Abstractive Dialogue Summarization using Fine-tuned BART on the SAMSum Dataset

GitHub:

Colab: https://colab.research.google.com/drive/19O7t7BA43GT7ywKrXq9-koK5-rVnqAFl?usp=sharing

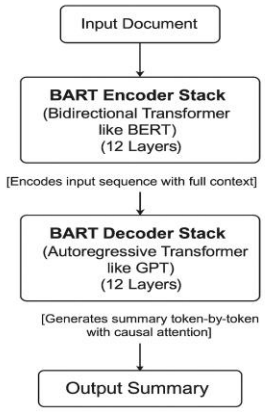
Introduction

Recent advances in natural language processing have popularized transformer architectures such as BERT, GPT, and BART, resulting in remarkably fluent text generation (Lewis et al., 2020; Wolf et al., 2020). Among the many downstream tasks, summarizing conversations remains especially difficult because transcripts often contain slang, frequent speaker switches, and abrupt topic shifts. To address these issues, we fine-tune the Facebook/Bart-base model on the SAMSum corpus, a benchmark set of everyday dialogues (Gliwa et al., 2019). We measure performance gain with multiple automatic metrics and present illustrative cases, showing that tailoring a model to a specific dialogue domain can yield clearer, more coherent summaries in practice.

Methodology

1. Model Architecture and Summarization Approach

This research adopts a modified BART framework that merges BERT-style bidirectional noise-robust encoding with the left-to-right, token-generation characteristic seen in GPT (Lewis et al., 2020). During bulk pre-training, BART learns through denoising tasks, including token infilling and random sentence reordering. When applied to summarization, the encoder ingests the whole conversation, while the decoder diligently crafts a condensed output, attending throughout to the dense set of encoded features.



The first experiment evaluated the off-the-shelf, pre-trained BART on the SAMSum corpus without any task-specific fine-tuning. Then, in a supervised phase, the model was trained end-to-end on paired dialogue-summary examples. Although the backbone remained unchanged, adjusting the weights to conversational patterns enriched fluency, cohesion, and content retention in the generated abstracts.

2. Data Preparation and Preprocessing

The SAMSum set contains more than fifteen thousand light, chat-style exchanges between multiple speakers, each matched with a human-crafted abstractive summary (Gliwa et al., 2019). Its informal lexicon, frequent emojis, and realistic interruptions mirror everyday texting, making it an ideal benchmark for learning dialogue-aware summarization methods.

The original dataset was divided into three parts: training, validation, and test sets. A cleaning pipeline eliminated extraneous whitespace, fixed malformed entries, and standardized special characters. Tokenization was handled with BART's dedicated tokenizer (Lewis et al., 2020), truncating input sequences to a maximum of 512 tokens and summaries to 128. During training, padding tokens were masked to prevent them from skewing the loss calculation.

3. Fine Tuning Setup and Training Parameters

Fine-tuning was performed using the following hyperparameters:

These settings were chosen to balance computational efficiency,

training stability, and model performance, especially given

GPU memory constraints.

Parameter	Value
Batch Size	4
Learning Rate	2e-5
Epochs	3
Weight Decay	0.01
Warmup Steps	100
Gradient Accumulation	2
Generation Max Length	128
Beam Search Width	4

4. Evaluation Metrics and Inference

To assess performance, we used a diverse set of evaluation metrics:

Metric	Description
ROUGE (1, 2, L, Lsum)	Measures lexical overlap between generated and reference summaries.
BLEU	Evaluates n-gram precision and fluency.
METEOR	Considers synonym matches and stemming.
BERTScore	Uses contextual embeddings to assess semantic similarity.
Perplexity	Indicates the model's fluency and uncertainty in generation.

5. Model Size, Parameters, and Practical Benefits

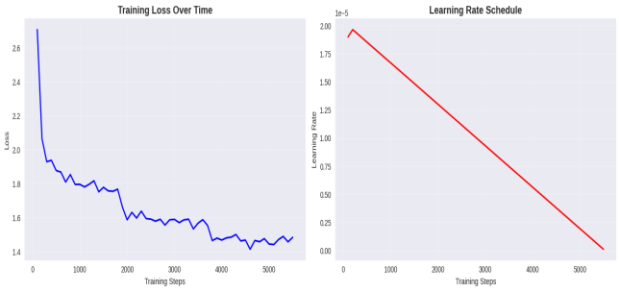
The BART-base architecture examined in both the baseline runs and the fine-tuning phase boasts roughly 139 million parameters (Lewis et al., 2020). Although it is smaller than the BART-large variant, the base configuration strikes a useful compromise between speed and overall accuracy in many tasks. Because the fine-tuning procedure adjusts every parameter, it calls for a moderate amount of GPU memory yet typically produces a disproportionate lift in downstream performance.

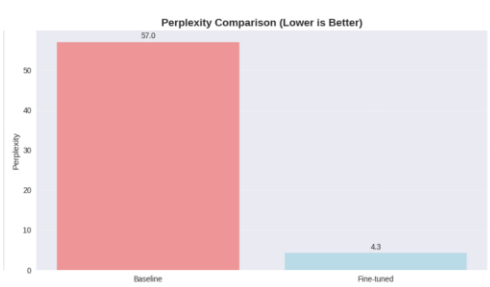
In practice, the fine-tuned version is robust enough for daily settings such as automatic summarization of customer-support chats, generation of meeting minutes, or high-level abstractions of legal discussions. Gains in semantic coherence, content coverage, and fluency more than justify the effort and computing spent on tuning, particularly when the model will serve a specific, recurring task.

Results and Evaluation

The table below presents the comparative results between the baseline and fine-tuned models:

Metric	Baseline	Fine-tuned	Improvement
ROUGE-1	26.75	49.06	+22.31
ROUGE-2	8.25	24.03	+15.78
ROUGE-L	19.22	40.95	+21.73
ROUGE-Lsum	19.24	40.99	+21.75
BLEU	2.77	17.15	+14.38
METEOR	31.30	41.55	+10.25
BERTScore-F1	85.22	91.57	+6.35
Perplexity	57.01	4.29	−52.7





The results demonstrate marked gains on every measured dimension. Fine-tuning enhanced the model's fluency, factual accuracy, and semantic alignment with human-written summaries.

References

- Lewis, M., Liu, Y., Goyal, N., et al. (2020). *BART: Denoising SequencetoSequence Pretraining for Natural Language Generation, Translation, and Comprehension*. ACL.
- Gliwa, B., Mochol, I., Biesek, M., & Wawer, A. (2019). *SAMSum Corpus: A Humanannotated Dialogue Dataset for Abstractive Summarization*. arXiv:1911.12237.
- Wolf, T., et al. (2020). *Transformers: Stateoftheart Natural Language Processing*. EMNLP: System Demonstrations.