Text Summarization using Pretrained Large Language Models

DSCI 6004 – Section 3: Natural Language Processing, Spring 2024

Final Project Report

Submitted by

Godswill Ikwan gikwa1@unh.newhaven.edu

Sumanth Ganimalla sgani4@unh.newhaven.edu

Contents

Abstract

Text summarization is a crucial task in natural language processing (NLP), with applications in various domains such as news summarization, document summarization, and scientific article summarization. In this project, we explore the effectiveness of pretrained large language models, specifically Pegasus and BART architectures, for text summarization. We fine-tune these models on two different datasets, SAMSUM and SCiTLDR, for abstractive text summarization, and evaluate their performance using the ROUGE metric. Our results show promising performance, with both models demonstrating competitive performance on the SAMSUM dataset. BART and Pegasus perform similarly on the SCiTLDR dataset, but both perform best when SCiTLDR dataset was used indicating that the performance of these models varies depending on the dataset.

Introduction

Text summarization, the task of condensing large bodies of text into shorter, coherent representations while preserving the essential meaning, has been a fundamental challenge in natural language processing (NLP). Over the years, research in this field has evolved from early extractive methods to more advanced abstractive approaches, driven by the increasing demand for automated summarization systems in various applications such as information retrieval, document analysis, and content generation.

Text summarization research has a long history, initially focusing on extractive methods that selected and rearranged sentences or passages from the original text. The Luhn Algorithm[1], proposed in 1958, was an early notable work in this area, identifying key sentences based on word frequency and position.

In the 1990s and early 2000s, research shifted to more sophisticated techniques, such as graph-based models and machine learning approaches. The TextRank algorithm[2], introduced in 2004, enabled the extraction of salient sentences based on graph centrality measures.

The advent of deep learning in the late 2000s and 2010s led to the development of abstractive methods, where summaries are generated by understanding and paraphrasing the original text. Neural network architectures like sequence-to-sequence models [3], introduced by Sutskever et al. in 2014, and attention mechanisms [4], introduced by Bahdanau et al. in 2015, have played a significant role in this evolution.

This project aims to explore the effectiveness of fine-tuning pre-trained models like PEGASUS [5] and BART [6] for text summarization tasks. We evaluate their performance on small subsets of the SAMSUM and SciTLDR datasets, aiming to assess their suitability for generating high-quality summaries in specific domains. Through our experiments, we aim to contribute to the broader understanding of text summarization techniques and their applications in real-world scenarios.

Methodology

PEGASUS which is a state-of-the-art abstractive summarization model introduced by Zhang et al. in 2020 is based on the transformer (encoder – decoder) architecture and is specifically designed for generating high-quality summaries. Pegasus was pretrained using a gap generation task. It masks out certain sentences in the input documents and trains the model to predict these masked sentences. The Pegasus large model was pretrained using xsum, cnn_dailymail, newsroom, multinews, gigaword, wikihow, arxiv, billsum, reddit_tifu, and bigpatent. Pegasus has 568 million parameters.

BART is which is another SoTA large language model that is [6] a denoising autoencoder for pretraining sequence-to-sequence models and uses a transformer architecture. BART is trained by (1) corrupting text with an arbitrary noising function, and (2) learning a model to reconstruct the original text [6]. The encoder does the denoising while the decoder does the text generation. It is a transformer with a bidirectional encoder and an autoregressive (GPT-like) decoder. These features make it like BERT [7]. BART has around 406 million parameters. The version used in this project was finetuned on the CNN daily mail dataset for text generation. This model is finetuned to perform text summarization using the SAMSUM and SciTLDR dataset.

The datasets used in this project are SAMSUM and SciTLDR. The SAMSUM dataset is a collection of human-annotated dialogues designed for abstractive summarization. It's a parallel data made up of two speakers, along with corresponding summaries of these dialogues. Each dialogue is paired with a human-written summary that captures the essential information or main points of the conversation. These human annotated summaries are quality assured and great for the field of abstractive text summarization research. SAMSUM reflects natural interactions between speakers discussing various topics/scenarios as it includes multi-turn dialogues. These dialogues cover a wide range of subjects. There are over 15k examples in the dataset.

The SCiTLDR dataset is the second dataset the models were finetuned on and it contains lengthy scientific documents covering a diverse range of topics in various fields such as computer science, biology, and physics. Each document is paired with a concise summary that captures the main points or findings of the article. SciTLDR helps in developing systems capable of generating informative and coherent summaries of scientific articles. Currently, the dataset contains over 1992 documents for training, 618 for validation and 618 for testing.

We finetune each of these models on each dataset separately and record the ROUGE metric on the datasets after finetuning. ROUGE (Recall-Oriented Understudy for Gisting Evaluation) used to evaluate the quality of machine generated text against the ground truth text. In this case ROUGE is used to evaluate the models' generated summaries. The following ROUGE scores shall be used:

ROUGE-1 measures the overlap of unigram(single-word) tokens between the generated summaries and reference summaries.

ROUGE-2 measures the overlap of bigram (two-word) tokens between the generated summaries and the reference summaries.

ROUGE-L measures the longest common subsequence between the generated summaries and the reference summaries. It focuses on matching sequences of words rather than individual tokens.

We compute precision, recall and F1 measure respectively for each of them.

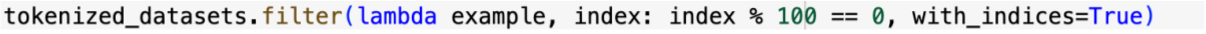
Experimental Setup

Hugging face, a hub for pretrained large language models and datasets was used to download the models and dataset onto the workspace.

The models were finetuned on each dataset from its initial checkpoint. Only a small subset of the datasets was used.

BART on SAMSUM

In finetuning BART on the SAMSUM dataset, 148 samples were selected. The criteria for selection were simply getting the 100th example in the dataset until we got to the end.



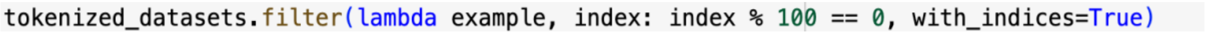
```
tokenized_datasets.filter(lambda example, index: index % 100 == 0, with_indices=True)
```

In the test and validation sets there were 9 samples each.

We finetuned only for one epoch and the loss was calculated for the model after the training was done for the that epoch. The loss function that was used the cross-entropy loss.

BART on SciTLDR

In finetuning BART on the SCiTLDR dataset, 100 samples were selected. The criteria for stayed the same as for the SAMSUM dataset.



```
tokenized_datasets.filter(lambda example, index: index % 100 == 0, with_indices=True)
```

In the test and validation sets both had 31 samples. We finetuned only for one epoch and the loss was calculated for the model after the training was done for the that epoch. The loss function that was used the cross-entropy loss.

PEGASUS on SAMSUM

The SAMSUM train dataset contained 148 documents as well and had 9 documents in both the validation and test sets. Similarly to the previous models, we finetuned only for one epoch and the loss was calculated for the model after the training was done for the that epoch. The loss function that was used the cross-entropy loss.

PEGASUS on SCiTLDR

The train set contained 100 documents/examples, the validation contained 31 and the test set contained 10 examples.

We tested the performance of all the models finetuned on each of these datasets using the ROUGE metric.

Results

BART on SAMSUM:

After finetuning BART on SAMSUM, these were the following ROUGE scores:

ROUGE-1: (precision: 1.28% – 2.1%, recall: 100% Recall, F1: 2.5% - 4.2%),

ROUGE-2: (precision: 1.23% - 2%, recall: 100%, F1: 2.4% - 3.9%) ,

ROUGE-L: (precision: 1.3% - 2.13%, recall: 100%, F1: 2.5% - 4.17%).

The model after one epoch of training achieved a 1.14 training loss and 1.13 validation loss.

BART on SCiTLDR

After finetuning BART on SAMSUM, these were the following ROUGE scores:

ROUGE-1: (precision: 6.5% - 7.9%, recall: 100%, F1: 12.3% - 14.6%),

ROUGE-2: (precision: 6.5% - 7.9%, recall: 100%, F1: 12.3% - 14.6%),

ROUGE-L: (precision: 6.5% - 7.9%, recall: 100%, F1: 12.3% - 14.6%)

The model after one epoch of training achieved a 1.52 training loss and 2.33 validation loss.

PEGASUS on SAMSUM

After finetuning PEGASUS on the subset of SAMSUM, these were the following ROUGE scores:

ROUGE-1: precision: 1.29% – 2.1%, recall: 100%, F1: 2.5% - 4.2%

ROUGE-2: precision: 1.25% - 2%, recall: 100%, F1: 2.4% - 3.9%

ROUGE-L: precision: 1.31% - 2.13%, recall: 100%, F1: 2.5% - 4.17%

The train loss after one epoch was 0.77 and the validation loss was 0.66.

PEGASUS on SCiTLDR

ROUGE-1: precision: 6.1% – 8.5%, recall: 100%, F1: 11.4% - 15.5%

ROUGE-2: precision: 6.0% - 8.4%, recall: 100%, F1: 11.2% - 15.3%

ROUGE-L: precision: 6.1% - 8.4%, recall: 100%, F1: 11.4% - 15.5%

The train loss after one epoch was 0.35 and the validation loss was 1.89.

Discussion

We firstly observe that the validation loss values for all the experiments with the models are low and this shows that both models are great for text summarization. The results also show that the performance of these models varies based on the dataset used. The models perform better as seen by the ROUGE scores when the SCiTLDR dataset was used. Furthermore, Pegasus and BART perform similarly on the SAMSUM dataset, while BART performs better on the SCiTLDR dataset. Overall, the PEGASUS model performed better than BART.

To improve the performance seen by these models need to be finetuned for longer epochs and with more data.

Conclusion

In conclusion, this study highlights the effectiveness of transformer-based models such as Pegasus and BART for text summarization tasks. While both models show promising results, their performance varies depending on the dataset. Future research should focus on improving the generalization of these models to other datasets and exploring techniques to enhance their performance further.

References

[1] Hans Peter Luhn. (1958). "The Automatic Creation of Literature Abstracts." IBM Journal of Research and Development, 2(2), 159-165.

[2] Rada Mihalcea and Paul Tarau. (2004). "TextRank: Bringing Order into Texts." Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP), 404-411.

[3] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. (2014). "Sequence to Sequence Learning with Neural Networks." Advances in Neural Information Processing Systems (NeurIPS), 3104-3112.

[4] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. (2015). "Neural Machine Translation by Jointly Learning to Align and Translate." Proceedings of the 3rd International Conference on Learning Representations (ICLR).

[5] Zhang, J., Lu, Y., & Lapata, M. (2020). PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization. arXiv preprint arXiv:1912.08777.

[6] Lewis, M. T., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., ... & Zettlemoyer, L. (2019). BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. arXiv preprint arXiv:1910.13461.

[7] Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv preprint arXiv:1810.04805.

[8] Gliwa, B., Mieskes, M., & Volske, M. (2019). SAMSum Corpus: A Human-annotated Dialogue Dataset for Abstractive Summarization. arXiv preprint arXiv:1911.12237

[9] Cohan, A., Feldman, S., Beltagy, I., Downey, D., & Weld, D. S. (2018). SciTLDR: A Large-Scale Scientific Long Document Dataset for Abstractive Summarization. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers) (pp. 601-612).