Abstractive Text Summarization – CNN/Daily Mail

Individual Final Report

Author: Hemangi Kinger

Email: hkinger1102@gwu.edu

Overview of Project

Abstractive text summarization is the task of generating a headline or a short summary consisting of a few sentences that captures the salient ideas of an article or a passage. We use the adjective ‘abstractive’ to denote a summary that is not a mere selection of a few existing passages or sentences extracted from the source, but a compressed paraphrasing of the main contents of the document, potentially using vocabulary unseen in the source document.

CNN/Daily Mail is a dataset for text summarization. Human generated abstractive summary bullets were generated from news stories in CNN and Daily Mail websites as questions (with one of the entities hidden), and stories as the corresponding passages from which the system is expected to answer the fill-in the-blank question. The authors released the scripts that crawl, extract, and generate pairs of passages and questions from these websites.

Roles and Responsibility

Team Member	Area of Work	Shared Responsibility
Varun Shah	Data Preprocessing and T5 Model	Fine-tuning
Hemangi Kinger	Model interpretation and BART Model	Fine-tuning
Ishan Kuchroo	Build own trainer, GPT-2, PL-BART, and other models	Fine-tuning

What is my responsibility?

I have taken the primary responsibility of working on BART model along with working on the model interpretation.

In addition to this:

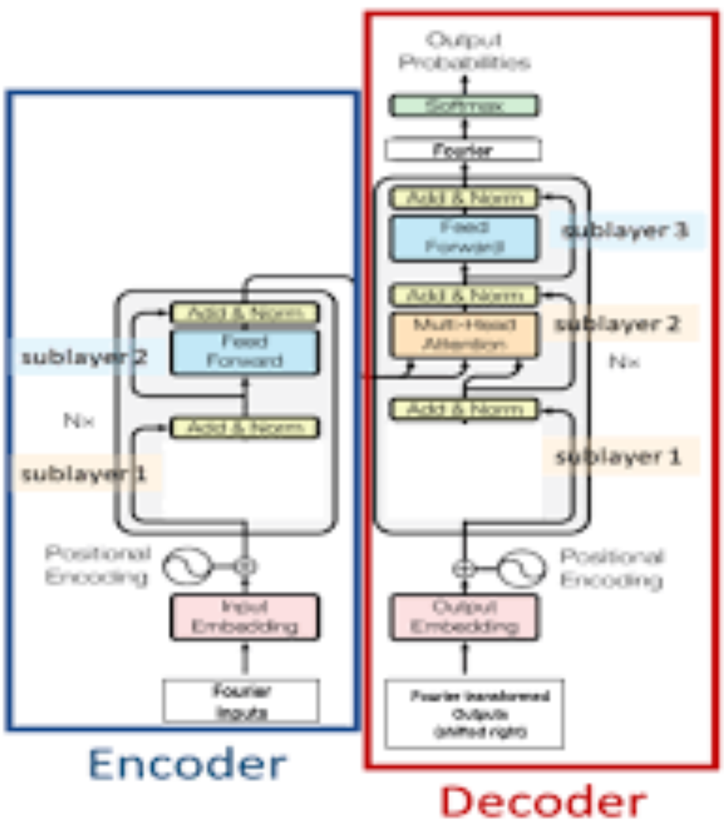
- I’ll be proof-reading and making changes in the summary report created by team
- Consolidating the code of data-preprocessing and modelling and creating a pipeline to ensure the code runs smoothly.

<u>Model Training and Fine-Tuning</u>

Multiple transformers were trained and fine-tuned (including PL-BART, GPT-2, Prophet-Net, MT5-Small etc.) but here we'll talk about BART network based on ROUGE score).

1. **BART**

Bart uses a standard seq2seq/machine translation architecture with a bidirectional encoder (like BERT) and a left-to-right decoder (like GPT). The pretraining task involves randomly shuffling the order of the original sentences and 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 with comparable training resources on GLUE and SQuAD, achieves new state-of-the-art results on a range of abstractive dialogue, question answering, and summarization tasks, with gains of up to 6 ROUGE.



RESULTS

	1	2	3	4	5 increased the train size to 5000
Optimizer	**AdamW**	**Adam**	**Adam**	**Adam**	**Adam**
Epoch	5	5	5	5	5
Learning Rate	0.00001	0.00001	0.00001	0.00001	0.00001
Batch Size	3	3	3	3	3
Max Length Input	264	264	264	264	264
Target Max length	64	64	80	90	90
rouge1	19.2038	19.2696	19.2645	19.2326	19.7834
rouge2	7.3176	7.2924	7.3135	7.3823	7.9065
rougeL	16.1014	16.11	16.2004	16.3171	16.7397
rougeLsum	16.0577	16.0609	16.1978	16.292	16.7586

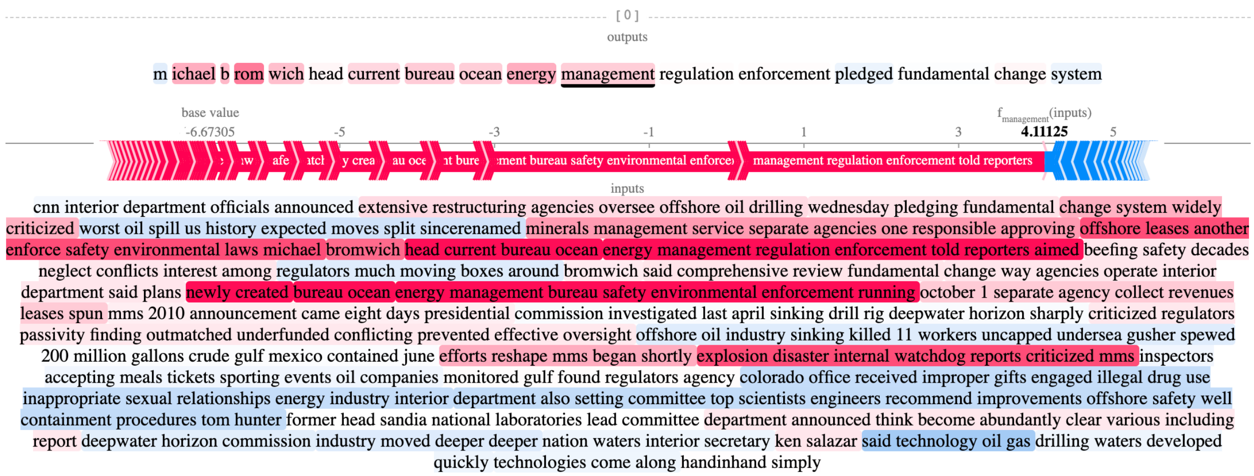
From my analysis of different transformers, we can conclude best configuration for BART is the fifth version as it has the highest rouge score

Model Interpretation

I have implemented SHAP for explainability of our BART model as BART was giving the best score for our model.

1. **SHAP**

   SHAP is a Python library that uses Shapley values to explain the output of any machine learning model. Finding the Shapley value of each feature can help us determine their contribution. When we pass a single instance to the text plot, we get the importance of each token overlayed on the original text that corresponds to that token. Red regions correspond to parts of the text that increase the output of the model when they are included, while blue regions decrease the output of the model when they are included.



Conclusion

From my analysis of different transformers, we can conclude best configuration for BART is the fifth version as it has the highest rouge score. Along with this we can see the model interpretation of the best BART model using SHAP

Referenced Code %

$$(450 - 190)/ 450 + 170 * 100 = \textbf{37\%}$$

References

https://huggingface.co/course/chapter7/5?fw=pt

https://shap.readthedocs.io/en/latest/example_notebooks/text_examples/summarization/Abstractive%20Summarization%20Explanation%20Demo.html

https://medium.com/analytics-vidhya/text-summarization-using-bert-gpt2-xlnet-5ee80608e961

https://huggingface.co/course/chapter3/4?fw=tf#the-training-loop

https://www.kaggle.com/code/sumantindurkhya/text-summarization-seq2seq-pytorch/notebook

https://huggingface.co/docs/transformers/training#train-in-native-pytorch

https://blog.paperspace.com/generating-text-summaries-gpt-2/

http://reyfarhan.com/posts/easy-gpt2-finetuning-huggingface/