2024-1 자연어 세미나

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

Mike Lewis, Yinhan Liu, Naman Goyal et al., ACL, 2020

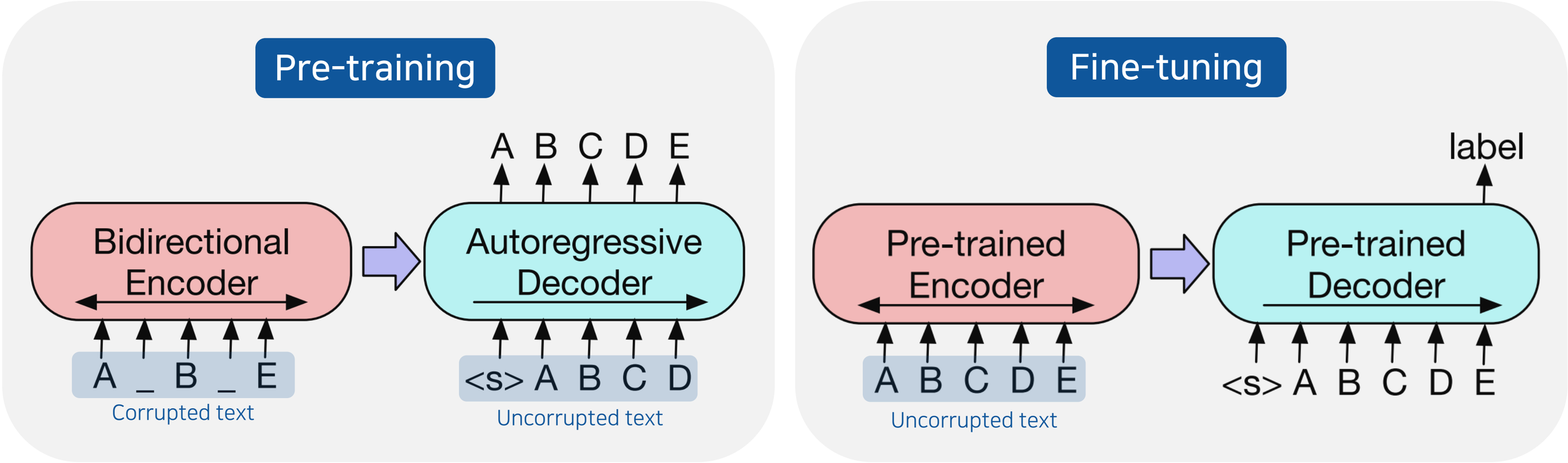
인공지능 연구실 석사 과정 3기 권예담

CONTENTS

Overview

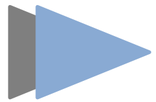
BART: Denoising autoencoder for pre-training seq2seq models

1. Corrupting text with an arbitrary noising function
2. Learning a model to reconstruct the original text



Introduction

- We present BART, denoising autoencoder
  - pre-trains a model combining Bidirectional and Auto-Regressive Transformers
  - Text is corrupted with an arbitrary noising function
  - Seq2seq model is learned to reconstruct the original text



Generalizing BERT, GPT, and recent pretraining schemes

- BART is effective for both text generation and comprehension tasks
  - Comparing pre-training objectives
  - Text generation: abstractive summarization, dialogue, abstractive question answering
  - Comprehension: NLI, extractive question answering, semantic similarity, classification

BART

- Architecture
  - For base model, we use 6 layers in the encoder and decoder
  - For large model, we use 12 layers in the encoder and decoder
  - 동일한 사이즈(= 같은 레이어 개수)의 BERT에 비해 약 10% 더 많은 파라미터
- Transformations for noising the input

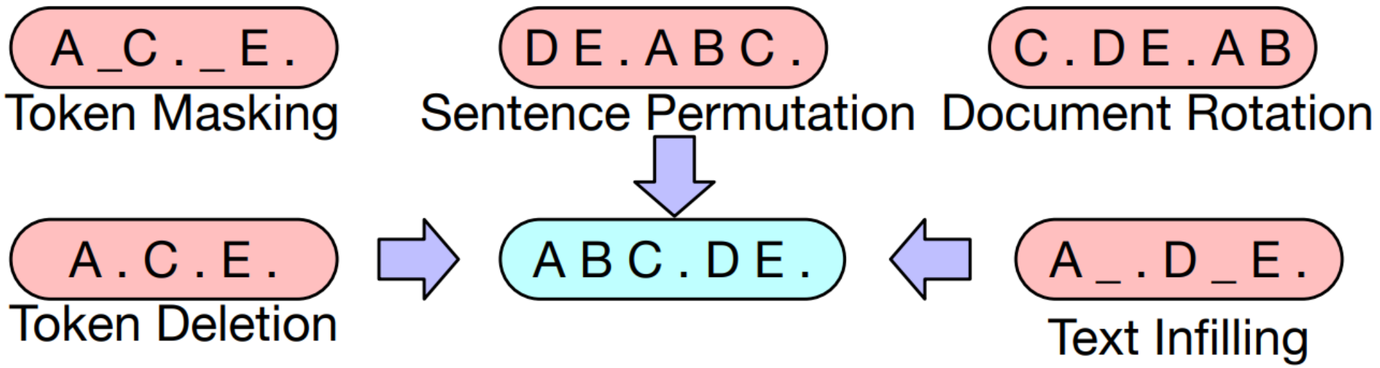


Figure 2: Transformations for noising the input that we experiment with. These transformations can be composed.

Pre-training BART

Token Masking

Following BERT, 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.

Text Infilling

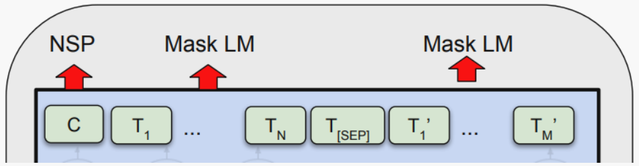
A number of text spans are sampled, with span lengths drawn from a Poisson distribution.
Each span is replaced with a single [MASK] token.
In contrast to SpanBERT, the model is learned to predict how many tokens are missing from a span.

Pre-training BART

Text Infilling

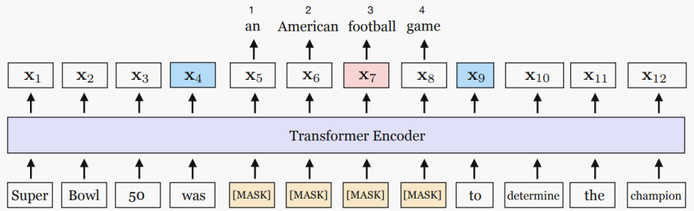
A number of text spans are sampled with span lengths drawn from a Poisson distribution.
Each span is replaced with a single [MASK] token.
In contrast to SpanBERT, the model is learned to predict how many tokens are missing from a span.

BERT



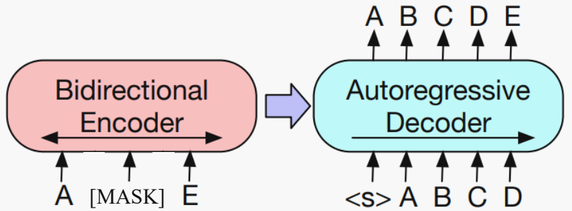
masking random tokens

SpanBERT



masking contiguous random spans
drawn from a clamped geometric distribution
and replacing each span
with [MASK] tokens of the same length

Text Infilling



masking contiguous random spans
drawn from a Poisson distribution
and replacing each span
with a single [MASK] token

* BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding (NAACL, 2019)
* SpanBERT: Improving Pre-training by Representing and Predicting Spans (TACL, 2020)

Pre-training BART

Sentence Permutation

A document is divided into sentences based on full stops, and these sentences are shuffled in a random order.

Document

sentence 1

sentence 2

sentence 3

sentence 4

Document

sentence 4

sentence 2

sentence 1

sentence 3

Document Rotation

A random token is chosen uniformly, and the document is rotated so that it begins with that token. This task trains the model to identify the start of the document.

Document

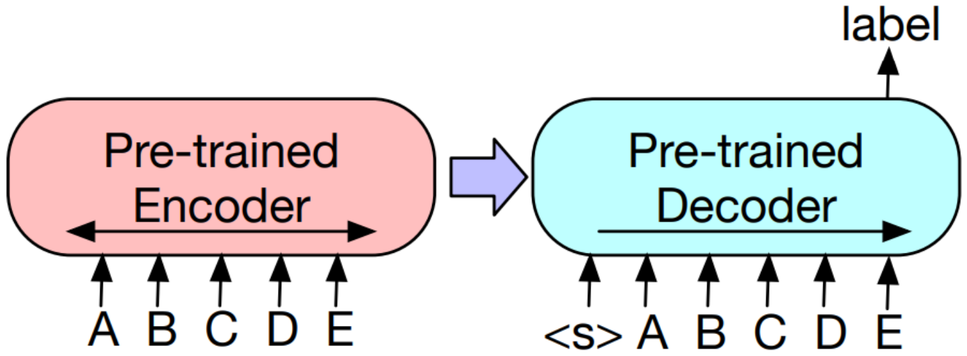
Lorem Ipsum is simply dummy text of the printing and **typesetting** industry. Lorem Ipsum has been the industry's standard dummy text.

Document

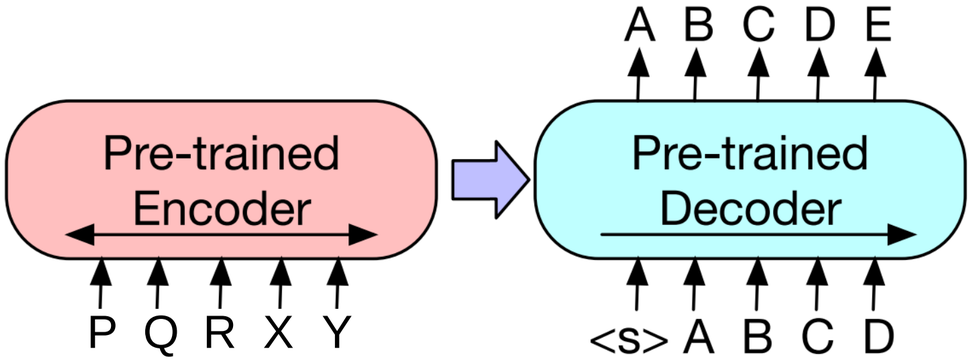
typesetting industry. Lorem Ipsum has been the industry's standard dummy text. **Lorem Ipsum** is simply dummy text of the printing and

Fine-tuning BART

Classification



Generation



Sequence Classification

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.

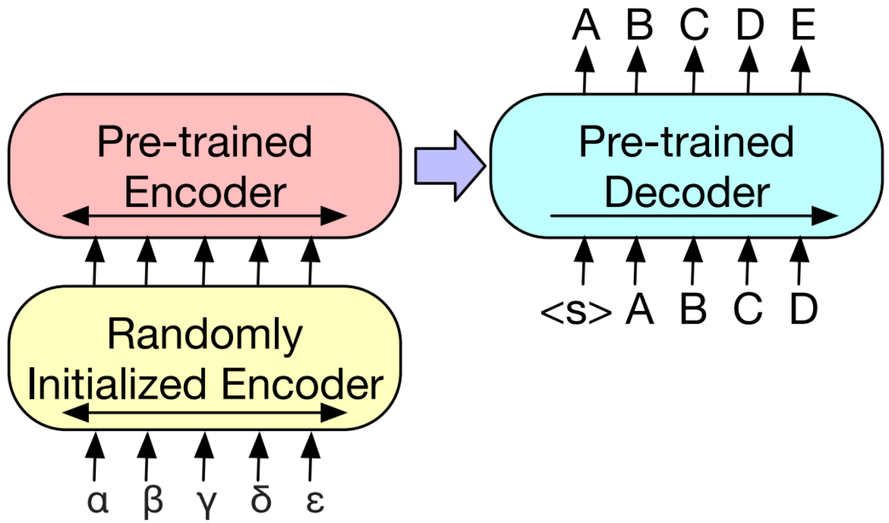
Token Classification

The same input is fed into the encoder and decoder, and use the top hidden state of the decoder as a representation for each word.

Sequence Generation

the encoder input is the input sequence, and the decoder generates outputs autoregressively.

Fine-tuning BART



Machine Translation

- 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.
- In the first step, we freeze most of BART parameters and only update the randomly initialized encoder, the 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.

Comparing Pre-training Objectives

Comparison Objectives

- 많은 pre-training objectives가 제안되었지만 공평한 비교 어려움
  - Differences in training data, training resources, architectural differences, fine-tuning procedures
- We re-implement strong pre-training approaches recently proposed.
  - Minor changes to the learning rate and usage of layer normalization
- We compare our implementations with published numbers from BERT
  - trained for 1M steps on a combination of books and Wikipedia data

Comparing Pre-training Objectives

Comparison Objectives

GPT Language Model

- Train a left-to-right Transformer language model.

XLNet Permuted Language Model

- We sample 1/6 of the tokens, and generate them in a random order autoregressively.
- We do not implement the relative positional embeddings or attention across segments from XLNet.

BERT Masked Language Model

- We replace 15% of tokens with [MASK] symbols, and train the model to independently predict the original tokens.

UniLM Multitask Masked Language Model

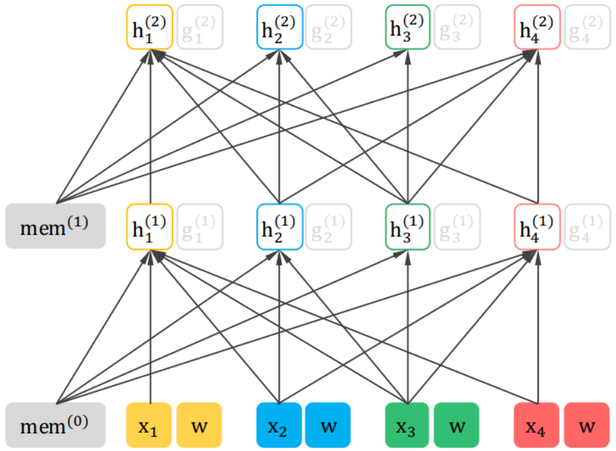
- We train the model with additional self-attention masks.
- 1/6 left-to-right, 1/6 right-to-left, 1/3 unmasked, and 1/3 with the first 50% of tokens unmasked and a left-to-right mask for the remainder.

MASS Masked Seq-to-Seq

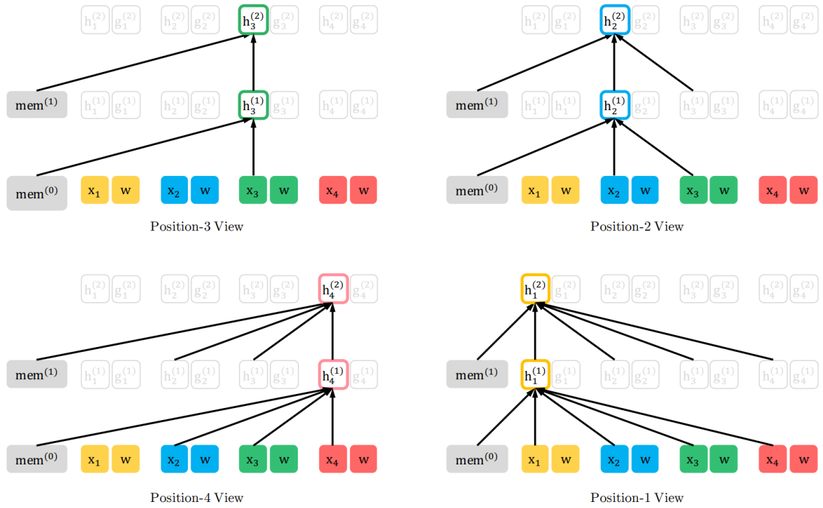
- We mask a span containing 50% of tokens, and train a seq2seq model to predict the masked tokens.

Appendix

XLNet



Joint View of the Content Stream
(Factorization order: 3 → 2 → 4 → 1)



Split View of the Content Stream
(Factorization order: 3 → 2 → 4 → 1)

To formalize the idea, let $\mathcal{Z}_T$ be the set of all possible permutations of the length-T index sequence $[1, 2, \ldots, T]$. We use z_t and $\mathbf{z}_{<t}$ to denote the t-th element and the first $t-1$ elements of a permutation $\mathbf{z} \in \mathcal{Z}_T$. Then, our proposed permutation language modeling objective can be expressed as follows:

$$\max_{\theta} \quad \mathbb{E}_{\mathbf{z} \sim \mathcal{Z}_T} \left[\sum_{t=1}^{T} \log p_\theta(x_{z_t} \mid \mathbf{x}_{\mathbf{z}_{<t}}) \right]. \tag{3}$$

Essentially, for a text sequence $\mathbf{x}$, we sample a factorization order $\mathbf{z}$ at a time and decompose the likelihood $p_\theta(\mathbf{x})$ according to factorization order. Since the same model parameter θ is shared across all factorization orders during training, in expectation, x_t has seen every possible element $x_i \neq x_t$ in the sequence, hence being able to capture the bidirectional context. Moreover, as this objective fits into the AR framework, it naturally avoids the independence assumption and the pretrain-finetune discrepancy discussed in Section 2.1.

Comparing Pre-training Objectives

Tasks

- SQuAD
  - Token classification with Extractive Question Answering
  - [Input] document, question [Output] Answer start position
- MNLI
  - Sequence Classification to predict whether one sentence entails another
  - [Input] premise, hypothesis [Output] Label
- ELI5
  - Abstractive Question Answering
  - [Input] Supported documents, question [Output] Answer Generation
- CNN/DM & XSum
  - Abstractive Summarization
  - [Input] documents [Output] Summary Generation
- ConvAI2
  - Dialogue response generation
  - [Input] Previous dialog, persona [Output] Response Generation

Comparing Pre-training Objectives

Results

	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
Permutation	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
- Token deletion, masking >>> document rotation, sentence permutation → Token masking is crucial
- Masked LM, Permuted LM perform less well than others on generation → Left-to-right pre-training improves generation
- Future context is crucial in classification decisions → Bidirectional encoders are crucial for SQuAD
- The pre-training objectives is not the only important factor
- Pure LM perform best on ELI5 → BART is less effective when the output is only loosely constrained by the input
- BART achieves the most consistently strong performance

Large-scale Pre-training Experiments

Experimental Setup

- Recent work has shown that downstream performance can dramatically improve when pre-training is scaled to large batch sizes and corpora

- We pre-train a large model with 12 layers in each of encoder and decoder
  We use a batch size of 8000, and train the model for 500000 steps

- We use a combination of text infilling and sentence permutation
  - We mask 30% of tokens in each document, and permute all sentences

- We use pre-training data, consisting of 160Gb of news, books, stories, and web text

Large-scale Pre-training Experiments

Discriminative Tasks

	MNLI m/mm	SST Acc	QQP Acc	QNLI Acc	STS-B Acc	RTE Acc	MRPC Acc	CoLA Mcc
BERT	86.6/-	93.2	91.3	92.3	90.0	70.4	88.0	60.6
UniLM	87.0/85.9	94.5	-	92.7	-	70.9	-	61.1
XLNet	89.8/-	95.6	91.8	93.9	91.8	83.8	89.2	63.6
RoBERTa	**90.2/90.2**	96.4	92.2	94.7	**92.4**	86.6	**90.9**	**68.0**
BART	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 GLUE tasks. BART performs comparably to RoBERTa and XLNet, suggesting that BART's uni-directional decoder layers do not reduce performance on discriminative tasks.

- The most directly comparable baseline is RoBERTa, which was pre-trained with the same resources, but a different objective.
- Overall, BART performs similarly, with only small differences between the models on most tasks.
- BART's improvements on generation tasks do not come at expense of classification performance.

Large-scale Pre-training Experiments

Generation 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
ROBERTASHARE (Rothe et al., 2019)	40.31	18.91	37.62	41.45	18.79	33.90
BART	**44.16**	**21.28**	**40.90**	**45.14**	**22.27**	**37.25**

Table 4: Results on two standard summarization datasets. BART outperforms previous work on summarization on both tasks and all metrics, including those based on large-scale pre-training.

- During fine-tuning, we use a label smoothed cross entropy loss
- CNN/DM tends to resemble source sentences → Nevertheless, BART outperforms all existing work.
- XSum is highly abstractive, and extractive models perform poorly → BART achieves a significant advance in performance.

Large-scale Pre-training Experiments

Generation Tasks

	ConvAI2	
	Valid F1	Valid PPL
Seq2Seq + Attention	16.02	35.07
Best System [2]	19.09	17.51
BART	**20.72**	**11.85**

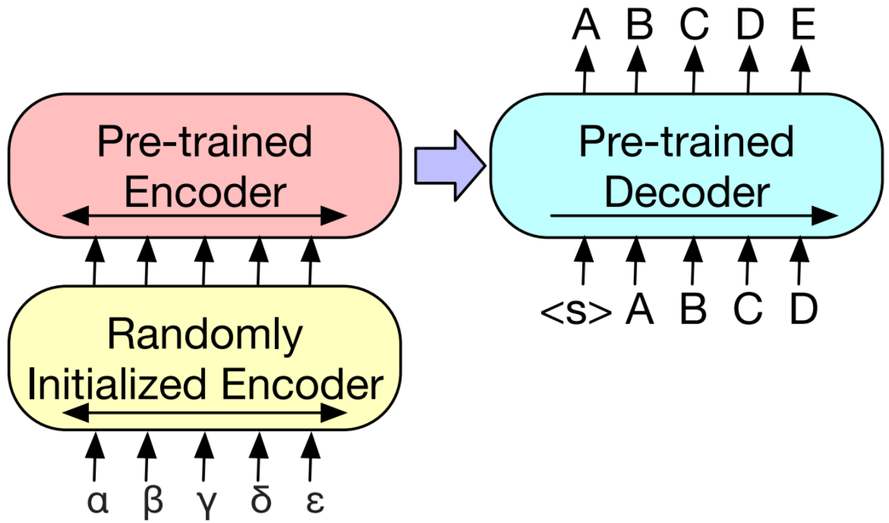
Table 6: BART outperforms previous work on conversational response generation. Perplexities are renormalized based on official tokenizer for ConvAI2.

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

- During fine-tuning, we use a label smoothed cross entropy loss
- ConvAI2, in which agents must generate responses conditioned on both the previous context and a persona.
- ELI5, the dataset remains a challenging, because answers are only weakly specified by the question.

Large-scale Pre-training Experiments



Machine Translation

- 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.
- In the first step, we freeze most of BART parameters and only update the randomly initialized encoder, the 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.

Large-scale Pre-training Experiments

Translation

		RO-EN
Transformer-Large	Baseline	36.80
First step	Fixed BART	36.29
Second step	Tuned BART	**37.96**

Table 8: BLEU scores of the baseline and BART on WMT'16 RO-EN augmented with back-translation data. BART improves over a strong back-translation baseline by using monolingual English pre-training.

- We evaluated performance on WMT16 Romanian-English, augmented with back-translation data.
- We use a 6-layer source encoder to map Romanian into a representation that a representation that BART is able to de-noise into English.
- Our approach was less effective without back-translation data, and prone to overfitting—future work should explore additional regularization techniques.

Conclusion

- We introduced BART, a pre-training approach that learns to map corrupted documents to the original.
- We compared strong pre-training approaches recently proposed, and BART demonstrates consistent improvement.
- BART achieves similar performance to RoBERTa on classification tasks, while achieving new state-of-the-art results on a number of text generation tasks.

감사합니다.