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

Mike Lewis*, Yinhan Liu*, Naman Goyal*, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, Luke Zettlemoyer
Facebook AI
mikelewis@fb.com,yinhan@ai2incubator.com,naman@fb.com

Abstract

We present BART, a denoising autoencoder for pretraining sequence-to-sequence models. BART is trained by (1) corrupting text with an arbitrary noising function, and (2) learning a model to reconstruct the original text. It 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 other recent pretraining schemes. We evaluate a number of noising approaches, finding the best performance by both randomly shuffling the order of sentences and using a novel in-filling scheme, where spans of text are replaced with a single mask token. BART is particularly effective when fine tuned for text generation but also works well for comprehension tasks. It matches the performance of RoBERTa on GLUE and SQuAD, and achieves new state-of-the-art results on a range of abstractive dialogue, question answering, and summarization tasks, with gains of up to 3.5 ROUGE. BART also provides a 1.1 BLEU increase over a back-translation system for machine translation, with only target language pretraining. We also replicate other pretraining schemes within the BART framework, to understand their effect on end-task performance.[1]

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.

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).

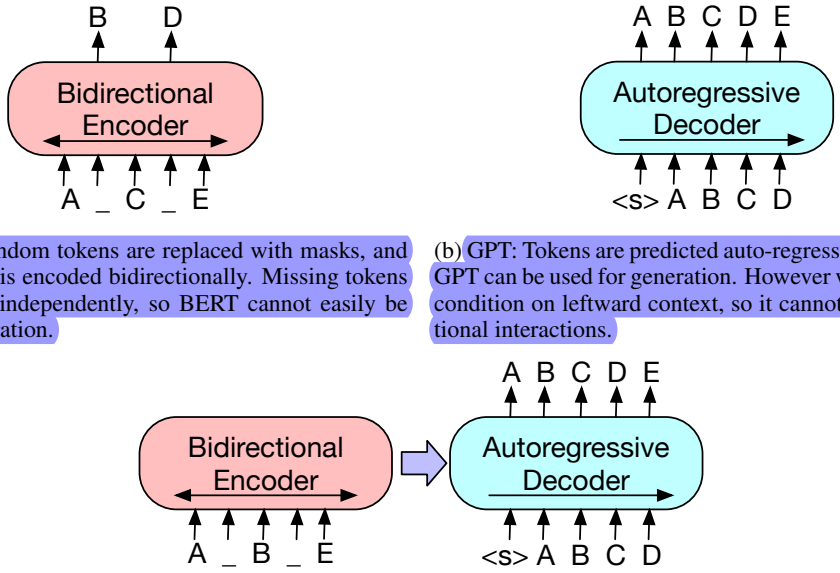
A key advantage of this setup is the noising flexibility; arbitrary transformations can be applied to the original text, including changing its length. We evaluate a number of noising approaches, finding the best performance by both randomly shuffling the order of the original sentences and using a novel in-filling scheme, where arbitrary length spans of text (including zero length) are replaced with a single mask token. This approach generalizes the original word masking and next sentence prediction objectives in BERT by forcing the model to reason more about overall sentence length and make longer range transformations to the input.

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 3.5 ROUGE over previous work on XSum (Narayan et al., 2018).

BART also opens up new ways of thinking about fine tuning. We present a new scheme for machine translation where a BART model is stacked above a few additional transformer layers. These layers are trained

[1]Code and pre-trained models for BART are available at https://github.com/pytorch/fairseq and https://huggingface.co/transformers

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 7871–7880



(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 a 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).

to essentially translate the foreign language to noised English, by propagation through BART, thereby using BART as a pre-trained target-side language model. This approach improves performance over a strong back-translation MT baseline by 1.1 BLEU on the WMT Romanian-English benchmark.

To better understand these effects, we also report an ablation analysis that replicates other recently proposed training objectives. This study allows us to carefully control for a number of factors, including data and optimization parameters, which have been shown to be as important for overall performance as the selection of training objectives (Liu et al., 2019). We find that BART exhibits the most consistently strong performance across the full range of tasks we consider.

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.

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

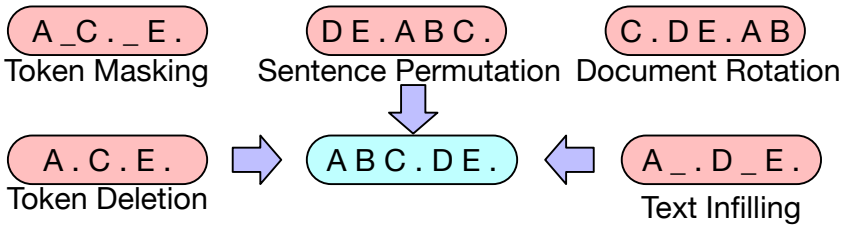


Figure 2: Transformations for noising the input that we experiment with. These transformations can be composed.

decide which positions are missing inputs.

Text Infilling A number of text spans are sampled, with span lengths drawn from a Poisson distribution ($\lambda = 3$). Each span is replaced with a *single* `[MASK]` token. 0-length spans correspond to the insertion of `[MASK]` tokens. Text infilling is inspired by SpanBERT (Joshi et al., 2019), but SpanBERT samples span lengths from a different (clamped geometric) distribution, and replaces each span with a sequence of `[MASK]` tokens of exactly the same length. Text infilling teaches the model to predict how many tokens are missing from a span.

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

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

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.

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.

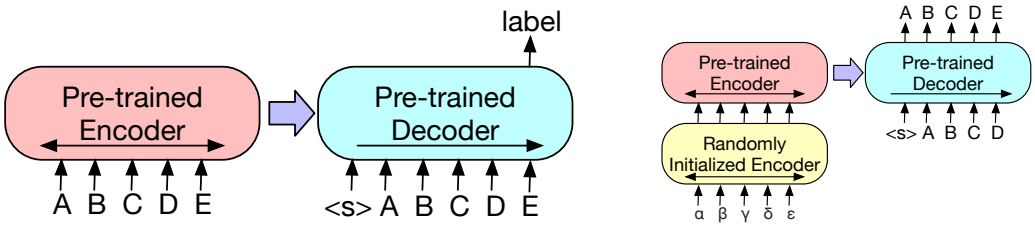
We train the source encoder in two steps, in both cases backpropagating the cross-entropy loss from the output of the BART model. In the first step, we freeze most of BART parameters and only update the randomly initialized source encoder, the BART positional embeddings, and the self-attention input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

4 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 differ-



(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.

(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.

Figure 3: Fine tuning BART for classification and translation.

ences 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:

Language Model Similarly to GPT (Radford et al., 2018), we train a left-to-right Transformer language model. This model is equivalent to the BART decoder, without cross-attention.

Permuted Language Model Based on XLNet (Yang et al., 2019), we sample 1/6 of the tokens, and generate them in a random order autoregressively. For consistency with other models, we do not implement the relative positional embeddings or attention across segments from XLNet.

Masked Language Model Following BERT (Devlin et al., 2019), we replace 15% of tokens with [MASK] symbols, and train the model to independently predict the original tokens.

Multitask Masked Language Model As in UniLM (Dong et al., 2019), we train a Masked Language Model with additional self-attention masks. Self attention masks are chosen randomly with the follow proportions: 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.

Masked Seq-to-Seq Inspired by MASS (Song et al., 2019), we mask a span containing 50% of tokens, and train a sequence to sequence model to predict the masked tokens.

For the Permuted LM, Masked LM and Multitask Masked LM, we use two-stream attention (Yang et al., 2019) to efficiently compute likelihoods of the output part of the sequence (using a diagonal self-attention mask on the output to predict words left-to-right).

We experiment with (1) treating the task as a standard sequence-to-sequence problem, where the source input to the encoder and the target is the decoder output, or (2) adding the source as prefix to the target in the decoder, with a loss only on the target part of the sequence. We find the former works better for BART models, and the latter for other models.

To most directly compare our models on their ability to model their fine-tuning objective (the log likelihood of the human text), we report perplexity in Table 1.

4.2 Tasks

SQuAD (Rajpurkar et al., 2016) an extractive question answering task on Wikipedia paragraphs. Answers are text spans extracted from a given document context. Similar to BERT (Devlin et al., 2019), we use concatenated question and context as input to the encoder of BART, and additionally pass them to the decoder. The model includes classifiers to predict the start and end indices of each token.

MNLI (Williams et al., 2017), a bitext classification task to predict whether one sentence entails another. The fine-tuned model concatenates the two sentences with appended an EOS token, and passes them to both the BART encoder and decoder. In contrast to BERT, the representation of the EOS token is used to classify the sentences relations.

ELI5 (Fan et al., 2019), a long-form abstractive question answering dataset. Models generate answers conditioned on the concatenation of a question and supporting documents.

XSum (Narayan et al., 2018), a news summarization dataset with highly abstractive summaries.

ConvAI2 (Dinan et al., 2019), a dialogue response generation task, conditioned on context and a persona.

CNN/DM (Hermann et al., 2015), a news summarization dataset. Summaries here are typically closely related to source sentences.

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**

Table 1: Comparison of pre-training objectives, including approaches inspired by BERT, MASS, GPT, XLNet and UniLM. All models are a similar size to BERT Base and are trained for 1M steps on the same 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.

4.3 Results

Results are shown in Table 1. Several trends are clear:

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.

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.

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.

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.

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

	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.

	SQuAD 1.1 EM/F1	SQuAD 2.0 EM/F1
BERT	84.1/90.9	79.0/81.8
UniLM	-/-	80.5/83.4
XLNet	**89.0**/94.5	86.1/88.8
RoBERTa	88.9/**94.6**	**86.5/89.4**
BART	88.8/**94.6**	86.1/89.2

Table 3: BART gives similar results to XLNet and RoBERTa on question answering.

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.

5.2 Discriminative Tasks

Tables 3 and 2 compares the performance of BART with several recent approaches on the well-studied SQuAD and GLUE tasks (Warstadt et al., 2018; Socher et al., 2013; Dolan & Brockett, 2005; Agirre et al., 2007; Williams et al., 2017; Dagan et al., 2006; Levesque et al., 2011).

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. suggesting that BART's improvements on generation tasks do not come at the expense of classification performance.

5.3 Generation Tasks

We also experiment with several text generation tasks. BART is fine-tuned as a standard sequence-to-sequence model from the input to the output text. During fine-tuning we use a label smoothed cross entropy loss (Pereyra et al., 2017), with the smoothing parameter set to 0.1. During generation, we set beam size as 5, remove duplicated trigrams in beam search, and tuned the model with min-len, max-len, length penalty on the validation set (Fan et al., 2017).

Summarization To provide a comparison with the state-of-the-art in summarization, we present results on two summarization datasets, CNN/DailyMail and XSum, which have distinct properties (Table 4).

Summaries in the CNN/DailyMail tend to resemble source sentences. Extractive models do well here, and even the baseline of the first-three source sentences is highly competitive. Nevertheless, BART outperforms all existing work.

In contrast, XSum is highly abstractive, and extractive models perform poorly. BART outperforms the best previous work, based on RoBERTa, by roughly 3.5 points on all ROUGE metrics—representing a significant advance in performance on this problem. Qualitatively, sample quality is high (see §6).

We also conduct human evaluation (Table 5). Annotators were asked to choose the better of two summaries for a passage. One summary was from BART, and the other was either a human reference or publicly available output from the BERTSUMEXTABS model. As with automated metrics, BART significantly outperforms prior work. However, it has not reach human performance on this task.

Dialogue We evaluate dialogue response generation on CONVAI2 (Dinan et al., 2019), in which agents must generate responses conditioned on both the previous context and a textually-specified persona. BART outperforms previous work on two automated metrics.

Abstractive QA We use the recently proposed ELI5 dataset to test the model's ability to generate long free-form answers. We find BART outperforms the best previous work by 1.2 ROUGE-L, but the dataset remains a challenging, because answers are only weakly specified by the question.

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.

	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.

	XSum	
	Prefer BART	Prefer Baseline
vs. BERTSUMEXTABS	73%	27%
vs. Reference	33%	67%

Table 5: Human Evaluation on XSum: BART summaries are preferred to those from previous work, but not to human-written reference summaries.

	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).

	RO-EN
Baseline	36.80
Fixed BART	36.29
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.

Experiment results are presented in Table 8. 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.

6 Qualitative Analysis

BART shows large improvements on summarization metrics, of up to 3.5 points over the prior state-of-the-art. To understand BART's performance beyond automated metrics, we analyse its generations qualitatively.

Table 9 shows representative example summaries generated by BART, illustrating its main strengths and weaknesses. 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, outputs are also highly abstractive, with few copied phrases. Summaries are generally factually accurate, and integrate 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 some effects of global warming requires non-trivial inference. However, the claim that the work was published in Science is not supported by the source—

Source Document (abbreviated)	BART Summary
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.	Fisheries off the coast of Fiji are protecting coral reefs from the effects of global warming, according to a study in the journal Science.
Sacoolas, who has immunity as a diplomat's wife, was involved in a traffic collision ... Prime Minister Johnson was questioned about the case while speaking to the press at a hospital in Watford. He said, "I hope that Anne Sacoolas will come back ... if we can't resolve it then of course I will be raising it myself personally with the White House."	Boris Johnson has said he will raise the issue of US diplomat Anne Sacoolas' diplomatic immunity with the White House.
According to Syrian state media, government forces began deploying into previously SDF controlled territory yesterday. ... On October 6, US President Donald Trump and Turkish President Recep Tayyip Erdoan spoke on the phone. Then both nations issued statements speaking of an imminent incursion into northeast Syria ... . On Wednesday, Turkey began a military offensive with airstrikes followed by a ground invasion.	Syrian government forces have entered territory held by the US-backed Syrian Democratic Forces (SDF) in response to Turkey's incursion into the region.
This is the first time anyone has been recorded to run a full marathon of 42.195 kilometers (approximately 26 miles) under this pursued landmark time. It was not, however, an officially sanctioned world record, as it was not an "open race" of the IAAF. His time was 1 hour 59 minutes 40.2 seconds. Kipchoge ran in Vienna, Austria. It was an event specifically designed to help Kipchoge break the two hour barrier.	Kenyan runner Eliud Kipchoge has run a marathon in less than two hours.
PG&E stated it scheduled the blackouts in response to forecasts for high winds amid dry conditions. The aim is to reduce the risk of wildfires. Nearly 800 thousand customers were scheduled to be affected by the shutoffs which were expected to last through at least midday tomorrow.	Power has been turned off to millions of customers in California as part of a power shutoff plan.

Table 9: Example summaries from the XSum-tuned BART model on WikiNews articles. For clarity, only relevant excerpts of the source are shown. Summaries combine information from across the article and prior knowledge.

and, in general, the main limitation of the model is a tendency to hallucinate unsupported information.

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

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. BART differs in masking more but shorter spans from the input, and in always predicting the complete output. Table 1 shows that in a controlled comparison, BART's pre-training objective outperforms MASS on five out of six tasks.

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.

Concurrently, Raffel et al. (2019) pre-trained a denoising sequence-to-sequence model named T5, experimenting with a similar range of noising tasks. BART uses a slightly different objective, in which spans are masked from the input but the complete output is predicted, to improve the decoder's language modelling ability. BART achieves higher performance with similar model sizes, particularly on summarization. T5 demonstrates that by scaling to very large models sizes, denoising sequence-to-sequence pre-training can achieve new state-of-the-art results on many tasks.

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 pre-training 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.

8 Conclusions

We introduced BART, a pre-training approach that learns to map corrupted documents to the original. BART performs comparably to RoBERTa on discriminative tasks, and achieves new state-of-the-art results on several text generation tasks. Future work should explore new methods for corrupting documents for pre-training, perhaps tailoring them to specific end tasks.

References

Eneko Agirre, Llu'is M'arquez, and Richard Wicentowski (eds.). *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*. Association for Computational Linguistics, Prague, Czech Republic, June 2007.

Ido Dagan, Oren Glickman, and Bernardo Magnini. The PASCAL recognising textual entailment challenge. In *Machine learning challenges. evaluating predictive uncertainty, visual object classification, and recognising tectual entailment*, pp. 177–190. Springer, 2006.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL https://www.aclweb.org/anthology/N19-1423.

Emily Dinan, Varvara Logacheva, Valentin Malykh, Alexander Miller, Kurt Shuster, Jack Urbanek, Douwe Kiela, Arthur Szlam, Iulian Serban, Ryan Lowe, et al. The second conversational intelligence challenge (convai2). *arXiv preprint arXiv:1902.00098*, 2019.

William B Dolan and Chris Brockett. Automatically constructing a corpus of sentential paraphrases. In *Proceedings of the International Workshop on Paraphrasing*, 2005.

Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. Unified language model pre-training for natural language understanding and generation. *arXiv preprint arXiv:1905.03197*, 2019.

Sergey Edunov, Alexei Baevski, and Michael Auli. Pre-trained language model representations for language generation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019.

Angela Fan, David Grangier, and Michael Auli. Controllable abstractive summarization. *arXiv preprint arXiv:1711.05217*, 2017.

Angela Fan, Yacine Jernite, Ethan Perez, David Grangier, Jason Weston, and Michael Auli. Eli5: Long form question answering. *arXiv preprint arXiv:1907.09190*, 2019.

Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. Teaching machines to read and comprehend. In *Advances in neural information processing systems*, pp. 1693–1701, 2015.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S Weld, Luke Zettlemoyer, and Omer Levy. Spanbert: Improving pre-training by representing and predicting spans. *arXiv preprint arXiv:1907.10529*, 2019.

Guillaume Lample and Alexis Conneau. Cross-lingual language model pretraining. *arXiv preprint arXiv:1901.07291*, 2019.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*, 2019.

Hector J Levesque, Ernest Davis, and Leora Morgenstern. The Winograd schema challenge. In *AAAI Spring Symposium: Logical Formalizations of Commonsense Reasoning*, volume 46, pp. 47, 2011.

Yang Liu and Mirella Lapata. Text summarization with pretrained encoders. *arXiv preprint arXiv:1908.08345*, 2019.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

Shashi Narayan, Shay B Cohen, and Mirella Lapata. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. *arXiv preprint arXiv:1808.08745*, 2018.

Gabriel Pereyra, George Tucker, Jan Chorowski, Łukasz Kaiser, and Geoffrey Hinton. Regularizing neural networks by penalizing confident output distributions. *arXiv preprint arXiv:1701.06548*, 2017.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*, 2018.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. *URL https://s3-us-west-2. amazonaws. com/openai-assets/researchcovers/languageunsupervised/language understanding paper. pdf*, 2018.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8), 2019.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*, 2019.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*, 2016.

Sascha Rothe, Shashi Narayan, and Aliaksei Severyn. Leveraging pre-trained checkpoints for sequence generation tasks. *arXiv preprint arXiv:1907.12461*, 2019.

Abigail See, Peter J Liu, and Christopher D Manning. Get to the point: Summarization with pointer-generator networks. *arXiv preprint arXiv:1704.04368*, 2017.

Rico Sennrich, Barry Haddow, and Alexandra Birch. Edinburgh neural machine translation systems for WMT 16. In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, 2016.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of EMNLP*, pp. 1631–1642, 2013.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. Mass: Masked sequence to sequence pre-training for language generation. In *International Conference on Machine Learning*, 2019.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pp. 5998–6008, 2017.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*, 2018.

Alex Warstadt, Amanpreet Singh, and Samuel R. Bowman. Neural network acceptability judgments. *arXiv preprint 1805.12471*, 2018.

Adina Williams, Nikita Nangia, and Samuel R Bowman. A broad-coverage challenge corpus for sentence understanding through inference. *arXiv preprint arXiv:1704.05426*, 2017.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V Le. Xlnet: Generalized autoregressive pretraining for language understanding. *arXiv preprint arXiv:1906.08237*, 2019.