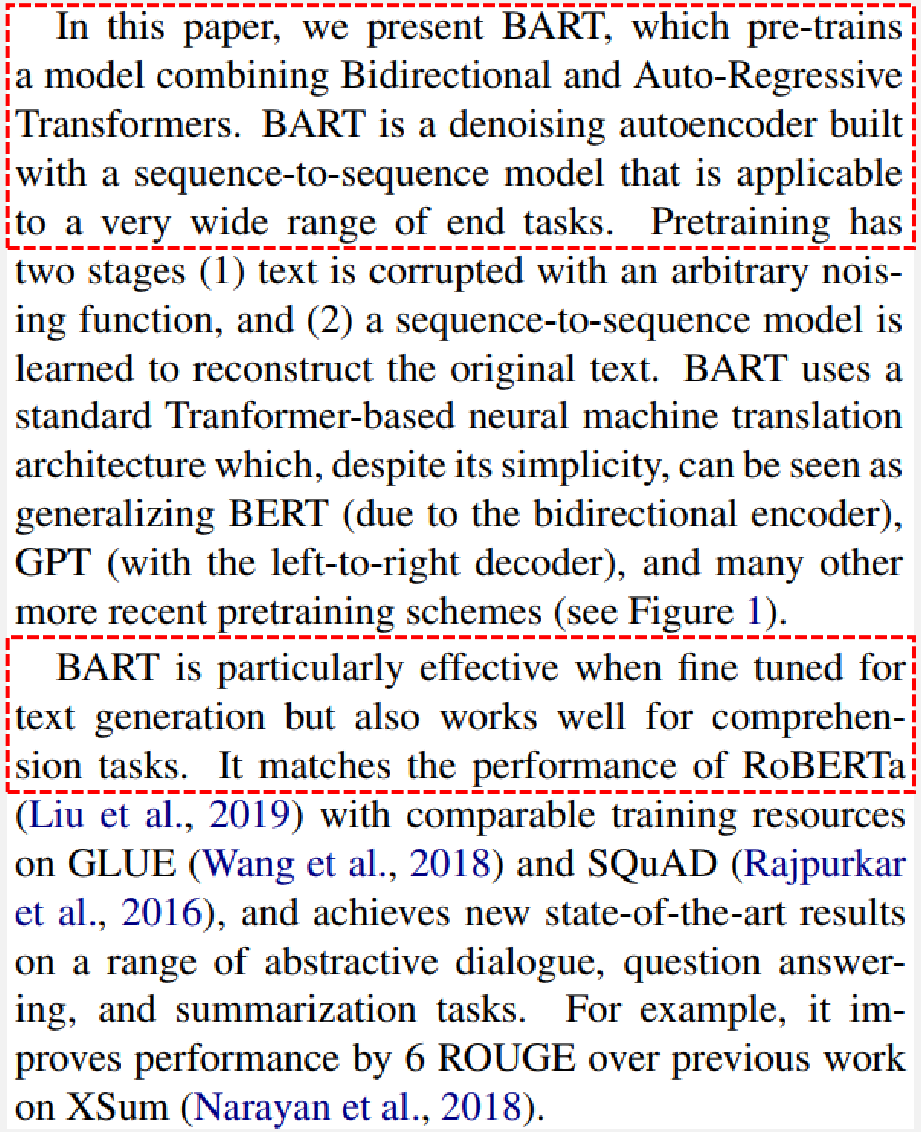
1 Introduction

Self-supervised methods have achieved remarkable success in a wide range of NLP tasks (Mikolov et al., 2013; Peters et al., 2018; Devlin et al., 2019; Joshi et al., 2019; Yang et al., 2019; Liu et al., 2019). The most successful approaches have been variants of masked language models, which are denoising autoencoders that are trained to reconstruct text where a random subset of the words has been masked out. Recent work has shown gains by improving the distribution of masked tokens (Joshi et al., 2019), the order in which masked tokens are predicted (Yang et al., 2019), and the available context for replacing masked tokens (Dong et al., 2019). However, these methods typically focus on particular types of end tasks (e.g. span prediction, generation, etc.), limiting their applicability.

그러나, 기존 방법들은 특정한 유형의 end task 에만 집중했고, 응용 가능성(applicability)이 떨어짐

Only for NLU

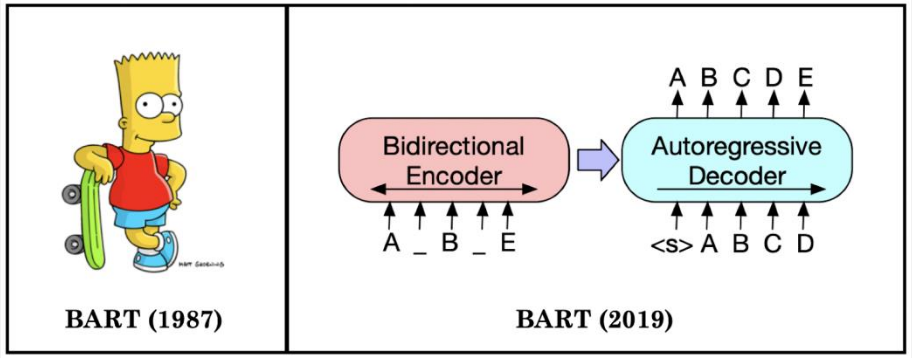
Only for NLG



In this paper, we present BART, which pre-trains a model combining Bidirectional and Auto-Regressive Transformers. BART is a denoising autoencoder built with a sequence-to-sequence model that is applicable to a very wide range of end tasks. Pretraining has two stages (1) text is corrupted with an arbitrary noising function, and (2) a sequence-to-sequence model is learned to reconstruct the original text. BART uses a standard Tranformer-based neural machine translation architecture which, despite its simplicity, can be seen as generalizing BERT (due to the bidirectional encoder), GPT (with the left-to-right decoder), and many other more recent pretraining schemes (see Figure 1).

BART is particularly effective when fine tuned for text generation but also works well for comprehension tasks. It matches the performance of RoBERTa (Liu et al., 2019) with comparable training resources on GLUE (Wang et al., 2018) and SQuAD (Rajpurkar et al., 2016), and achieves new state-of-the-art results on a range of abstractive dialogue, question answering, and summarization tasks. For example, it improves performance by 6 ROUGE over previous work on XSum (Narayan et al., 2018).

BART는 Bidirectional Transformer와 Auto-Regressive Transformer를 결합한 Seq-to-seq 구조의 모델로서, 굉장히 다양한 end task에 적용 가능함



BART는 텍스트 생성 문제에 파인 튜닝 되었을 때 특히 효과적이지만, NLU 문제에서도 훌륭하게 작동함

1. Introduction

2. Model

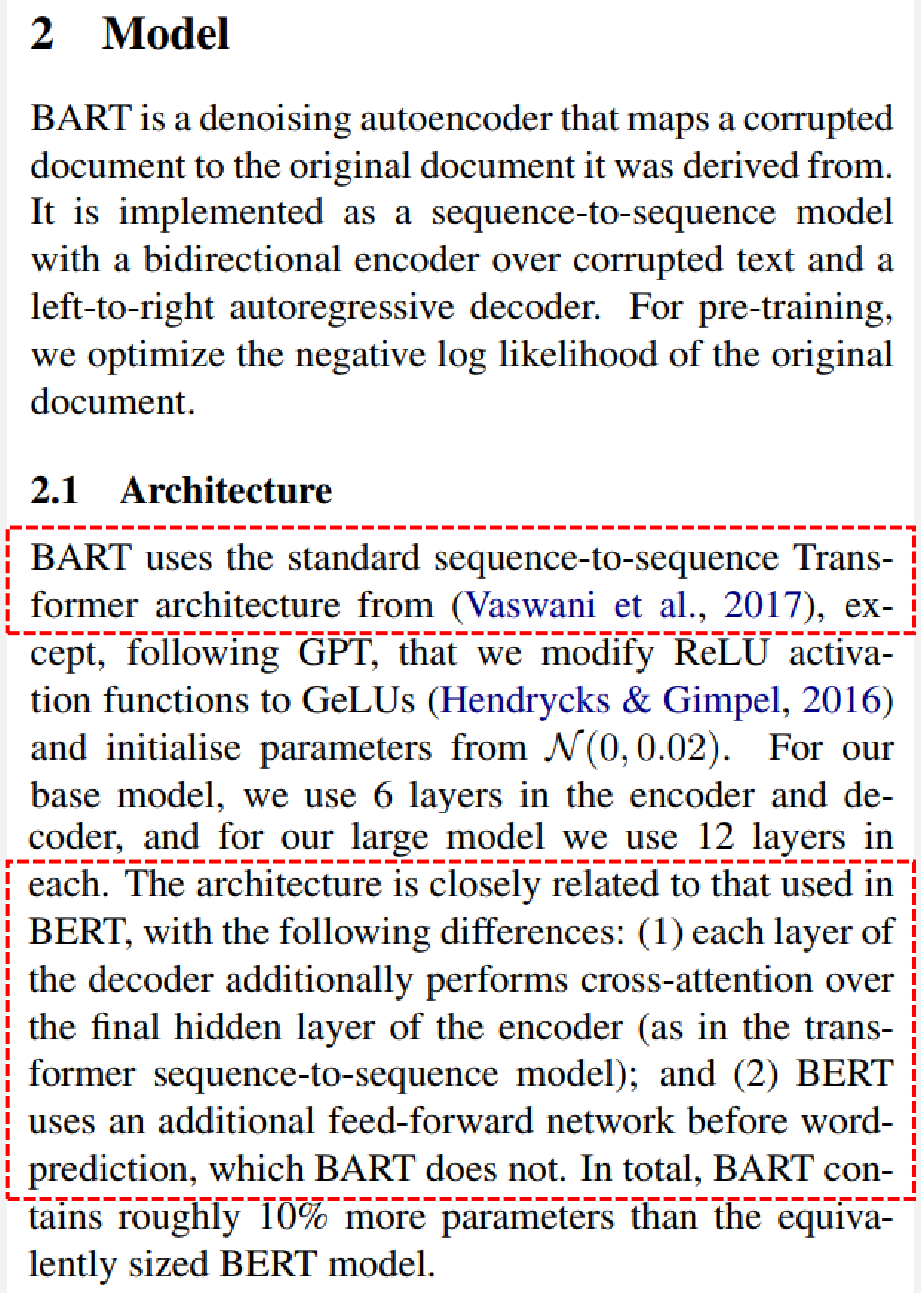
3. Fine-tuning BART

4. Comparing Pre-training Objectives

5. Large-scale Pre-training Experiments

6. Qualitative Analysis

7. Related Work

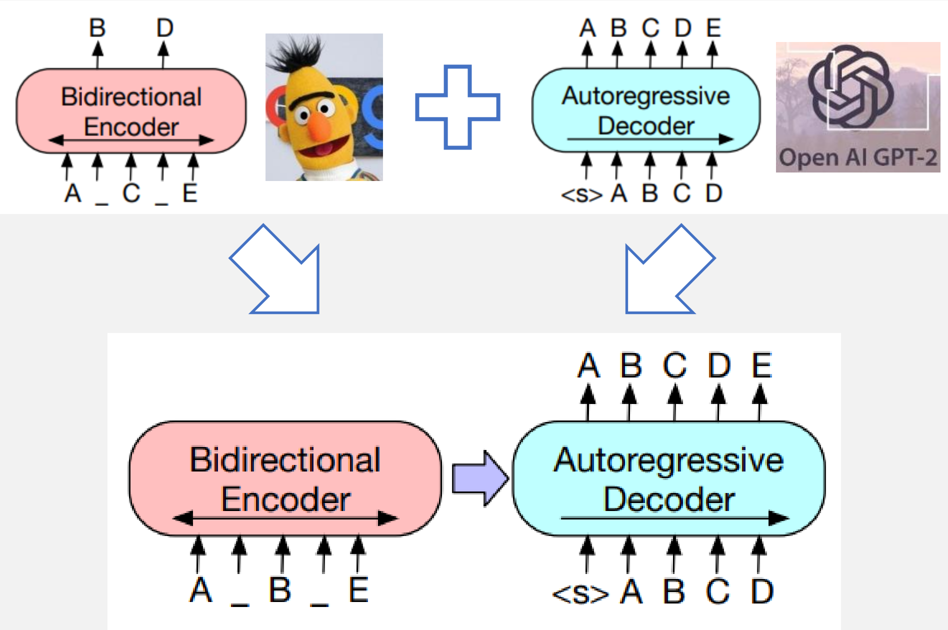


2 Model

BART is a denoising autoencoder that maps a corrupted document to the original document it was derived from. It is implemented as a sequence-to-sequence model with a bidirectional encoder over corrupted text and a left-to-right autoregressive decoder. For pre-training, we optimize the negative log likelihood of the original document.

2.1 Architecture

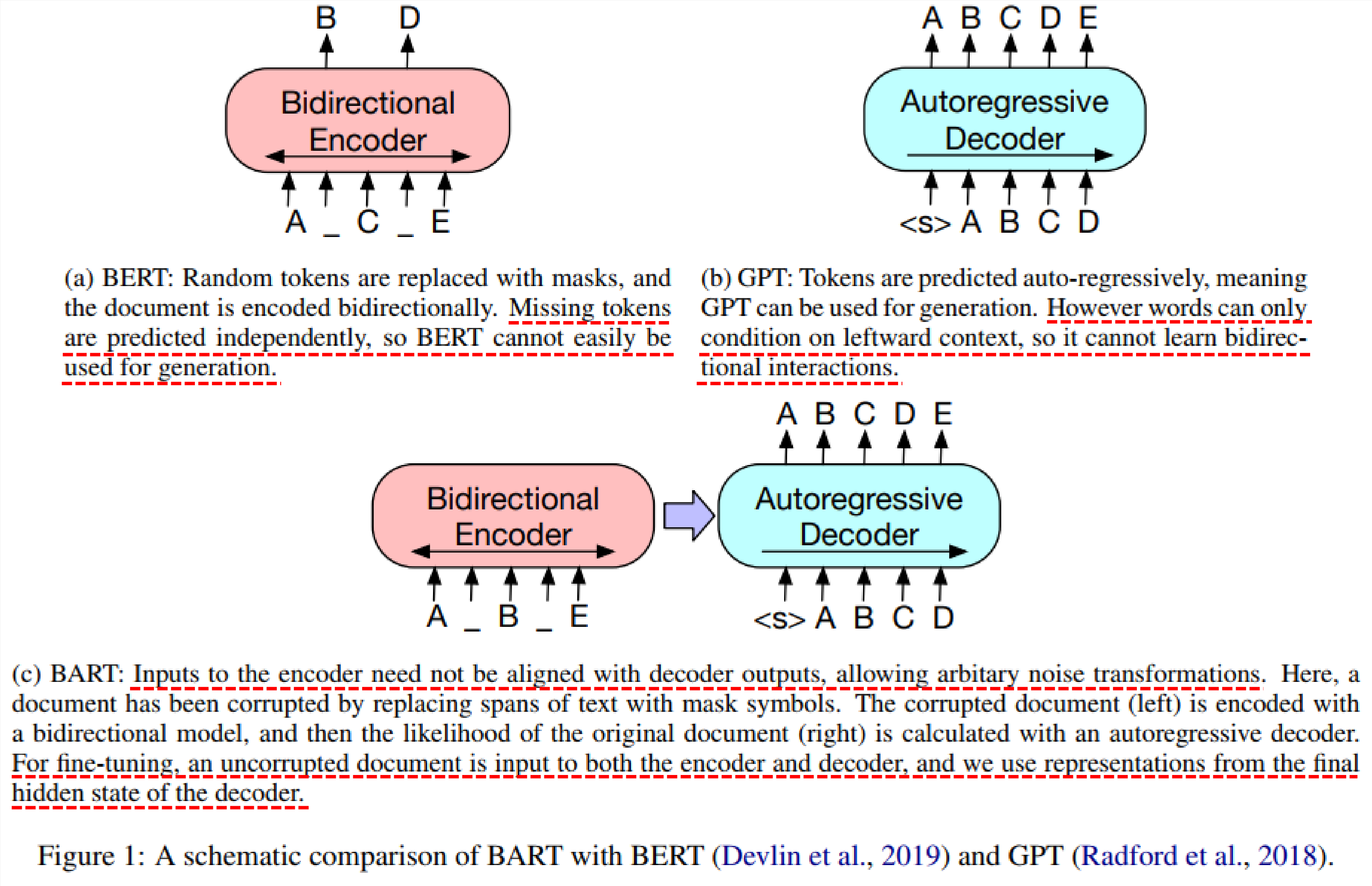
BART uses the standard sequence-to-sequence Transformer architecture from (Vaswani et al., 2017), except, following GPT, that we modify ReLU activation functions to GeLUs (Hendrycks & Gimpel, 2016) and initialise parameters from $\mathcal{N}(0, 0.02)$. For our base model, we use 6 layers in the encoder and decoder, and for our large model we use 12 layers in each. The architecture is closely related to that used in BERT, with the following differences: (1) each layer of the decoder additionally performs cross-attention over the final hidden layer of the encoder (as in the transformer sequence-to-sequence model); and (2) BERT uses an additional feed-forward network before word-prediction, which BART does not. In total, BART contains roughly 10% more parameters than the equivalently sized BERT model.



BART는 seq-to-seq transformer 아키텍처를 사용함
이 아키텍처는 BERT와 관계가 가깝지만, 두 가지 측면에서 BERT와 차이가 있음

(1) decoder의 각 레이어가 encoder의 마지막 hidden layer와 cross-attention 수행

(2) BERT는 word prediction을 위해 추가적인 FFN을 사용하지만, BART는 추가적인 FFN이 불필요함



(a) BERT: Random tokens are replaced with masks, and the document is encoded bidirectionally. Missing tokens are predicted independently, so BERT cannot easily be used for generation.

(b) GPT: Tokens are predicted auto-regressively, meaning GPT can be used for generation. However words can only condition on leftward context, so it cannot learn bidirectional interactions.

(c) BART: Inputs to the encoder need not be aligned with decoder outputs, allowing arbitary noise transformations. Here, a document has been corrupted by replacing spans of text with mask symbols. The corrupted document (left) is encoded with a bidirectional model, and then the likelihood of the original document (right) is calculated with an autoregressive decoder. For fine-tuning, an uncorrupted document is input to both the encoder and decoder, and we use representations from the final hidden state of the decoder.

Figure 1: A schematic comparison of BART with BERT (Devlin et al., 2019) and GPT (Radford et al., 2018).

1. BART는 bidirectional encoder(BERT)와 left-to-right decoder(GPT)가 결합된 구조임
2. BART에서는 encoder input과 decoder output의 순서가 동일할 필요가 없기 때문에, 임의의 nosing 적용 가능함
3. BART를 파인 튜닝할 때는 encoder와 decoder에서 모두 변형되지 않은 텍스트를 입력으로 전달함

BART

1. Introduction

2. Model

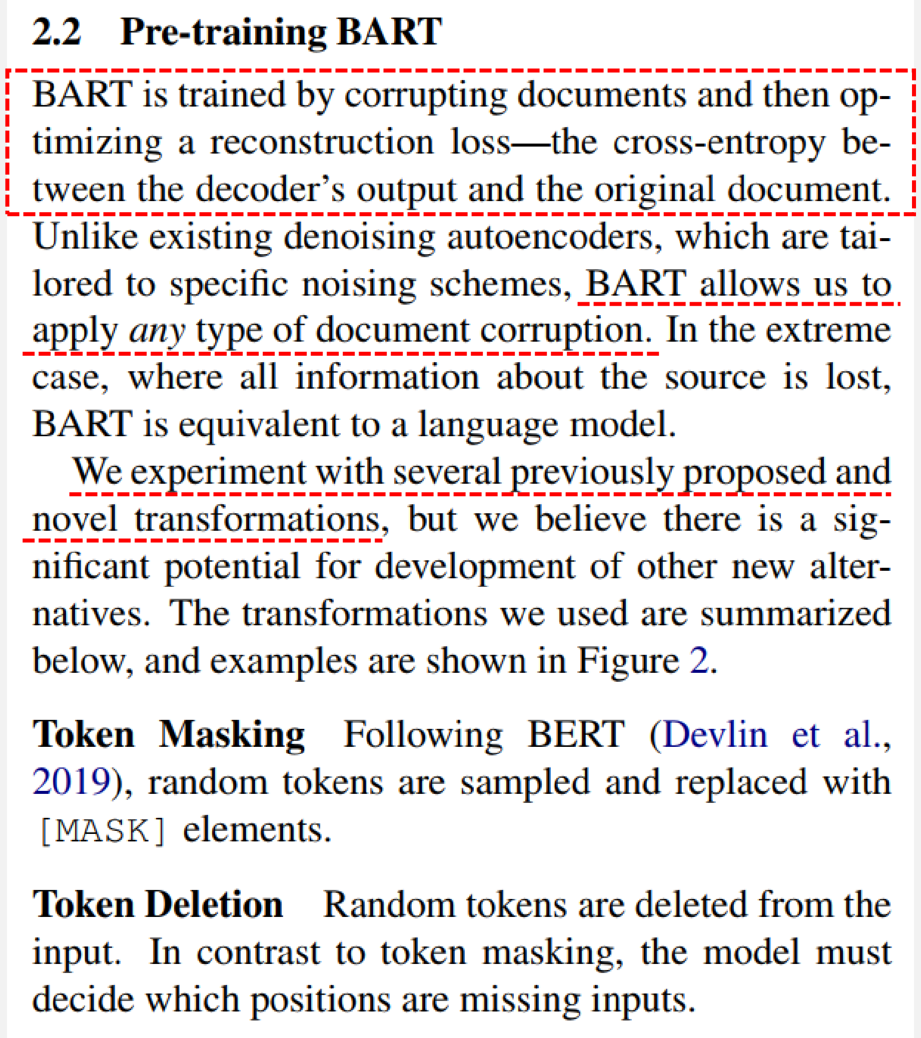
3. Fine-tuning BART

4. Comparing Pre-training Objectives

5. Large-scale Pre-training Experiments

6. Qualitative Analysis

7. Related Work



2.2 Pre-training BART

BART is trained by corrupting documents and then optimizing a reconstruction loss—the cross-entropy between the decoder's output and the original document. Unlike existing denoising autoencoders, which are tailored to specific noising schemes, BART allows us to apply *any* type of document corruption. In the extreme case, where all information about the source is lost, BART is equivalent to a language model.

We experiment with several previously proposed and novel transformations, but we believe there is a significant potential for development of other new alternatives. The transformations we used are summarized below, and examples are shown in Figure 2.

Token Masking Following BERT (Devlin et al., 2019), random tokens are sampled and replaced with `[MASK]` elements.

Token Deletion Random tokens are deleted from the input. In contrast to token masking, the model must decide which positions are missing inputs.

- BART는 텍스트에 noise를 추가하고 이를 되살리는 방식(denoising)으로 학습되며, 이 때 decoder 출력과 원 텍스트 간의 cross entropy를 loss로 사용함

- BART는 기존 denosing autoencoder들과 달리 어떤 방식의 document corruption도 학습에 적용할 수 있음. (본 논문에서는 5가지의 방식 실험)

BART

1. Introduction

2. Model

3. Fine-tuning BART

4. Comparing Pre-training Objectives

5. Large-scale Pre-training Experiments

6. Qualitative Analysis

7. Related Work

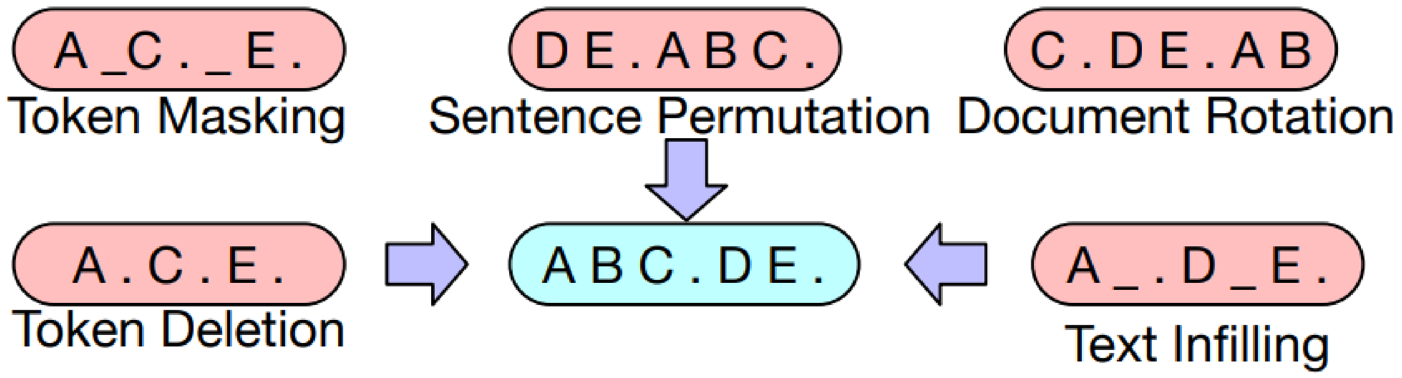


Figure 2: Transformations for noising the input that we experiment with. These transformations can be composed.

1. **Token Masking** : BERT처럼 임의의 토큰들이 [MASK] 토큰으로 대체됨
2. **Token Deletion** : 임의의 토큰들이 삭제됨. 모델은 어떤 위치에서 토큰이 삭제되었는지 결정해야 함
3. **Text Infilling** : 매번 일정한 길이(0~6)가 임의로 결정되고, 그 길이만큼의 토큰들이 하나의 [MASK] 토큰으로 대체됨. 모델은 얼마나 많은 수의 토큰들이 대체되었는지 예측해야 함
4. **Sentence Permutation** : 개별 문장의 순서를 임의로 뒤섞음
5. **Document Rotation** : 임의의 토큰이 선택되고, 이 토큰이 document의 시작이 되도록 문장들의 순서를 바꿈(the document is rotated). 모델은 document의 시작을 예측하도록 훈련됨

BART

1. Introduction

2. Model

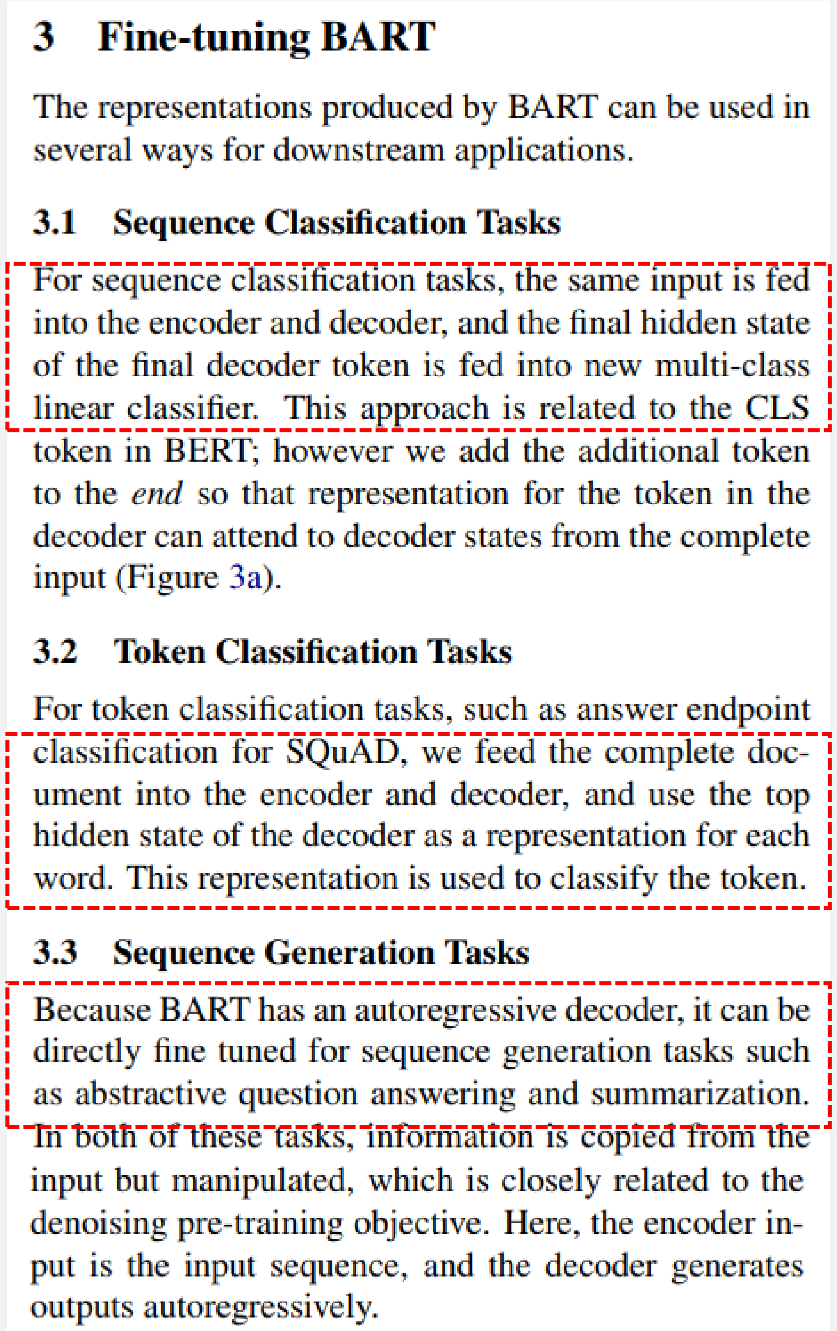
3. Fine-tuning BART

4. Comparing Pre-training Objectives

5. Large-scale Pre-training Experiments

6. Qualitative Analysis

7. Related Work



3 Fine-tuning BART

The representations produced by BART can be used in several ways for downstream applications.

3.1 Sequence Classification Tasks

For sequence classification tasks, the same input is fed into the encoder and decoder, and the final hidden state of the final decoder token is fed into new multi-class linear classifier. This approach is related to the CLS token in BERT; however we add the additional token to the *end* so that representation for the token in the decoder can attend to decoder states from the complete input (Figure 3a).

3.2 Token Classification Tasks

For token classification tasks, such as answer endpoint classification for SQuAD, we feed the complete document into the encoder and decoder, and use the top hidden state of the decoder as a representation for each word. This representation is used to classify the token.

3.3 Sequence Generation Tasks

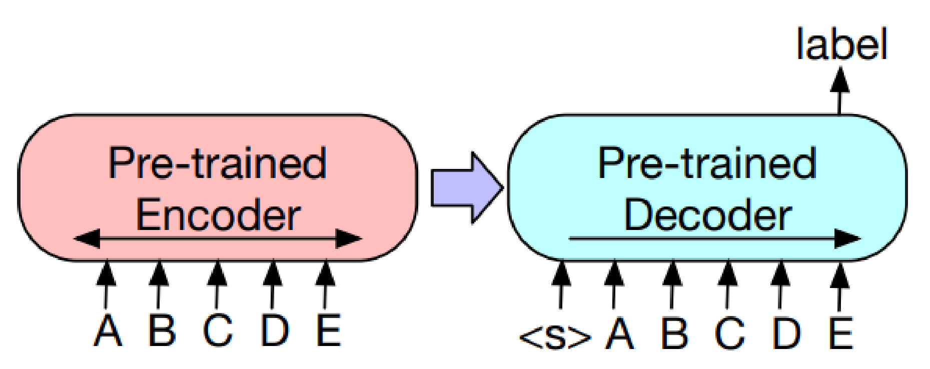
Because BART has an autoregressive decoder, it can be directly fine tuned for sequence generation tasks such as abstractive question answering and summarization. In both of these tasks, information is copied from the input but manipulated, which is closely related to the denoising pre-training objective. Here, the encoder input is the input sequence, and the decoder generates outputs autoregressively.

1) Sequence Classification Tasks

encoder와 decoder에 동일한 입력을 넣어주고, decoder의 마지막 hidden state를 새로운 multi-class linear classifier에 전달함

2) Token Classification Tasks

encoder와 decoder에 완전한 텍스트를 넣어주고, decoder의 top hidden state로 개별 토큰을 분류함



3) Sequence Generation Tasks

BART decoder가 그 자체로 Autoregressive하므로, 바로 파인 튜닝이 가능함

BART

1. Introduction

2. Model

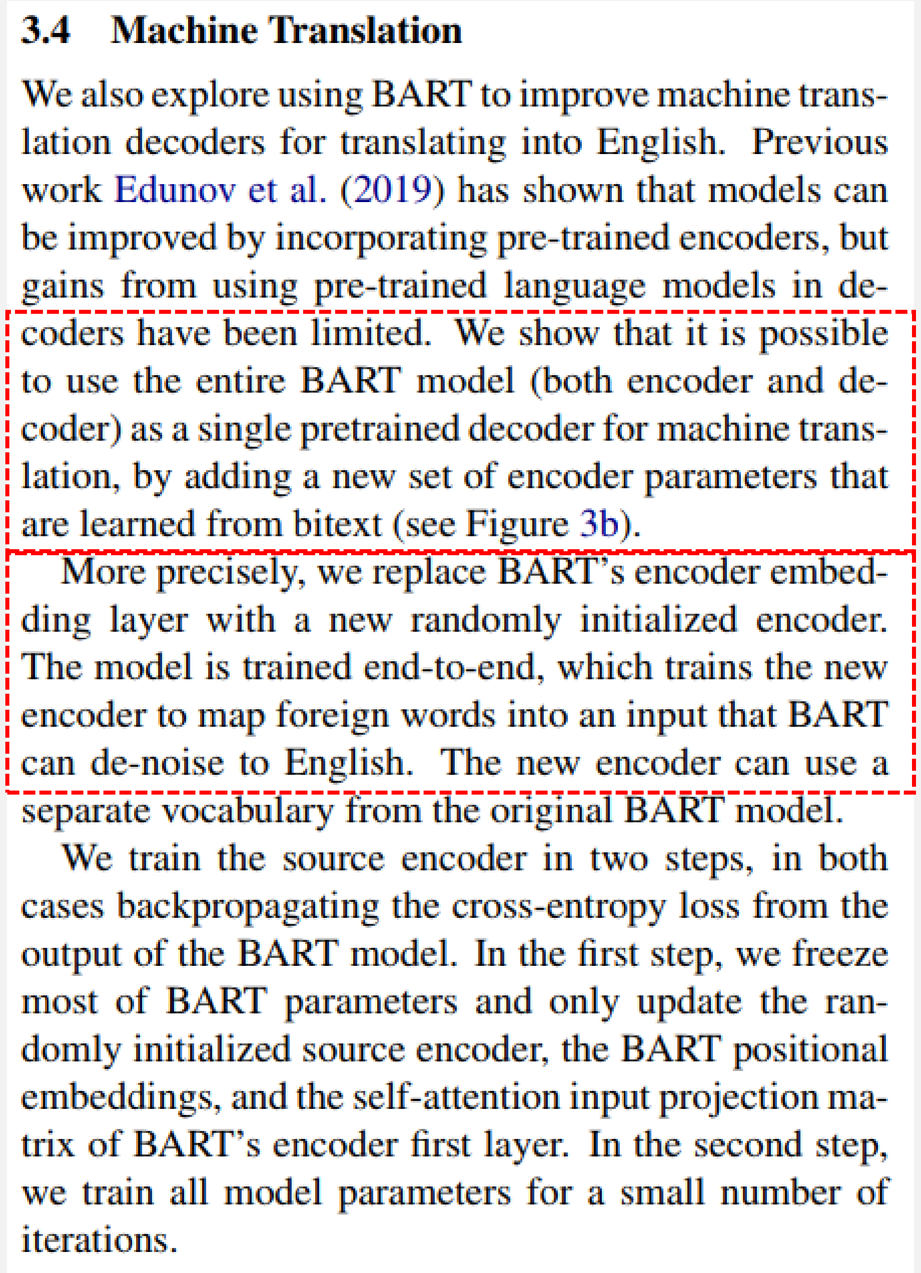
3. Fine-tuning BART

4. Comparing Pre-training Objectives

5. Large-scale Pre-training Experiments

6. Qualitative Analysis

7. Related Work



3.4 Machine Translation

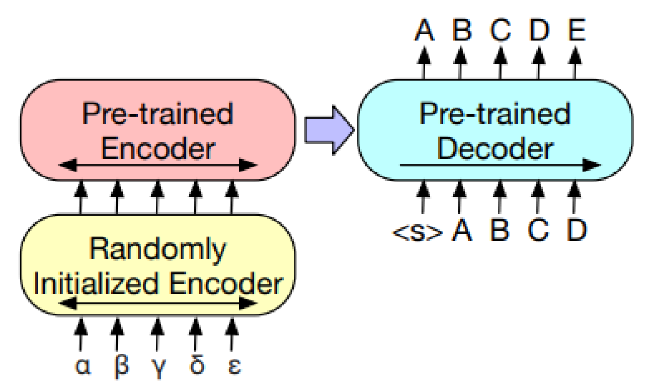
We also explore using BART to improve machine translation decoders for translating into English. Previous work Edunov et al. (2019) has shown that models can be improved by incorporating pre-trained encoders, but gains from using pre-trained language models in decoders have been limited. We show that it is possible to use the entire BART model (both encoder and decoder) as a single pretrained decoder for machine translation, by adding a new set of encoder parameters that are learned from bitext (see Figure 3b).

More precisely, we replace BART's encoder embedding layer with a new randomly initialized encoder. The model is trained end-to-end, which trains the new encoder to map foreign words into an input that BART can de-noise to English. The new encoder can use a separate vocabulary from the original BART model.

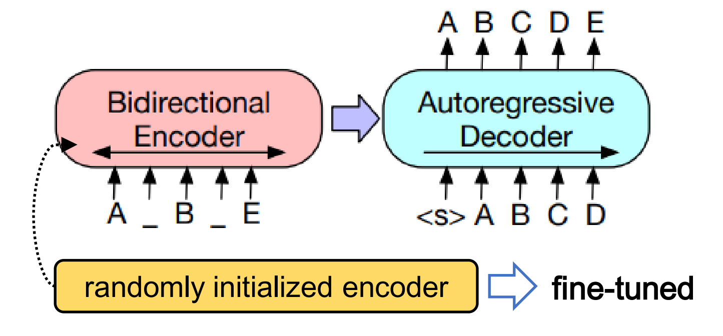
We train the source encoder in two steps, in both cases backpropagating the cross-entropy loss from the output of the BART model. In the first step, we freeze most of BART parameters and only update the randomly initialized source encoder, the BART positional embeddings, and the self-attention input projection matrix of BART's encoder first layer. In the second step, we train all model parameters for a small number of iterations.

4) Machine Translation

BART 모델을 기계번역기의 pretrained decoder로 사용하고, 새로운 encoder layer만 더해 파인 튜닝함



보다 구체적으로, BART의 **encoder embedding layer**를 새로운 encoder로 대체하고, 파인 튜닝함. 파인 튜닝으로 새롭게 학습된 encoder는 다른 언어의 토큰을 BART가 denoise할 영어 토큰으로 대체함



BART

1. Introduction

2. Model

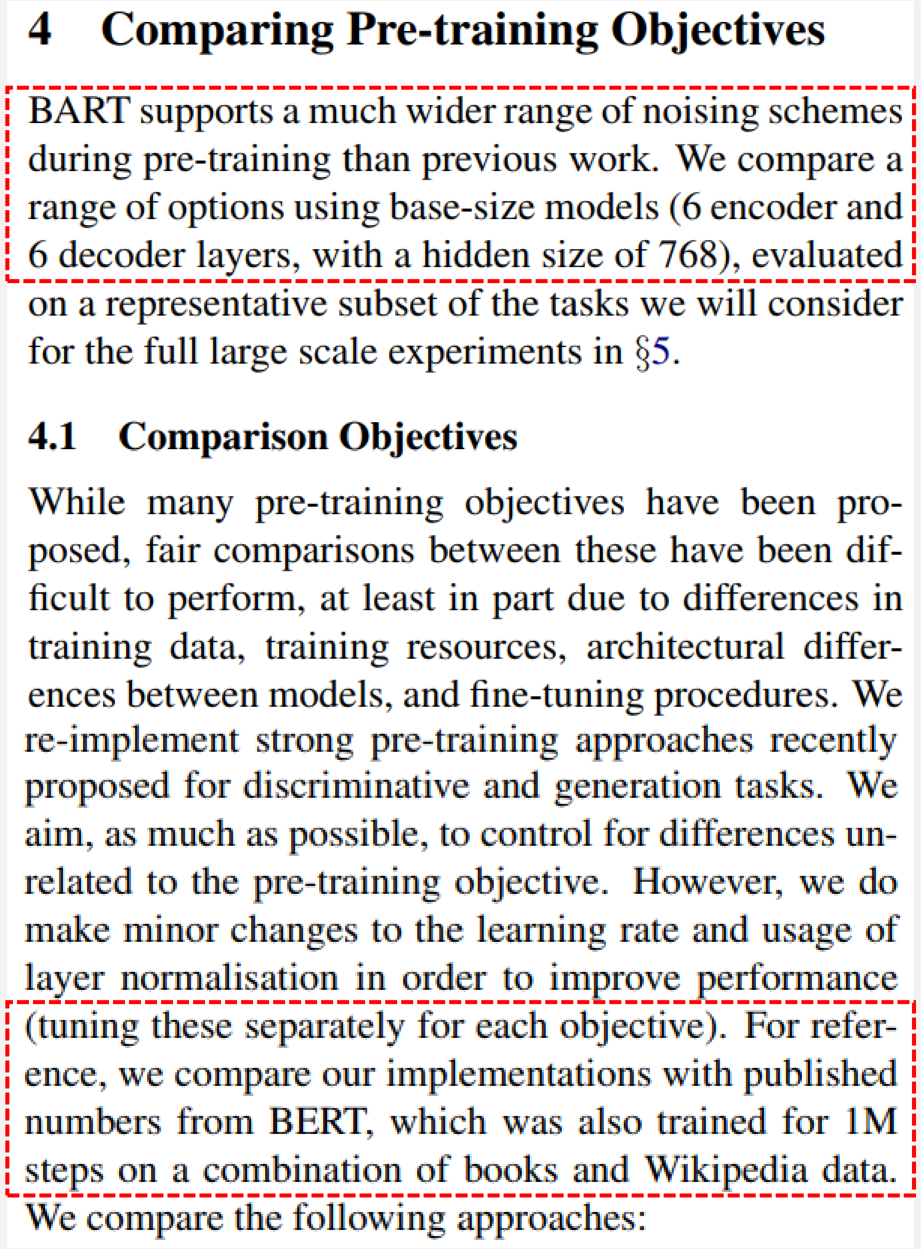
3. Fine-tuning BART

4. Comparing Pre-training Objectives

5. Large-scale Pre-training Experiments

6. Qualitative Analysis

7. Related Work



4 Comparing Pre-training Objectives

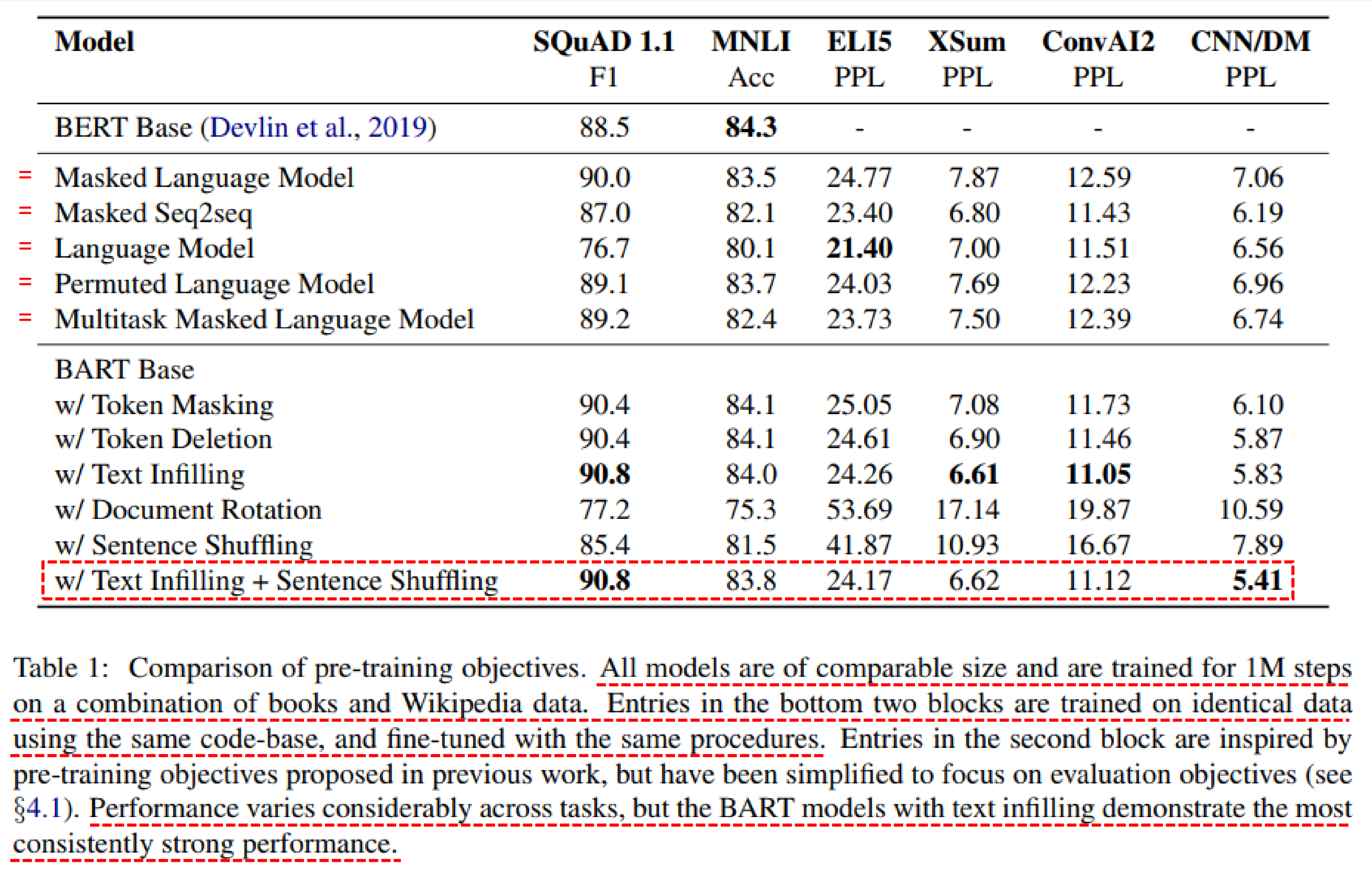
BART supports a much wider range of noising schemes during pre-training than previous work. We compare a range of options using base-size models (6 encoder and 6 decoder layers, with a hidden size of 768), evaluated on a representative subset of the tasks we will consider for the full large scale experiments in §5.

4.1 Comparison Objectives

While many pre-training objectives have been proposed, fair comparisons between these have been difficult to perform, at least in part due to differences in training data, training resources, architectural differences between models, and fine-tuning procedures. We re-implement strong pre-training approaches recently proposed for discriminative and generation tasks. We aim, as much as possible, to control for differences unrelated to the pre-training objective. However, we do make minor changes to the learning rate and usage of layer normalisation in order to improve performance (tuning these separately for each objective). For reference, we compare our implementations with published numbers from BERT, which was also trained for 1M steps on a combination of books and Wikipedia data. We compare the following approaches:

BART 기본 모델(6 encoder layer + 6 decoder layer)
사전학습 진행 후 다른 기존 모델들과 성능 비교함

비교를 위해 BERT paper의 성능 보고와 다른 다양한
모델들의 학습 결과를 함께 비교함
(books & Wikipedia data 활용)



		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**	-	-	-	-
BERT	=	Masked Language Model	90.0	83.5	24.77	7.87	12.59	7.06
MASS	=	Masked Seq2seq	87.0	82.1	23.40	6.80	11.43	6.19
GPT	=	Language Model	76.7	80.1	**21.40**	7.00	11.51	6.56
XLNet	=	Permuted Language Model	89.1	83.7	24.03	7.69	12.23	6.96
UniLM	=	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**

Table 1: Comparison of pre-training objectives. All models are of comparable size and are trained for 1M steps on a combination of books and Wikipedia data. Entries in the bottom two blocks are trained on identical data using the same code-base, and fine-tuned with the same procedures. Entries in the second block are inspired by pre-training objectives proposed in previous work, but have been simplified to focus on evaluation objectives (see §4.1). Performance varies considerably across tasks, but the BART models with text infilling demonstrate the most consistently strong performance.

동일한 조건에서 학습 후 성능을 비교했을 때,
BART (Text infilling + Sentence shuffling)가 가장 일정하고 훌륭한 성능을 보임

BART

1. Introduction

2. Model

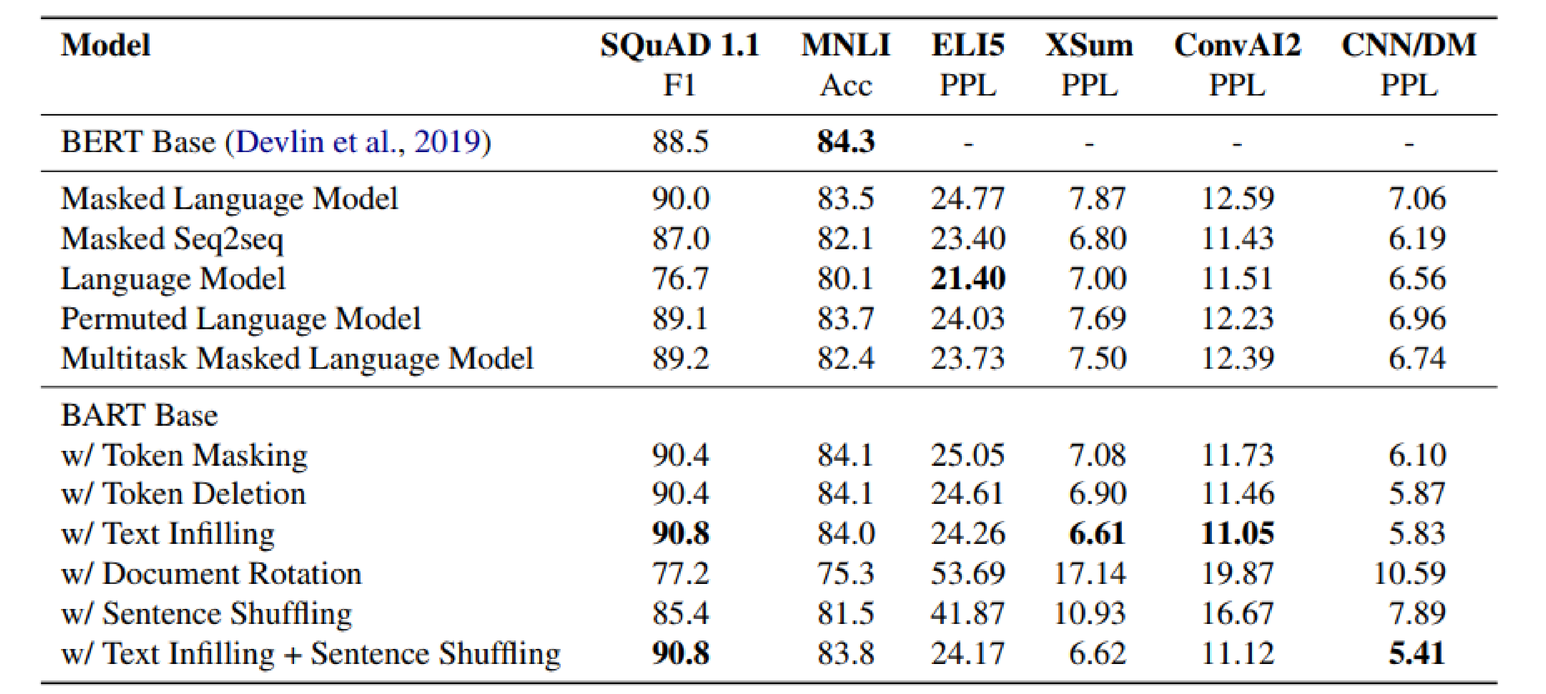
3. Fine-tuning BART

4. Comparing Pre-training Objectives

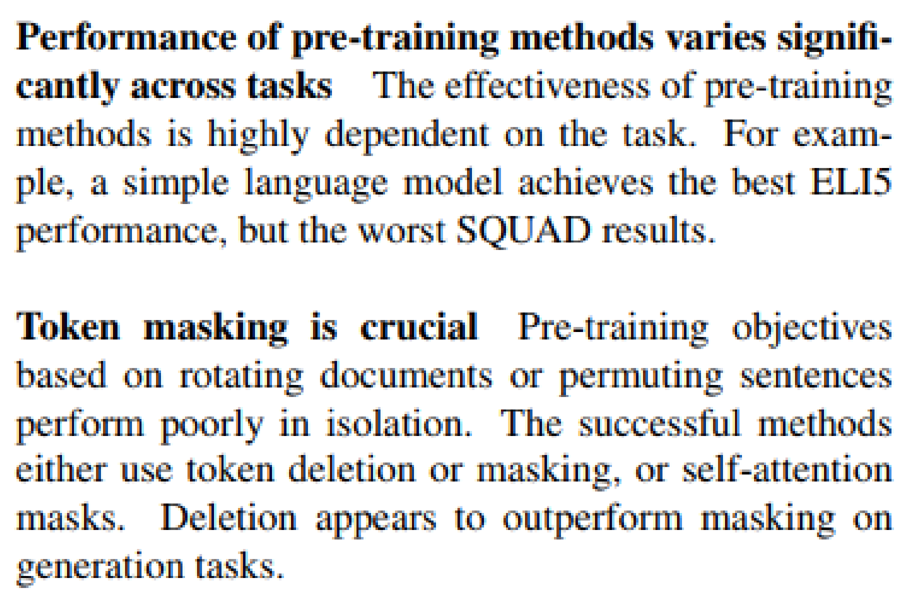
5. Large-scale Pre-training Experiments

6. Qualitative Analysis

7. Related Work



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**



Performance of pre-training methods varies significantly across tasks The effectiveness of pre-training methods is highly dependent on the task. For example, a simple language model achieves the best ELI5 performance, but the worst SQUAD results.

Token masking is crucial Pre-training objectives based on rotating documents or permuting sentences perform poorly in isolation. The successful methods either use token deletion or masking, or self-attention masks. Deletion appears to outperform masking on generation tasks.

1. 각 사전 학습 방식의 성능은 task에 따라 편차가 큼

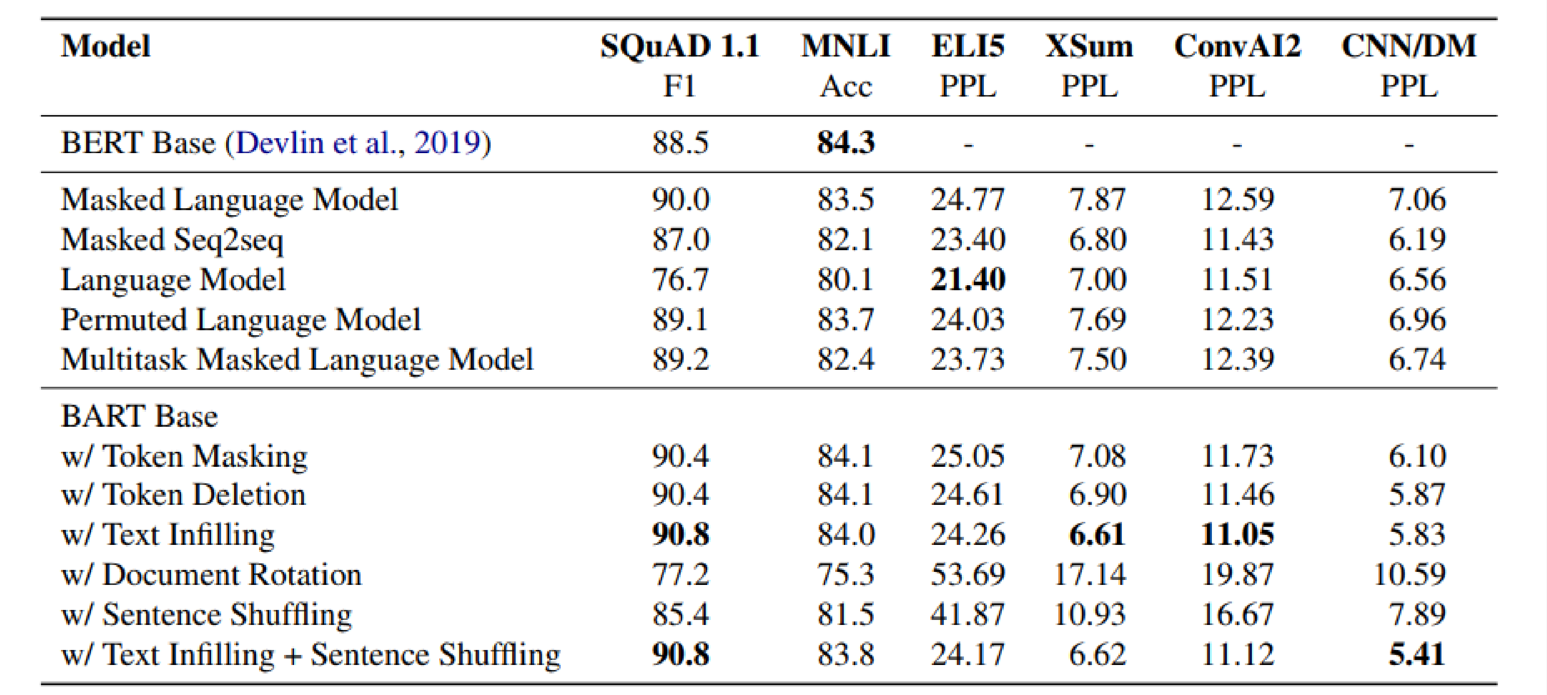
   예를 들어, simple LM 모델은 ELI5에서 가장 높은 성능을 보이지만, SQUAD에서 가장 낮은 성능을 보임

2. Token Masking 은 성능 향상에 중요하게 기여함

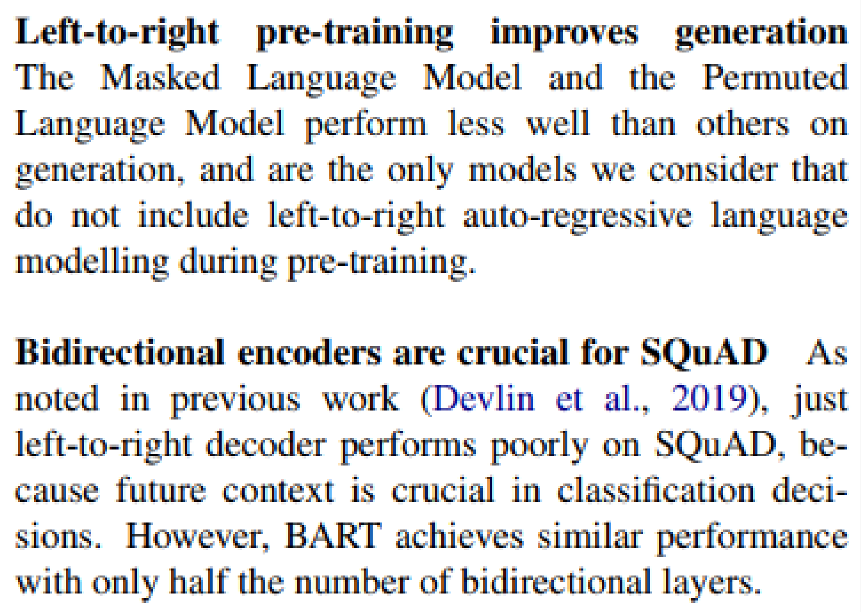
   rotation 또는 permutation은 단독으로 사용될 때 성능을 높이지 못함

   token deletion 또는 masking 이 성능을 높임

   특히 deletion은 생성 task에서 masking보다 뛰어남



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**



Left-to-right pre-training improves generation The Masked Language Model and the Permuted Language Model perform less well than others on generation, and are the only models we consider that do not include left-to-right auto-regressive language modelling during pre-training.

Bidirectional encoders are crucial for SQuAD As noted in previous work (Devlin et al., 2019), just left-to-right decoder performs poorly on SQuAD, because future context is crucial in classification decisions. However, BART achieves similar performance with only half the number of bidirectional layers.

3. Left-to-right 사전 학습은 생성 task 성능을 높임

MLM과 PLM은 다른 모델보다 생성 task에서 성능이 낮음
두 모델은 사전학습 단계에서 left-to-right 사전학습이 이뤄지지 않는 유일한 모델임

4. Bidirectional encoder는 SQuAD에서 중요하게 작용함

SQuAD에서는 뒷부분의 맥락이 중요하기 때문에,
left-to-right decoder의 성능이 떨어짐
그러나 BART는 절반의 bidirectional layer 수 만으로 비슷한 성능을 달성함

1. Introduction

2. Model

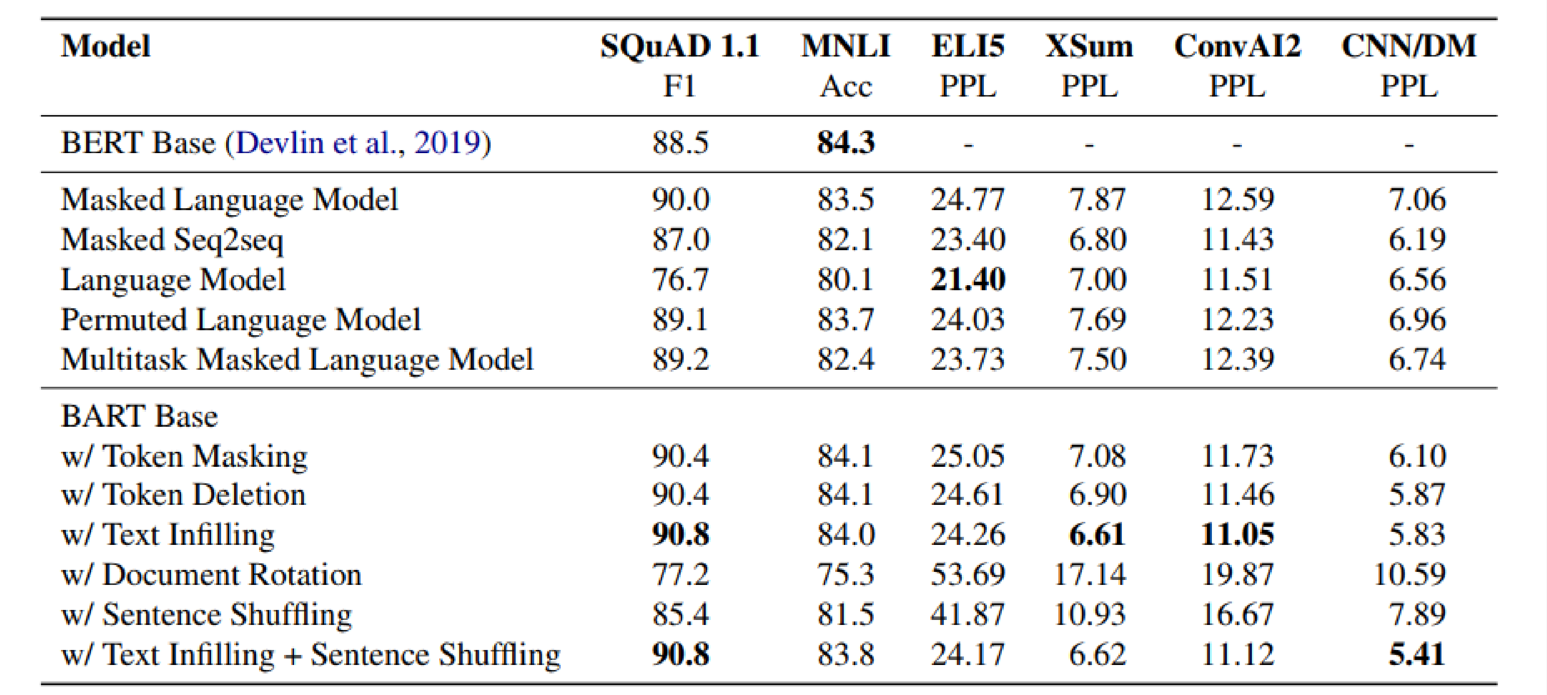
3. Fine-tuning BART

4. Comparing Pre-training Objectives

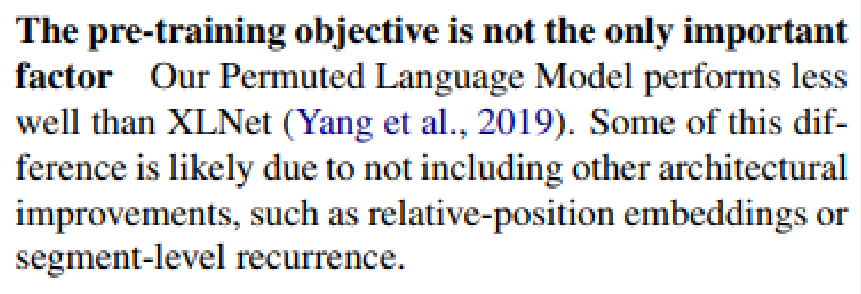
5. Large-scale Pre-training Experiments

6. Qualitative Analysis

7. Related Work



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**



The pre-training objective is not the only important factor Our Permuted Language Model performs less well than XLNet (Yang et al., 2019). Some of this difference is likely due to not including other architectural improvements, such as relative-position embeddings or segment-level recurrence.

5. 사전학습 방식(objective)만 중요한 것이 아님

본 논문에서 학습시킨 PLM은 XLNet보다 성능이 낮음
이러한 차이의 일부는 relative-position embedding 또는 segment-level recurrence 등 다른 요소들이 제외되었기 때문으로 보임

BART

1. Introduction

2. Model

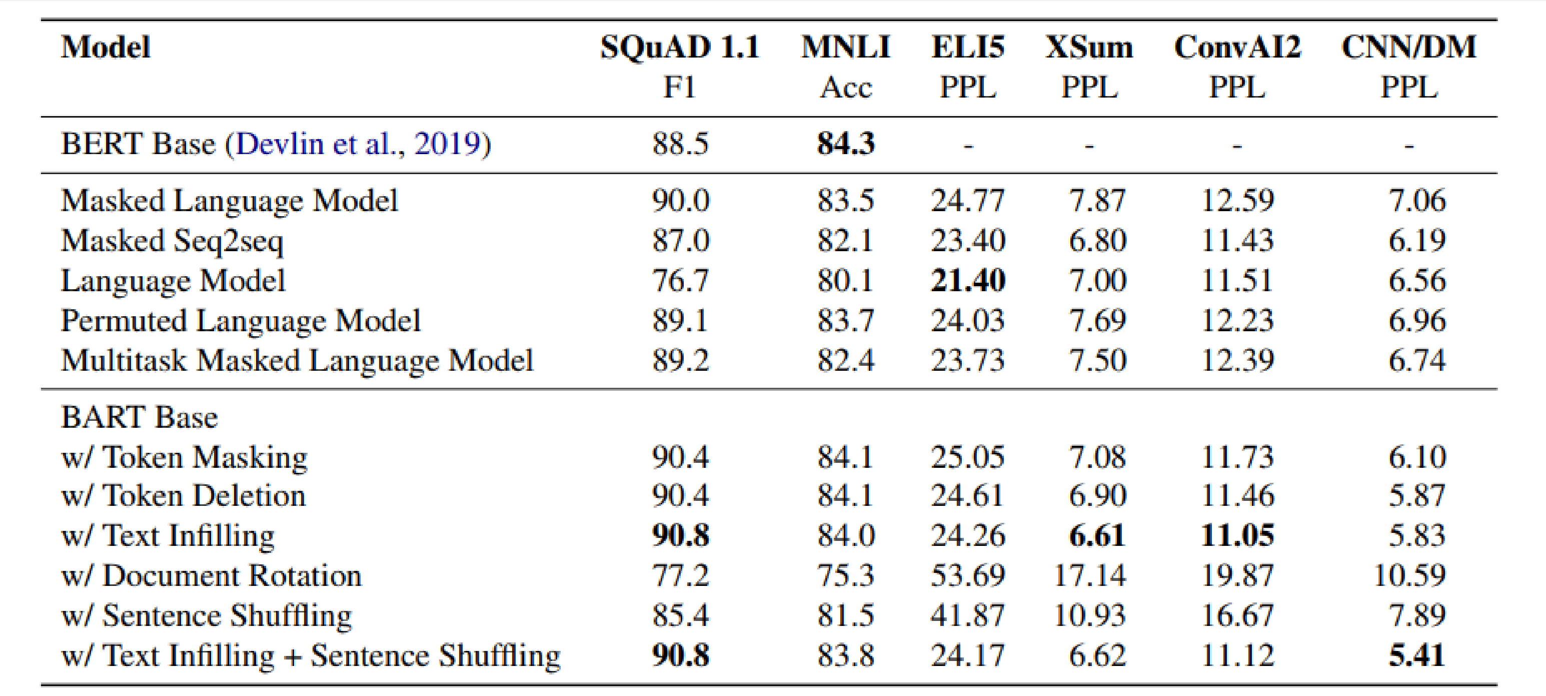
3. Fine-tuning BART

4. Comparing Pre-training Objectives

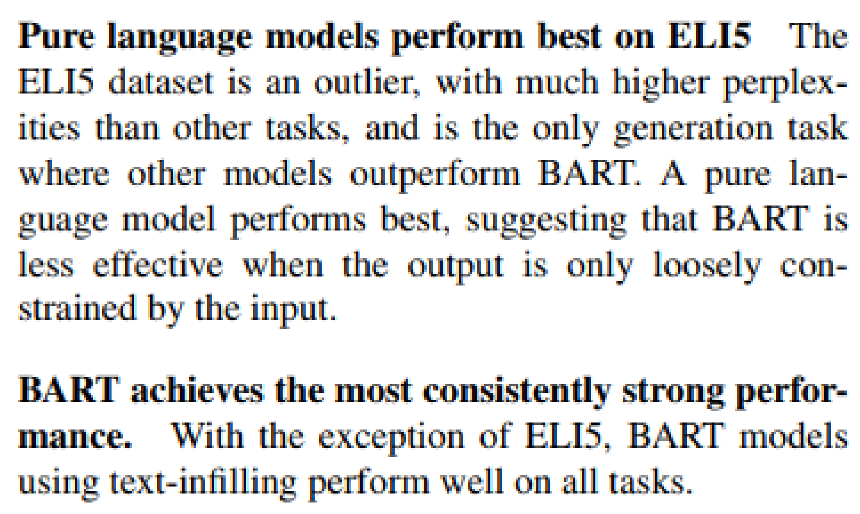
5. Large-scale Pre-training Experiments

6. Qualitative Analysis

7. Related Work



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**



Pure language models perform best on ELI5 The ELI5 dataset is an outlier, with much higher perplexities than other tasks, and is the only generation task where other models outperform BART. A pure language model performs best, suggesting that BART is less effective when the output is only loosely constrained by the input.

BART achieves the most consistently strong performance. With the exception of ELI5, BART models using text-infilling perform well on all tasks.

6. Language Model은 ELI5에서 다른 모델을 압도함

ELI 는 BART와 성능 1위 모델 간의 차이가 컸던 유일한 데이터임
1위 모델은 LM으로, 이는 output이 input에 의해 느슨하게 제약되면(?) BART의 성능이 떨어짐을 시사함

7. BART는 가장 일관되게 높은 성능을 유지한 모델임

ELI5를 제외하면 BART 모델(text infilling + sentence shuffling)은 모든 task에서 높은 성능을 유지함

1. Introduction

2. Model

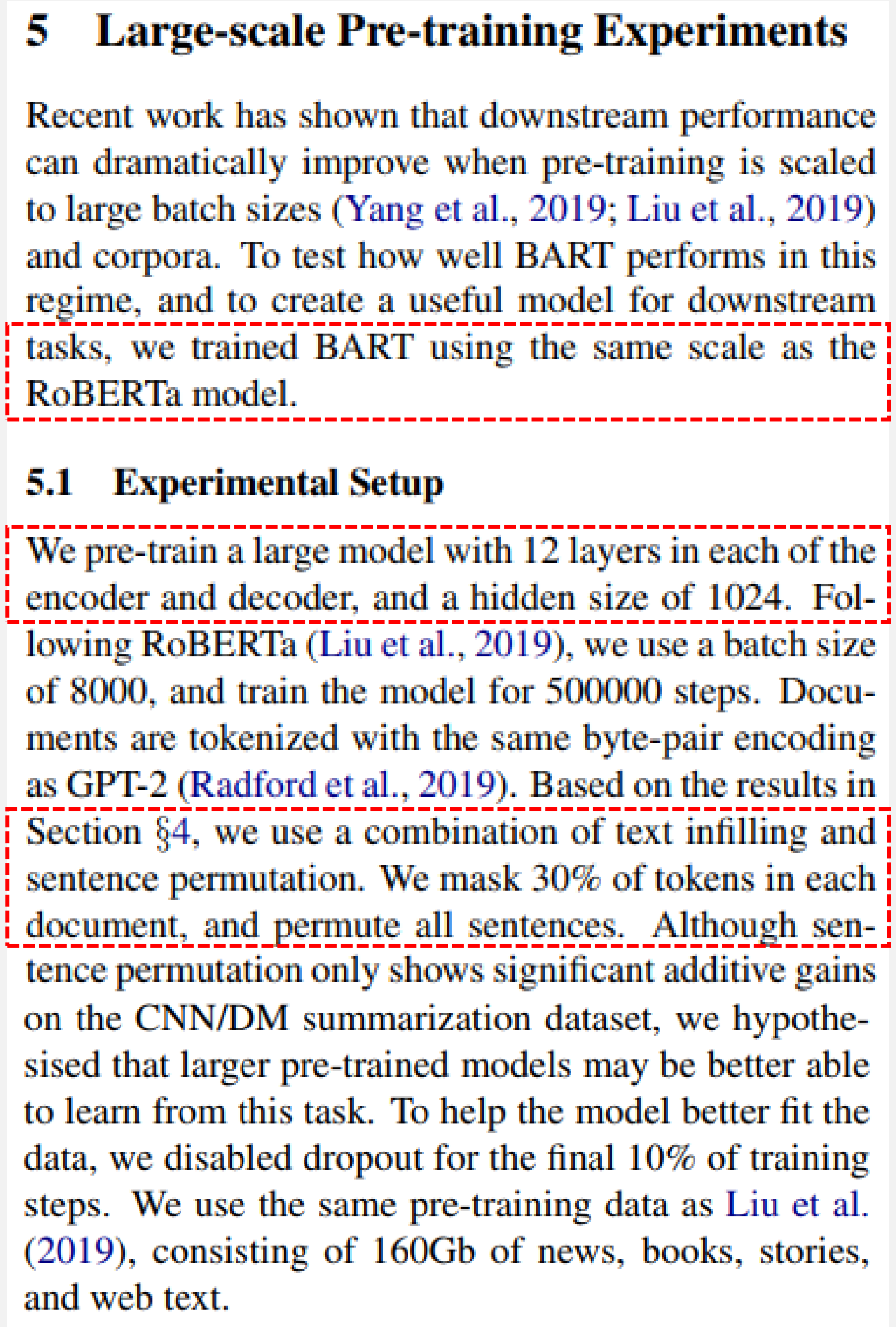
3. Fine-tuning BART

4. Comparing Pre-training Objectives

5. Large-scale Pre-training Experiments

6. Qualitative Analysis

7. Related Work



5 Large-scale Pre-training Experiments

Recent work has shown that downstream performance can dramatically improve when pre-training is scaled to large batch sizes (Yang et al., 2019; Liu et al., 2019) and corpora. To test how well BART performs in this regime, and to create a useful model for downstream tasks, we trained BART using the same scale as the RoBERTa model.

5.1 Experimental Setup

We pre-train a large model with 12 layers in each of the encoder and decoder, and a hidden size of 1024. Following RoBERTa (Liu et al., 2019), we use a batch size of 8000, and train the model for 500000 steps. Documents are tokenized with the same byte-pair encoding as GPT-2 (Radford et al., 2019). Based on the results in Section §4, we use a combination of text infilling and sentence permutation. We mask 30% of tokens in each document, and permute all sentences. Although sentence permutation only shows significant additive gains on the CNN/DM summarization dataset, we hypothesised that larger pre-trained models may be better able to learn from this task. To help the model better fit the data, we disabled dropout for the final 10% of training steps. We use the same pre-training data as Liu et al. (2019), consisting of 160Gb of news, books, stories, and web text.

large BART model의 성능을 검증하기 위해,
RoBERTa와 같은 조건으로 학습을 수행함

large BART model은
(12 encoder layer + 12 decoder layer)로 구성됨

앞서 기본 모델의 실험 결과에서 확인한 내용에 근거해,
text infilling & sentence permuation으로
사전 학습을 진행함

BART

1. Introduction

2. Model

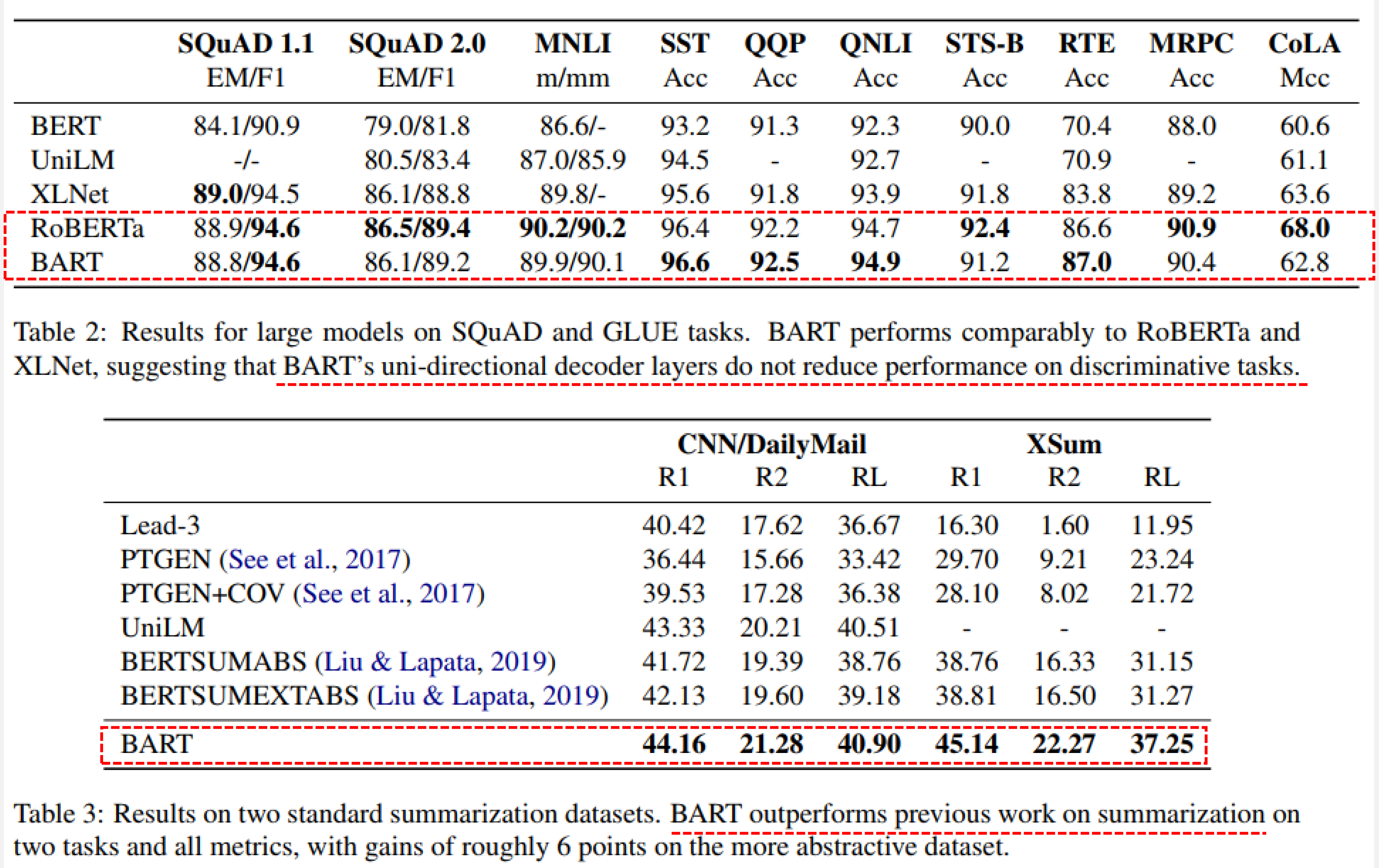
3. Fine-tuning BART

4. Comparing Pre-training Objectives

5. Large-scale Pre-training Experiments

6. Qualitative Analysis

7. Related Work



	SQuAD 1.1	SQuAD 2.0	MNLI	SST	QQP	QNLI	STS-B	RTE	MRPC	CoLA
	EM/F1	EM/F1	m/mm	Acc	Acc	Acc	Acc	Acc	Acc	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

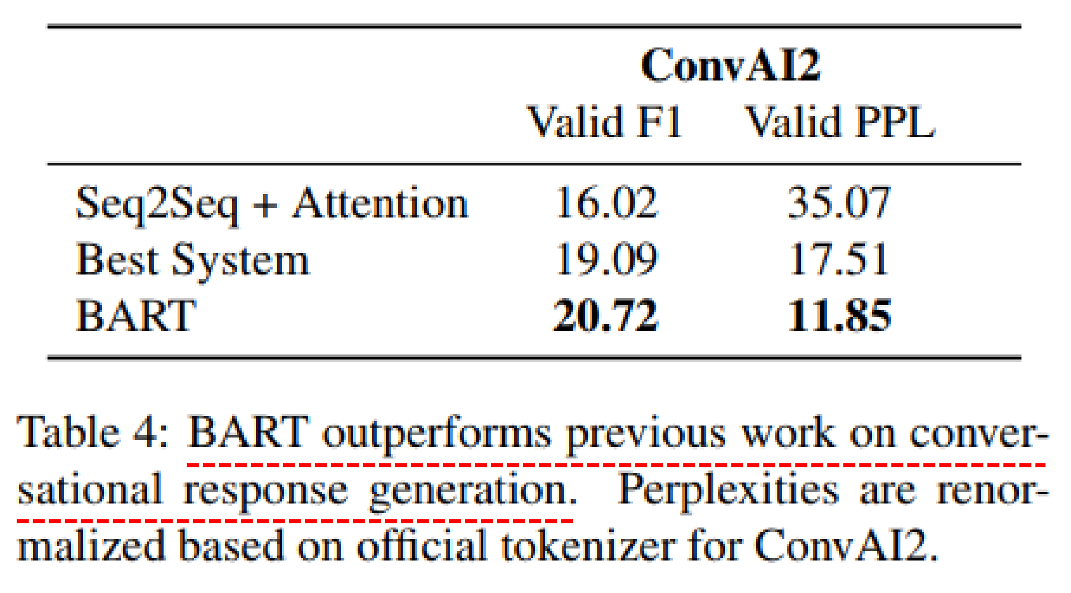
Table 2: Results for large models on SQuAD and GLUE tasks. BART performs comparably to RoBERTa and XLNet, suggesting that BART's uni-directional decoder layers do not reduce performance on discriminative tasks.

	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.

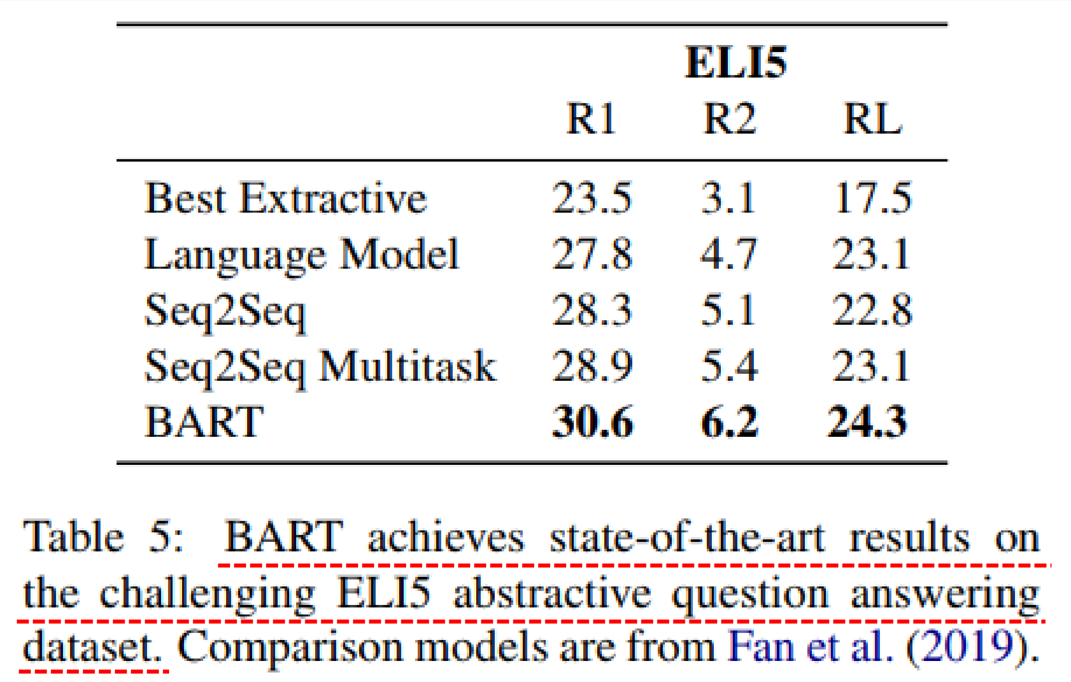
BART

1. Introduction
2. Model
3. Fine-tuning BART
4. Comparing Pre-training Objectives
5. Large-scale Pre-training Experiments
6. Qualitative Analysis
7. Related Work



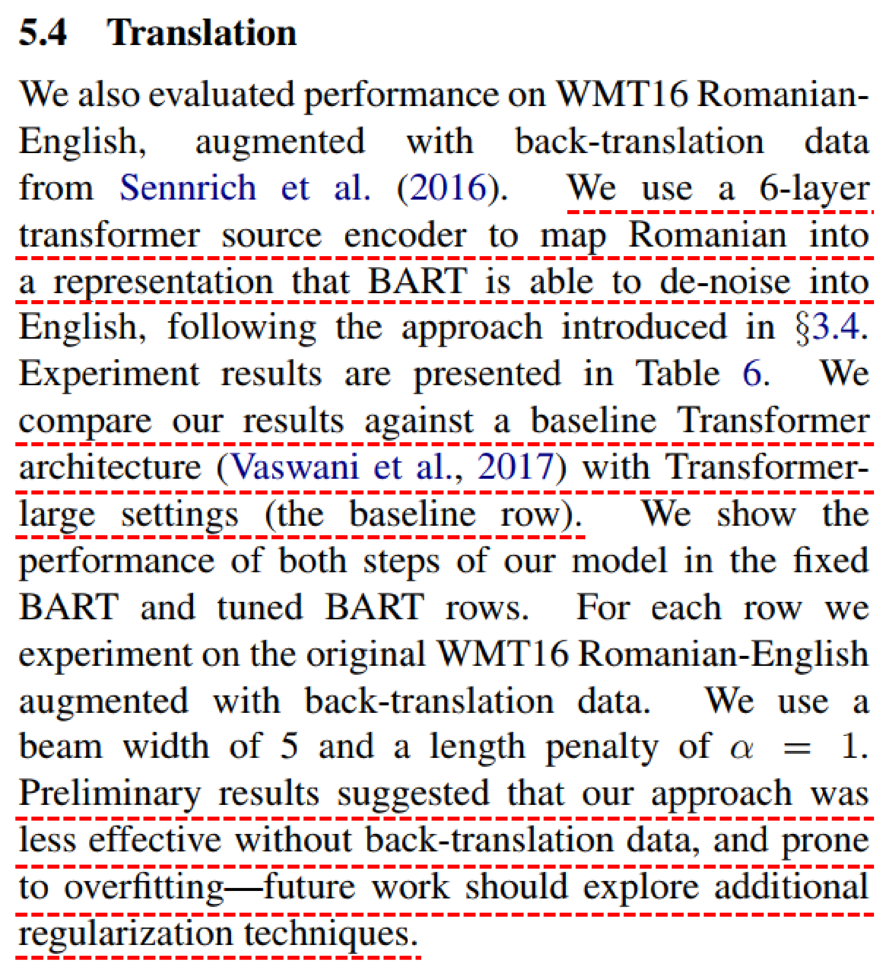
	ConvAI2	
	Valid F1	Valid PPL
Seq2Seq + Attention	16.02	35.07
Best System	19.09	17.51
BART	**20.72**	**11.85**

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



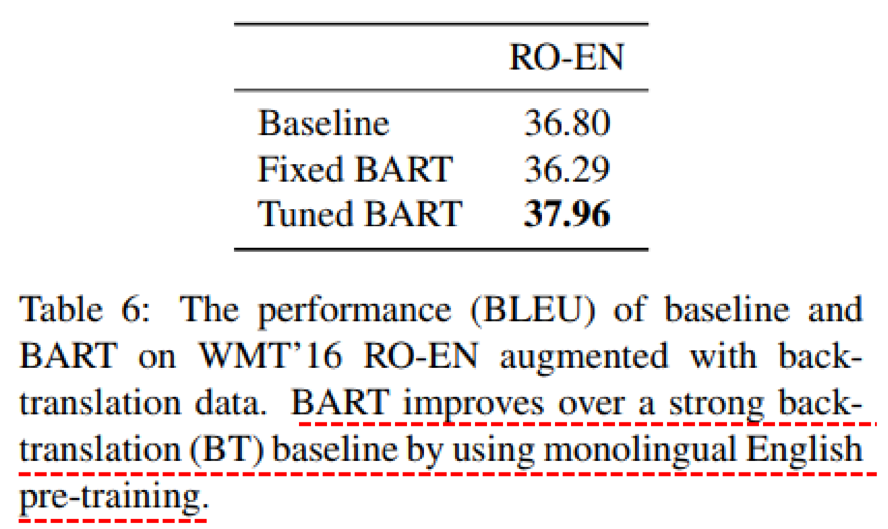
	ELI5		
	R1	R2	RL
Best Extractive	23.5	3.1	17.5
Language Model	27.8	4.7	23.1
Seq2Seq	28.3	5.1	22.8
Seq2Seq Multitask	28.9	5.4	23.1
BART	**30.6**	**6.2**	**24.3**

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).



5.4 Translation

We also evaluated performance on WMT16 Romanian-English, augmented with back-translation data from Sennrich et al. (2016). We use a 6-layer transformer source encoder to map Romanian into a representation that BART is able to de-noise into English, following the approach introduced in §3.4. Experiment results are presented in Table 6. We compare our results against a baseline Transformer architecture (Vaswani et al., 2017) with Transformer-large settings (the baseline row). We show the performance of both steps of our model in the fixed BART and tuned BART rows. For each row we experiment on the original WMT16 Romanian-English augmented with back-translation data. We use a beam width of 5 and a length penalty of $\alpha = 1$. Preliminary results suggested that our approach was less effective without back-translation data, and prone to overfitting—future work should explore additional regularization techniques.



	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.

- 6-layer의 encoder를 추가해 병렬 코퍼스로 파인 튜닝함
- baseline Transformer(2017)와 성능을 비교함
- Tuned BART를 사용할 때만 baseline보다 높은 성능
- back-translation 데이터를 사용하지 않으면 성능이 떨어졌고, overfitting에도 취약함 (후속 연구 예고)

BART

1. Introduction

2. Model

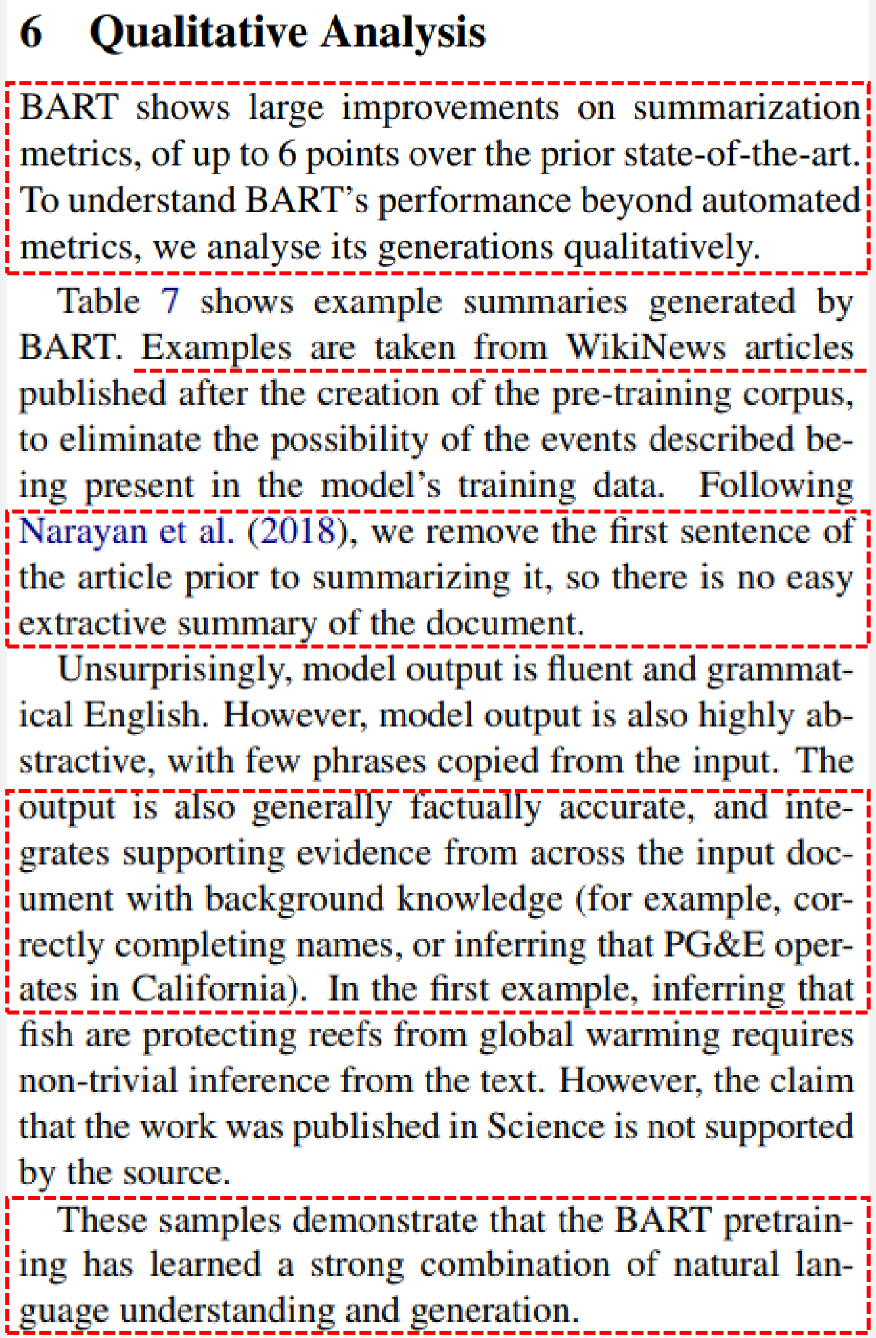
3. Fine-tuning BART

4. Comparing Pre-training Objectives

5. Large-scale Pre-training Experiments

6. Qualitative Analysis

7. Related Work



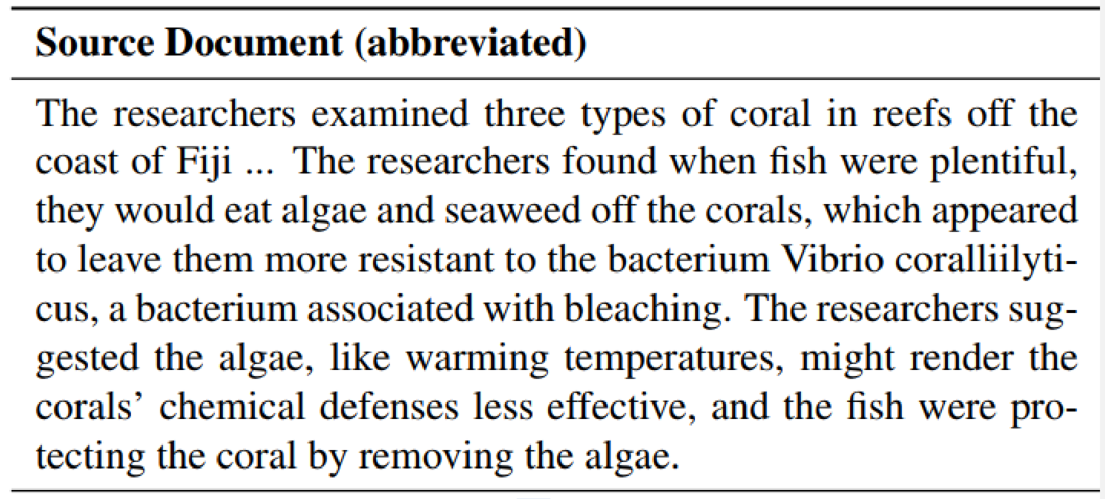
6 Qualitative Analysis

BART shows large improvements on summarization metrics, of up to 6 points over the prior state-of-the-art. To understand BART's performance beyond automated metrics, we analyse its generations qualitatively.

Table 7 shows example summaries generated by BART. Examples are taken from WikiNews articles published after the creation of the pre-training corpus, to eliminate the possibility of the events described being present in the model's training data. Following Narayan et al. (2018), we remove the first sentence of the article prior to summarizing it, so there is no easy extractive summary of the document.

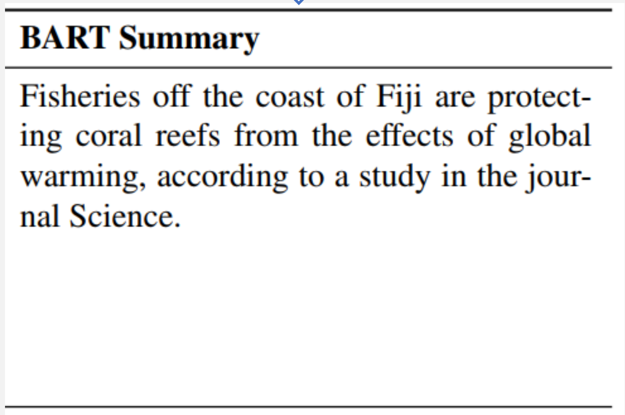
Unsurprisingly, model output is fluent and grammatical English. However, model output is also highly abstractive, with few phrases copied from the input. The output is also generally factually accurate, and integrates supporting evidence from across the input document with background knowledge (for example, correctly completing names, or inferring that PG&E operates in California). In the first example, inferring that fish are protecting reefs from global warming requires non-trivial inference from the text. However, the claim that the work was published in Science is not supported by the source.

These samples demonstrate that the BART pretraining has learned a strong combination of natural language understanding and generation.



Source Document (abbreviated)

The researchers examined three types of coral in reefs off the coast of Fiji ... The researchers found when fish were plentiful, they would eat algae and seaweed off the corals, which appeared to leave them more resistant to the bacterium Vibrio coralliilyticus, a bacterium associated with bleaching. The researchers suggested the algae, like warming temperatures, might render the corals' chemical defenses less effective, and the fish were protecting the coral by removing the algae.



BART Summary

Fisheries off the coast of Fiji are protecting coral reefs from the effects of global warming, according to a study in the journal Science.

- BART의 요약 성능을 지표뿐만 아니라 질적으로도 분석함
- 학습에 포함되지 않은 WikiNews 기사에서 첫 문장을 제외하고 요약 수행함
- 요약 결과는 상당히 그럴듯하고, 특히 사실에 기반한 추론 능력도 갖춘 것으로 보임

BART

1. Introduction

2. Model

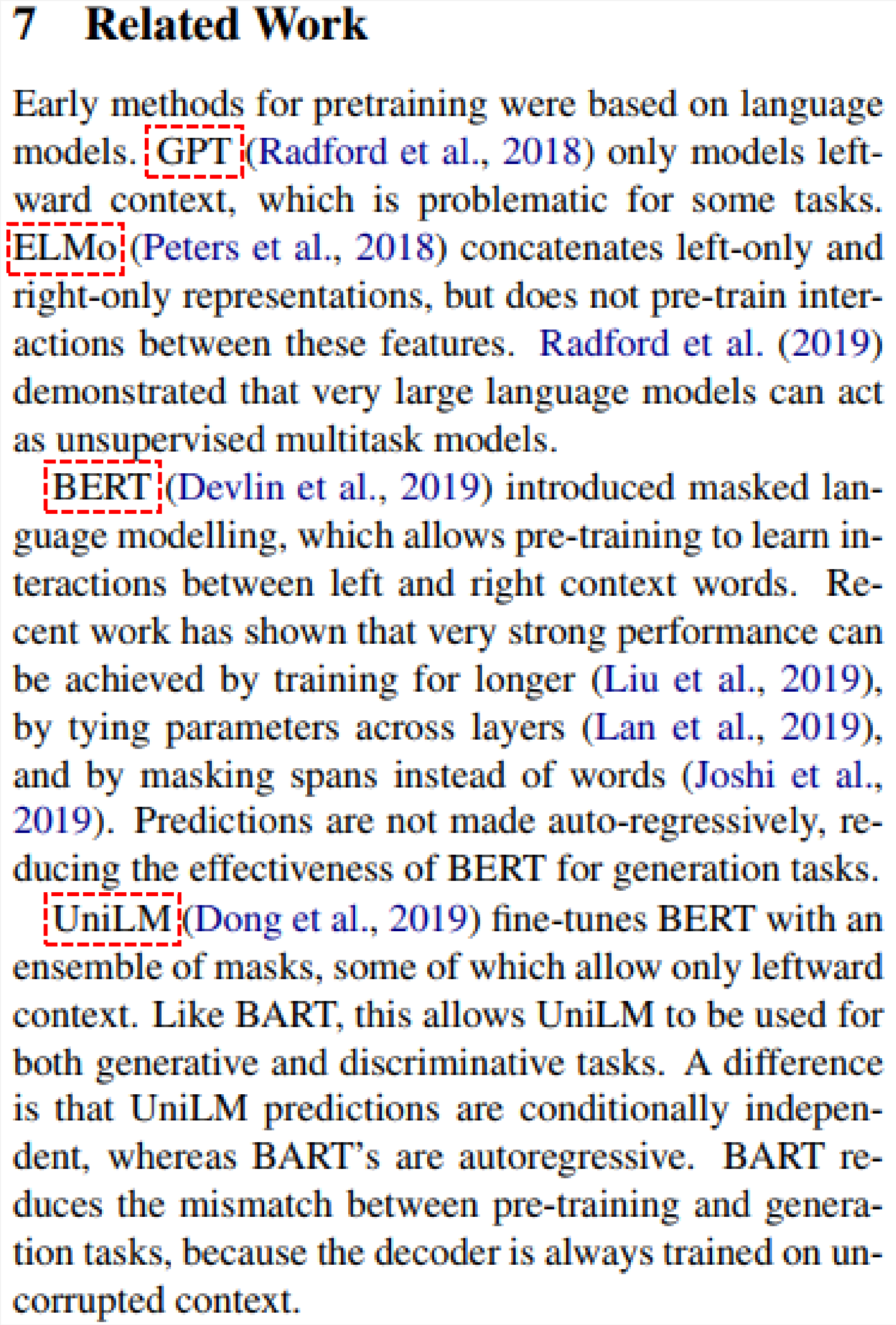
3. Fine-tuning BART

4. Comparing Pre-training Objectives

5. Large-scale Pre-training Experiments

6. Qualitative Analysis

7. Related Work

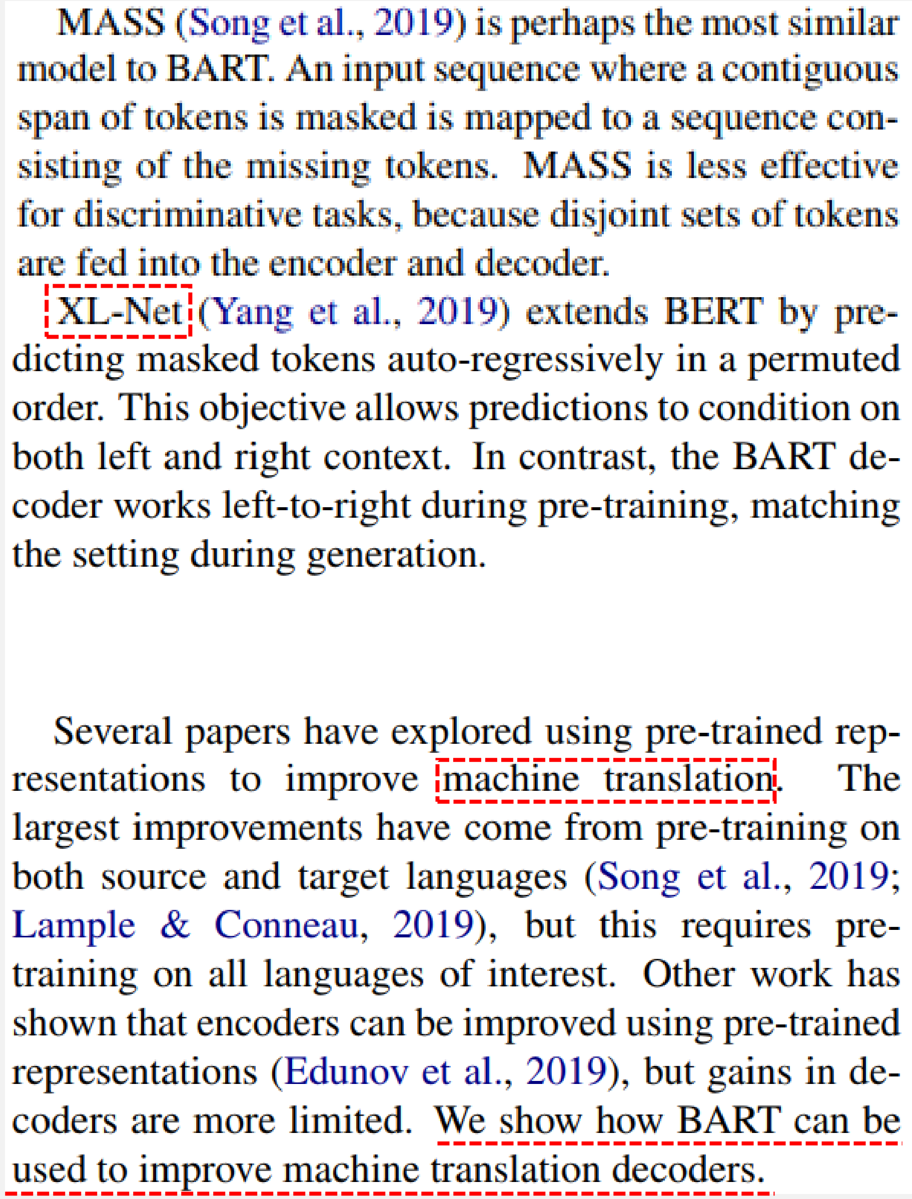


7 Related Work

Early methods for pretraining were based on language models. GPT (Radford et al., 2018) only models leftward context, which is problematic for some tasks. ELMo (Peters et al., 2018) concatenates left-only and right-only representations, but does not pre-train interactions between these features. Radford et al. (2019) demonstrated that very large language models can act as unsupervised multitask models.

BERT (Devlin et al., 2019) introduced masked language modelling, which allows pre-training to learn interactions between left and right context words. Recent work has shown that very strong performance can be achieved by training for longer (Liu et al., 2019), by tying parameters across layers (Lan et al., 2019), and by masking spans instead of words (Joshi et al., 2019). Predictions are not made auto-regressively, reducing the effectiveness of BERT for generation tasks.

UniLM (Dong et al., 2019) fine-tunes BERT with an ensemble of masks, some of which allow only leftward context. Like BART, this allows UniLM to be used for both generative and discriminative tasks. A difference is that UniLM predictions are conditionally independent, whereas BART's are autoregressive. BART reduces the mismatch between pre-training and generation tasks, because the decoder is always trained on uncorrupted context.



MASS (Song et al., 2019) is perhaps the most similar model to BART. An input sequence where a contiguous span of tokens is masked is mapped to a sequence consisting of the missing tokens. MASS is less effective for discriminative tasks, because disjoint sets of tokens are fed into the encoder and decoder.

XL-Net (Yang et al., 2019) extends BERT by predicting masked tokens auto-regressively in a permuted order. This objective allows predictions to condition on both left and right context. In contrast, the BART decoder works left-to-right during pre-training, matching the setting during generation.

Several papers have explored using pre-trained representations to improve machine translation. The largest improvements have come from pre-training on both source and target languages (Song et al., 2019; Lample & Conneau, 2019), but this requires pretraining on all languages of interest. Other work has shown that encoders can be improved using pre-trained representations (Edunov et al., 2019), but gains in decoders are more limited. We show how BART can be used to improve machine translation decoders.

References

- BART original paper
  https://arxiv.org/pdf/1910.13461.pdf
- [딥러닝논문읽기모임] BART – 진명훈
  https://www.youtube.com/watch?v=VmYMnpDLPEo
- [고려대학교 DSBA] Transformer to T5 – 이유경
  https://www.youtube.com/watch?v=v7diENO2mEA