1. What is this project?

Summarization has been and continues to be a hot research topic in the data science arena. Summarization is a technique that reduces the size of a document while preserving its meaning. It is one of the most researched areas among the Natural Language Processing (NLP) community.

Summarization techniques are categorized into two classes based on whether the exact sentences are considered as they appear in the original text. New sentences are generated using NLP techniques, which are extractive and abstractive summarization.

Extractive summarization: In extractive summarization, the most meaningful sentences in an article are selected and arranged comprehensively. In short, the summarized sentences are extracted from the article without any modifications.

Abstractive Summarization: It is a task in NLP that aims to generate a concise summary of a source text. Unlike the extractive summarization technique, abstractive summarization does not simply copy essential phrases from the source text but also potentially come up with new relevant phrases, which can be seen as paraphrasing.

Abstractive summarization has several applications in different domains such as,

- Science and R&D
- Books and literature.
- Financial research and legal documents analysis
- Meetings and video conferencing

This project aims to build a BART model that will perform abstractive summarization on a given text data. Those who are busy but still want to catch up the daily news will be our primary customers.

2. Data Description

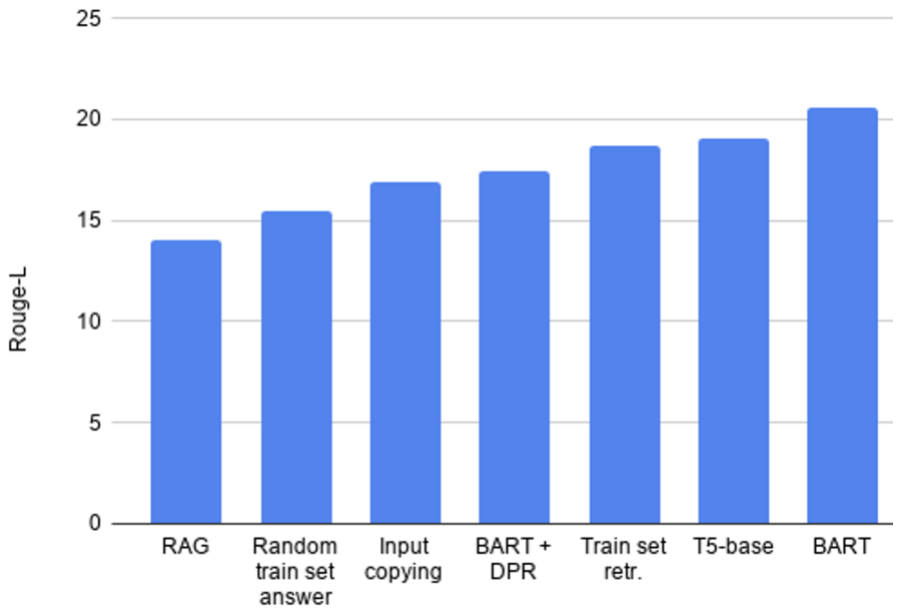
The data used is from the curation base repository, which has a collection of 40,000 professionally written summaries of news articles, with links to the articles themselves. The data is cloned from GitHub. Then data is downloaded in the form of a CSV file and has the following features:

- Article titles – title for the texts
- Summaries – Summary for each text
- URLs – the URL links
- Dates
- Article content – content under each article

3. Approach

1. Import the dataset from the dataset library and load a subset of the dataset.(To get an overview of the summarized data)
2. Clone the repository.
3. Download the article titles, summaries, URLs, and dates (a CSV file)
4. Create a new environment, install the requirements and scrape the data.
5. Change the run time to GPU.
6. Import the required packages and libraries.
7. Create a class function for the dataset.
8. Create a class function for the BART data loader.
9. Create an abstractive summarization model class function.
10.Create a BART tokenizer
11.Define the data loader
12.Read and prepare the data.
13.Perform train test split.
14.Create the main class that runs the 'BART For Conditional Generation' model and tokenizer as an input.
15.Define the trainer class and then fit the model.
16.Perform the BART summarization using the pre-trained model. 17.Understand the concept behind the BART evaluation metric – Rouge.

4. Plots
a).
Metrics: Rouge-L, the larger, the better.



b). Visualize the performance of models over dif- ferent evaluation metrics.