FLAN-T5 Fine-Tuning for Text Summarization

Michael Buloichyk

April 13, 2024

1 Introduction

This report outlines the process undertaken to fine-tune a FLAN-T5 model for text summarization, conducted as part of a hiring task. The objective was to demonstrate rigor and proficiency in adapting a small language model to data summarization.

1.1 Tasks

- Fine-tune a small language model from Hugging Face such as GPT-2, on a CPU environment.
- Design and execute a rigorous experimental setup for model evaluation.
- Utilize a provided dataset tailored for summarization tasks to train and evaluate the model.

2 Tools and Models

Before detailing the data preprocessing steps, it is crucial to outline the tools and models that were instrumental in this fine-tuning task.

2.1 Model Selection

Initially, the GPT-2 model was considered for this text summarization task. However, it was later decided to switch to the FLAN-T5 models for several reasons:

- **Model Size:** The GPT-2 model proved to be computationally expensive to train efficiently on available CPU resources and even on GPU. This constraint necessitated a shift to a more manageable model in terms of computational requirements.
- **Model Architecture:** GPT-2, being an autoregressive decoder-only model, posed limitations for summarization tasks that benefit from bidirectional context understanding. FLAN-T5, with its encoder-decoder architecture, offers better functionality for generating concise summaries by considering the full context of the input text.
- **Data Size:** GPT-2, as an autoregressive, decoder-only model without instruction-based pre-training, struggles with summarization tasks on small datasets. What was revealed during initial Fine Tune imlementation.

2.2 Adopted Models and Tools

Following the switch, the following models and tools were adopted:

- **FLAN-T5-small and FLAN-T5-base:** These models were chosen for their suitability in handling the intricacies of text summarization. FLAN-T5-small was primarily used for experimentation due to its lower computational demands, while FLAN-T5-base was considered for scenarios where more complex modeling was feasible.
- **Transformers Module:** The experiment used the Transformers library by Hugging Face.

2.3 Evaluation Metrics

For evaluating the performance of the fine-tuned models, the ROUGE metric was used. ROUGE is crucial for summarization tasks as it measures the overlap of n-grams between the generated summaries and the reference summaries, providing a quantitative measure of the model's summarization quality.

3 Dataset and Preprocessing

The dataset used for this experiment is designed to test the summarization capabilities of a fine-tuned language model, containing converted text from website and PDF content. The dataset is small, with 173 data input rows.

3.1 Data cleaning and preprocessing

The initial step in data preprocessing involved thorough cleaning to ensure the quality and utility of the dataset:

- **Filtering for Duplicates:** All duplicate entries were removed from the dataset to prevent the model from being biased towards repeated text.
- **Filtering for Emptiness:** Entries that were empty or contained insufficient textual content were filtered out.

After cleaning, the dataset was converted into the `DatasetDict` format used by the Hugging Face Transformers library.

3.2 Analysis of the data

The preprocessing phase of the data revealed that a significant portion of the input needed to be refined:

- Empty rows filtered: 16
- Duplicate rows removed: 31
- Inputs below 1024 tokens: 8%
- Inputs above 1024 tokens: 92%

3.3 Input Token Lengths

An analysis of token lengths within the inputs highlighted challenges for small model capacities:

- Mean token length: 4005.9, far exceeding the GPT-2 and T5 models' max sequence length.
- The range of token lengths spans from a minimum of 363 to a maximum of 8527.
- The 25th percentile sits at 1633, and the 75th percentile at 7794 tokens.

Figure 1 visualizes the distibution of the tokens length accross the dataset

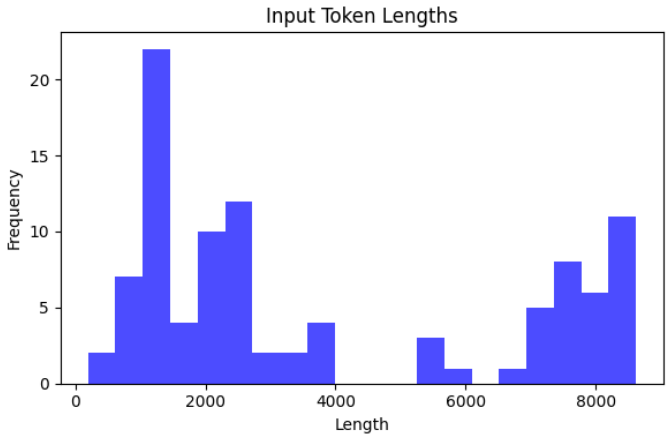

Figure 1: Distribution of token lengths in the dataset.

Table 1: Input Token Lengths Summary

Measure	Value
Count	100
Mean	4005.1
Standard Deviation	2913.7
Minimum	197
25% Percentile	1400.25
Median	2615.5
75% Percentile	7386
Maximum	8617

The input data exceeds the `max_token` parameter for both GPT-2 and T-5 models, necessitating a dynamic input truncation strategy to optimize text summarization performance. This approach must prioritize retaining critical content for effective model processing.

4 Tokenization and Truncation

Given the constraints posed by the model's maximum token length and the substantial size of the input documents, it was necessary to implement truncation strategy:

4.1 Straightforward truncation

- In this approach, inputs were truncated from the right side during tokenization to conform to the model's token maximum limit, without additional processing.
- The prompt instructions were added to the dataset.
- Outputs were truncated by providing a `max_length` argument of 512 to the tokenizer, ensuring the model's maximum token limit was not exceeded.

4.2 Relevance truncation

Relevance truncation: To improve the quality of training and output, a relevance truncation approach was adopted. Utilizing Term Frequency-Inverse Document Frequency (TF-IDF) analysis, this method identifies and retains text segments most relevant to the target summary, ensuring that crucial information is preserved. This method employs the following process:

1. **Segmentation of Input Text:** The input text is segmented into a set of sentences using NLTK's punkt tokenizer.
2. **Calculation of Relevance Scores:** TF-IDF analysis is applied to each sentence against the summary to determine its relevance. These scores are used to select sentences that are most relevant.
3. **TF-IDF Methodology:** TF-IDF evaluates the importance of a word to a document within a collection of documents. It combines term frequency (TF) within a given document and the inverse document frequency (IDF) of the term across the entire document set. The formula is given by:
$$\text{TF-IDF}_{\text{word,doc}} = \text{TF}_{\text{word,doc}} \times \log\left(\frac{N}{\text{IDF}_{\text{word}}}\right) \tag{1}$$
Where N is a number of documents.
4. **Matrix Representation:** The relevance of each sentence to the summary is calculated, resulting in a TF-IDF matrix that represents term weights across sentences and the summary. The matrix formulates as:
$$\begin{bmatrix} \text{TF-IDF}_{\text{sentence1, word1}} & \cdots & \text{TF-IDF}_{\text{sentence1, wordN}} \\ \vdots & \ddots & \vdots \\ \text{TF-IDF}_{\text{sentenceN, word1}} & \cdots & \text{TF-IDF}_{\text{sentenceN, wordN}} \\ \text{TF-IDF}_{\text{summary, word1}} & \cdots & \text{TF-IDF}_{\text{summary, wordN}} \end{bmatrix}$$
5. **Cosine Similarity:** The cosine similarity is computed to measure the relevance of sentences to the ideal summary, which is mathematically represented as:
$$\text{Cosine Similarity} = \frac{\vec{A} \cdot \vec{B}}{\|\vec{A}\|\|\vec{B}\|} \tag{2}$$
where $\vec{A}$ and $\vec{B}$ are vector representations of sentences and the summary, respectively.

Upon computing the relevance scores using the TF-IDF method and cosine similarity, the next step is to select the most pertinent sentences to the summary. The approach taken is detailed as follows:

6. **Identification of Top Relevance Sentences:** Sentences are ranked based on their relevance scores, and the ones with the highest scores are identified.
7. **Filtering by Relevance Score:** The dataset is filtered to include only the sentences with the highest relevance.
8. **Top-K Selection Strategy:** A top-k strategy is employed where 'k' represents the number of most relevant sentences to retain.

The identification and selection process ensures that the training data fed into the model consists of sentences that are crucial for generating accurate summaries, thus enhancing the model's learning and summarization capabilities.

4.3 Computational Resources

To efficiently manage the computational demands of training FLAN-T5 models, the experiment leveraged external resources:

- **Lightning AI Cloud Platform:** The Lightning AI cloud platform enabled scalable computational resources, facilitating model deployment and management in a cloud environment.
- **L4 GPU:** NVIDIA L4 GPU, was utilized for training for time-saving purposes.

5 Training

5.1 Evaluation Setup

For model evaluation, we employ ROUGE-2 metrics to assess bigram overlap between generated and reference summaries. This metric is particularly effective in capturing phrase and sentence-level coherence, essential for evaluating summarization quality.

5.2 Fine-Tuning FLAN-T5-small

- Fine-tuning initiated with setting up Weights & Biases for hyperparameter sweeps and training monitoring.

Following experimentation with multiple hyperparameters sweeps, the optimal hyperparameters for fine-tuning the FLAN-T5-small model were established as follows:

The following table 3 provides a summary of the training results observed across several epochs, illustrating improvements and stabilizations in model performance:

The training process and its outcomes are comprehensively visualized on the Weights & Biases platform.Detailed graphs and metrics can be viewed by following the link: Training Visualization on Weights & Biases.

Parameter	Value
Learning Rate	$1 \cdot 10^{-5}$
Per Device Train Batch Size	1
Per Device Eval Batch Size	1
Weight Decay	0.01
Save Total Limit	2
Number of Training Epochs	5
Gradient Accumulation Steps	3

Table 2: Optimal Hyperparameters for FLAN-T5-small Fine Tuning

Epoch	Training Loss	Validation Loss	Rouge1	Rouge2	Rougel	Rougelsum
0	1.9528	1.9737	0.0706	0.0439	0.0643	0.0669
1	1.9576	1.9708	0.0713	0.0451	0.0656	0.0674
3	1.9935	1.9614	0.0739	0.0509	0.0698	0.0706
4	1.9764	1.9600	0.0739	0.0509	0.0698	0.0706

Table 3: Training and Validation Performance

5.3 Fine-Tuning FLAN-T5-base with LORA

- Fine-tuning applied on FLAN-T5-Base with 248M parameters using LORA.
- Learning rate decay was scheduled.
- Loss showed a significant reduction, pointing to stabilization in learning.
- Training metrics were monitored using Weights & Biases

Parameter	Value
r	32
lora_alpha	32
target_modules	[”q”, ”v”]
lora_dropout	0.05
bias	None
task_type	SEQ_2_SEQ_LM

Table 4: LORA Configuration for FLAN-T5-Base Model

This adaptation reduced the number of trainable parameters to 3,538,944, significantly decreasing the computational resources required for effective fine-tuning. For the input data preparation, smart truncation techniques were applied as described in Chapter 4.2, ensuring the retention of relevant content for model training.

6 Evaluation

Model Summarization Comparison

Included Jupyter notebook, benchmarks three distinct models: the Pre-trained, Fine-tuned FLAN-T5-small, and PEFT. A quantitative analysis of summarization performance is performed, which

is essential for understanding the effectiveness of fine-tuning and PEFT adaptation.

Pre-Trained Model Summary:

"5G is the fastest way to unlock industry 4.0. It's the best way for the world to use IoT. The world' s fastest and most efficient IOT devices are able to be used in the industrial world. And it'll be the most reliable and fastest IOC device ever.... 5 G is needed for high data rate transmissions and ultra-low latency. But as the industry continues to develop, an entirely new level of speed, efficiency, and performance will help create new use cases for IOOT, which is a major factor in industrial applications. 5g will be necessary for higher data rates and low latence.' The 'Internet of Things' is poised to impact virtually every industry, but none more so than manufacturing. In fact, 5GS is improving five main areas of Industry 5.0 supply chain management, product quality, predictive maintenance, production efficiency and digital twins. As in other sectors, I OCT and Industrial IOT related to manufacturing are made possible by wireless connectivity and the technologies that support it. Today, the Industrial Allied Ions are making possible."

Fine-Tuned FLAN-T5-Small Model Summary

"This document is an article titled 'Taming Network Complexity with Digital Twins' by Informa Network Computing. It discusses the adoption of digital twin technology in enterprises, telecom companies, and cloud providers. The article discusses the role of virtual replicas in network configuration and state. Key points from the article include: - Digital twins can help manage and troubleshoot complex networks by providing an accurate replica of the networks configuration, state, etc. – Digital networks are increasingly becoming increasingly demanding and demanding than ever before, demanding faster speeds, higher availability, quick resolution of problems, or no major outages. In conclusion, the document highlights the importance of integrating digital Twin technology into networks and introducing new technologies and expanding networks. Overall, it highlights how digital networks can be used for network management and control, enabling the use of data and data to manage complex network networks, improve network performance, reduce disruption, optimize network infrastructure, enhance network connectivity, provide a reliable and reliable service, enable unified network service management, protect network integrity, mitigate network disruptions, leverage network resources, create scalable network solutions, build detailed digital versions of networks for customers, increase network efficiency, accelerate network maintenance, streamline network operations, simplify network administration, strengthen network security, promote network resilience, develop network systems, implement network-wide network deployments and improve networks' performance."

Finally, the PEFT model's summary for the same prompt.

PEFT Model Summary

"This document is an article titled 'Taming Network Complexity with Digital Twins' by Informa Network Computing. It discusses the use of digital twins in managing and troubleshooting complex networks. The article highlights the importance of virtual replicas of physical systems, such as jet engines or machines, to managing network complexity. Key points from the article include: - Digital twin technology is becoming more popular in the context of manufacturing, where virtual replicates mimic physical system workings. – Digital replica technology can help manage and troubleshoot network complexities by providing an accurate replica of the networks configuration and state. This article emphasizes that digital replica technologies are becoming increasingly popular, and there is a growing interest in digital Twin technology. In conclusion, the document highlights how digital hybrids can be used to manage network problems and improve network performance. However, it emphasizes that the benefits of using digital replicated data are still largely unknown. Overall, this article provides an overview of Digital Twins, their applications, challenges, applications, and their potential applications in network management and maintenance."

The following plot, shown in Figure 2, represents the ROUGE-2 scores of all three models for 5 prompt-summary pairs. It is observed that the utilization of PEFT with the truncation strategy led to a 12% improvement in the performance of the FLAN-T5-small model.

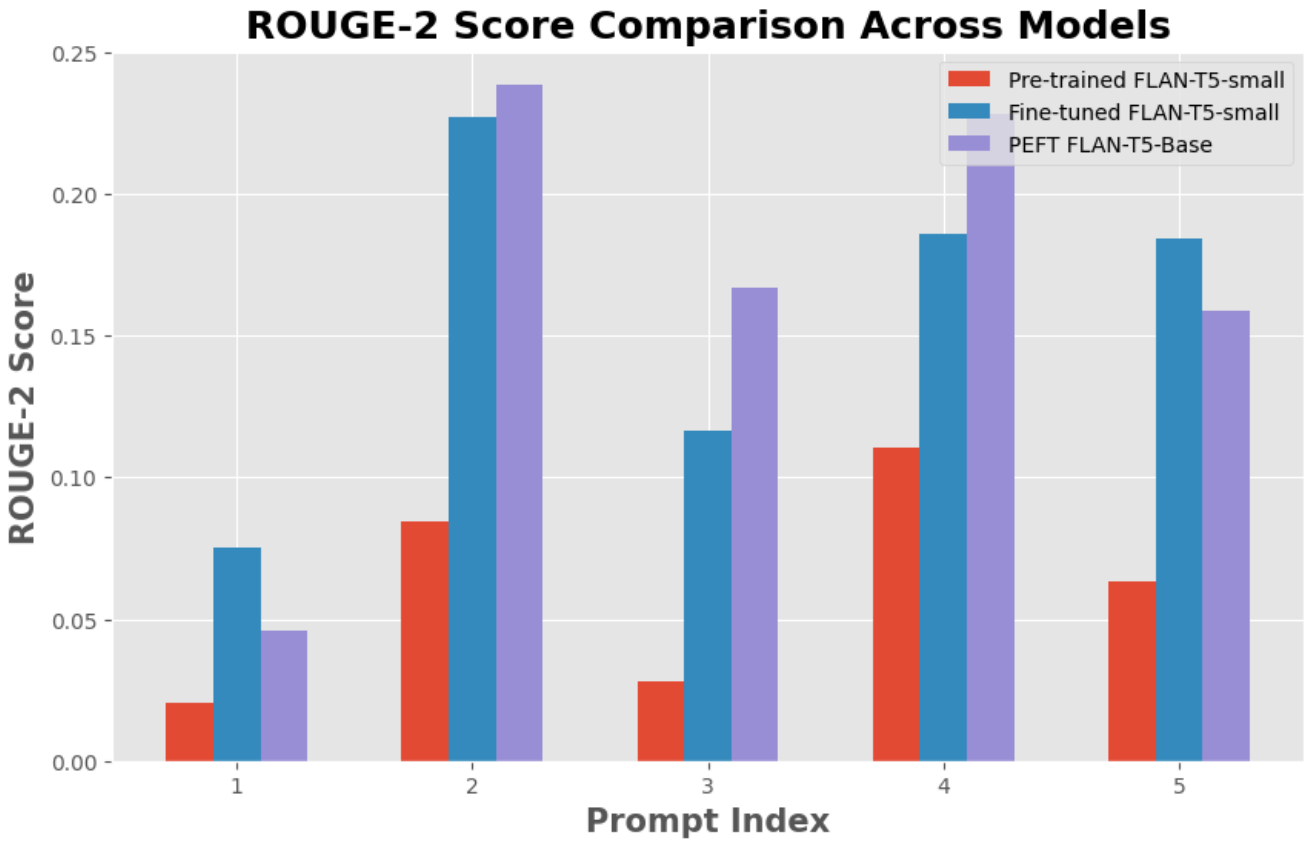

Figure 2: ROUGE 2 Score comparison across models

7 Conclusion

This report undertook the fine-tuning of large language models (LLMs) for text summarization on a specified dataset. I quantitatively evaluated the outputs of various models, including pre-trained, fine-tuned small, and PEFT-enhanced versions. Adaptive strategies such as dynamic input truncation and PEFT were implemented, significantly enhancing model performance and efficiency in summarization tasks. We discovered that the PEFT utilization provided 12% improvement over the fine tuned FLAN-T5-small model with respect to ROUGE2 metric.

Further improvements

For future enhancements, consider implementing Reinforcement Learning from Human Feedback (RLHF) and Proximal Policy Optimization (PPO) to refine outputs. Enhancing prompt engineering could improve responsiveness. For more nuanced metrics, explore alternative evaluation strategies like https://arxiv.org/abs/2112.01589, https://huggingface.co/spaces/evaluate-metric/bleu, https://huggingface.co/spaces/evaluate-metric/meteor, https://huggingface.co/spaces/evaluate-metric/bertsc and using other LLMs as judges in the DeepEval framework.

Augmenting the dataset could yield better results. Additionally, exploring truncation strategies, such as segmenting input data and adjusting labels accordingly, might be beneficial but must be approached carefully to maintain data integrity.

A Appendix

A.1 GitHub Repository

The code and data for this project, including the Jupyter notebooks for training and evaluation, are available in the following GitHub repository:

`https://github.com/Michailbul/FLAN-T5-small-base-FT`