NLP Project: Text Summarization

Ahmed Serry, Farida Helmy and Seif Maged

May 8, 2023

Abstract

This NLP project aims to develop a text summarization [Moh21] model that can generate a concise and coherent summary of a given text document while preserving the most important information and key ideas. The project will explore various approaches to extract and prioritize relevant information. The model will be trained and evaluated on a large dataset of diverse text documents, The dataset That we chose to experiment on is the CNN Daily News Dataset from Huggingface [HKG+15]. The project's ultimate goal is to improve the efficiency of information processing and facilitate effective communication by providing a quick and accurate summary of lengthy texts.

1 Introduction

Text summarization is an important task in all large language-generating models. With the outburst of Chat-GPT, our team decided to investigate one of its main tasks which is Text Summarization. There are two ways to look at this task: Constructing an Extractive and Abstractive model. Extractive techniques choose the most significant sentences from the huge bulk of text, regardless of whether they comprehend the context or not, and hence create a summary that is merely a portion of the complete document. On the other hand, Abstractive approaches require sophisticated NLP techniques, such as word embeddings, to comprehend the meaning of the text and produce a coherent summary. As a result, Abstractive methods are considerably more challenging to develop from the beginning, as they require a vast amount of data and parameters. That's why in our project, to benefit from NLP techniques that we got familiar with in this course, our team decided to focus on the Abstractive techniques when implementing this task. In this project, we used the CNN Daily news dataset to experiment the different techniques of abstractive summarization using Attention-based Transformers. In the following sections, we will propose the data preprocessing required to shape the data correctly for our models, the data analysis to have general knowledge and expectations from our data and we will propose the architecture for our project to how we will achieve the text summarization task successfully.

2 Fetching the Dataset

The dataset that we have at our hand was extracted from huggingface. Then we split the dataset into training dataframe consisting of 287113 rows, validation dataframe consisting of 13368 and testing dataframe consisting of 11490 rows.

3 Data Preprocessing

Looking at our dataset, we can see that it has 3 columns denoting the articles, their corresponding summaries and ids. The Data Preprocessing techniques will be done mainly on the 2 columns of the articles and the summaries.

3.1 Articles Preprocessing

This column is the most interesting column for our preprocessing task, because it denotes the raw text that we will experiment on to reach the desired summaries. Hence there are multiple techniques that

need to be done in order to properly handle the corpus of the articles.

1. Removing Stop words and punctuation
2. Normalisation
3. Removing Null values
4. Tokenizing each article into words.
5. Tokenizing each article into sentences.
6. Stemming

3.2 Summary Preprocessing

This column is our target column and consists of the raw text summary of the corresponding article in the row. Hence, there are multiple preprocessing techniques that should be done, which are the same techniques mentioned above:

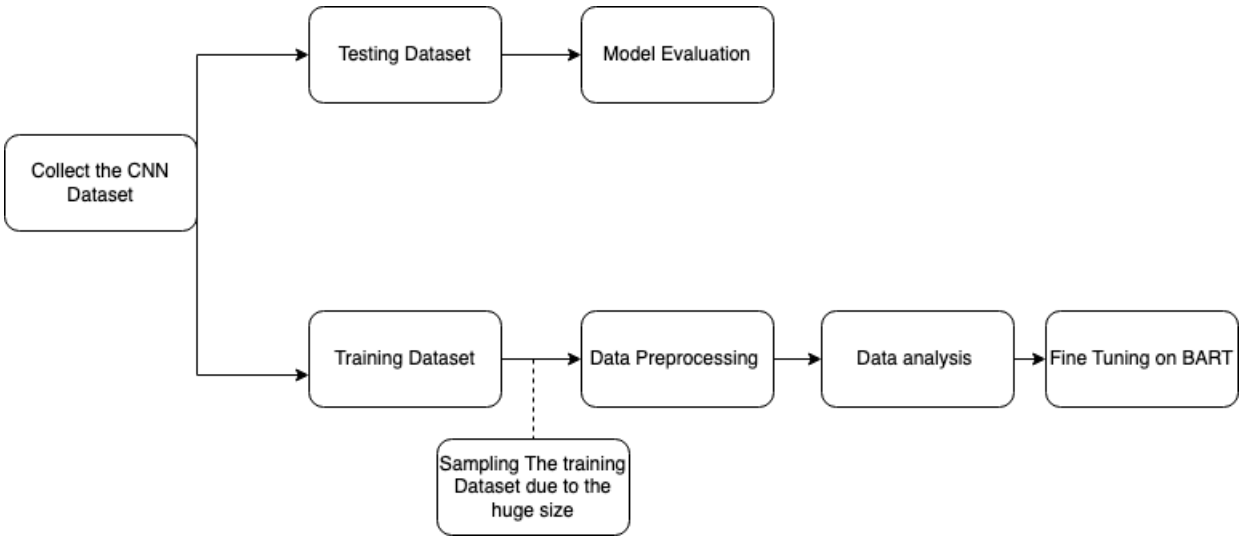
1. Removing Stop words and punctuation
2. Normalisation
3. Removing Null values
4. Tokenizing each article into words.
5. Tokenizing each article into sentences.
6. Stemming

4 Data Analysis

We want to investigate the nature of our articles, hence we studied the number of words in our corpus, the most frequent words in our corpus, and the number of unique words in our corpus. Furthermore, we wanted to study the average sentences per article in our dataset to investigate the size of the raw data that we want to summarize. This is applied on both columns: the article and the summary.

5 System Architecture

In this section, we will propose our system architecture for completing this project: After exploring the previous work done on text summarization, to the best of our knowledge, the area of applying BART [LLG+20] and GPT is the state of the art of accomplishing this task. Hence, we will try to analyze the power of this Bart after fine-tuning it for the CNN Daily News Dataset. A diagram displaying our system architecture below describes the road map for our project.



References

[HKG+15] Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. Cnn/daily mail. https://huggingface.co/datasets/cnn_dailymail, 2015.

[LLG+20] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. pages 270–278, July 2020.

[Moh21] Abhishek Mohan. Text summarization with nlp: Textrank vs seq2seq vs bart. *Towards Data Science*, June 2021.