CS 6120: Text Summarization and Visualization using stable diffusion

Pravin Anand Pawar
Northeastern University
pawar.prav@northeastern.edu

Ameya Rane
Northeastern University
rane.am@dnortheastern.edu

Gourav Beura
Northeastern University
beura.g@northeastern.edu

Abstract

The BBC News Summary Dataset is analyzed in this study using three well-liked text summarization methods: the centroid approach, the ranking approach, and the BERT model. We experiment with ROUGE evaluation metrics, and investigate the effects of various parameters, such as the summary's length and the weighting scheme applied to the ranking approach. According to our findings, the BERT model performs better at producing high-quality summaries than the centroid and ranking approaches. Both approaches continue to produce comparable results, despite the centroid approach consistently outperforming the ranking approach. Additionally, our analysis shows that the best course of action depends on the particular dataset and evaluation metric being used. This paper adds to the body of text summarization research by offering an in-depth examination of three well-known text summarization methods and their performance on the BBC News Summary Dataset, including recent advancements utilizing pre-trained models like BERT. Submission Link[10]

1 Introduction

The necessity of summarizing lengthy text passages has grown due to the recent explosion of digital information. The process of condensing a longer text into a shorter version while preserving its main ideas and meaning is known as text summarization. It has numerous uses in the areas of machine learning, natural language processing, and information retrieval.

There are many different methods for summarizing text, including extractive, abstractive, and hybrid approaches. Our focus is on extractive summarization. Extractive summarization is one of the most popular approaches to text summarization, and it involves selecting and combining important sentences or phrases from the original text.

Two of the most popular extractive methods for text summarization are the centroid approach and the ranking approach. The centroid approach selects sentences closest to the centroid of the document, whereas the ranking approach ranks sentences based on their importance and selects the highest-ranked sentences for the summary. Another popular extractive method is the ranking approach, which ranks sentences according to their significance and selects the top-ranked sentences for the summary.

However, with the emergence of advanced deep learning techniques, new approaches to text summarization have also been proposed. In particular, the use of pre-trained models like BERT (6) has shown promising results in generating high-quality summaries.

In this research paper, we investigate the implementation of the centroid and ranking approaches as well as the BERT model for text summarization. We conduct experiments on BBC News Summary dataset and compare the performance of these three methods using various evaluation metrics, including ROUGE and BLEU. Additionally, we explore the impact of different parameters, such as the length of the summary and the weighting scheme used in the ranking approach.

The remainder of the paper is structured as follows. A review of related work in text summarization is given in Section 2. The dataset and preprocessing procedures used in our experiments are described in Section 3. The implementation of the centroid and ranking approaches is covered in detail in Section 4. The analysis and results of the experiment are reported in Section 5. Section 6 wraps up the paper and discusses further research. Finally, future work associated with this project in Section 7.

2 Background/Related Work

There are multiple approaches to extractive text summarization as outlined in (5). This includes Machine Learning and Artificial Intelligence ap-

proaches, Conditional Relational Field, Graph based approaches, Concept Based approaches and Fuzzy Logic approaches. In this project we will be focusing on two approaches: Ranking and Centroid Based approaches to identify sentences to be used in the extractive summary.

3 Data

3.1 Dataset Summary

The Dataset used is the BBC News Articles Extractive Summarization dataset[4]. This dataset for extractive text summarization has four hundred and seventeen political news articles of BBC from 2004 to 2005 in the News Articles folder. For the purpose of evaluation, all the news articles have been selected from the politics section.

The news articles have a mean number of sentences as 23.3, and the test summaries have a mean of 2.12 sentences. The news articles considered has a total vocabulary size of 12134 words.

4 Methods

4.1 Word embedding/ analysis

These days word embeddings have become most frequently used approach to represt words in natural language processing. Word embeddings are dense, low-dimensional representations of words that capture their semantic and syntatic meanings. Using these embeddings powerful machine learning algorithms can analyze and understand relationships between words in a document. We have used Word2Vec algorithm, which learns word embeddings by predicting the context of word in a large corpus of text. These embeddings can be used in various text summarization approaches, to identify important sentences or phrases and create a summary of the original text.

4.2 Maximal Marginal Relevance (MMR)

This algorithm builds a reranking criterion that ranks the sentences in the corpus in order of relevance and novelty. After fixing the number of sentences that need to be added to the summary, the algorithm adds the highest ranked sentence to the summary and then recalculates the ranking. This process is repeated until the size of the summary is filled. The ranking (2) is calculated as:

$$\lambda sim_1(D_i, Q) - (1-\lambda) \max_{D_j \in S} sim_2(D_i, D_j) \quad (1)$$

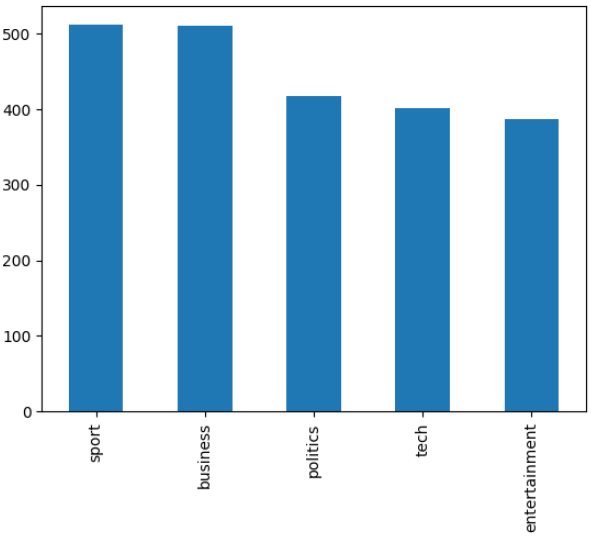

Figure 1: Distribution of summaries for each category

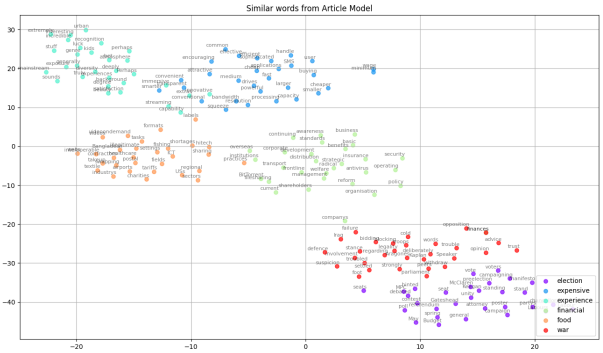

Figure 2: Graph showing similar words embeddings for news articles

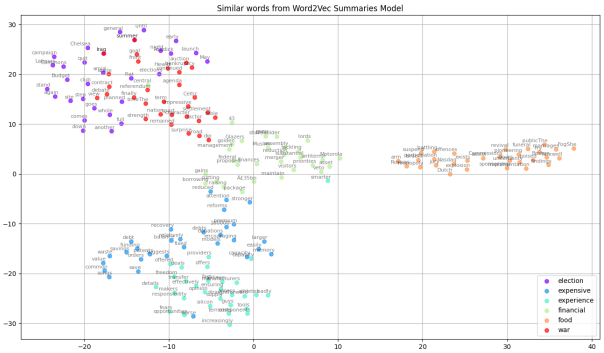

Figure 3: Graph showing similar words embeddings for summaries

This takes into account the similarity of the sentence to the query and also penalizes insertion of sentences into the summary set that are similar to the other sentences in the summary set.

4.3 Centroid-based: Global Selection

In this algorithm, sentence for summary is selected greedily based on highest cosine similarity between document centroid vector and potential summary vectors, combination of sentence with current summary. Instead of searching for best sentence individually for summary, this approach focuses on finding sentence combinations that best summaries given document. Hence, sentence combinations having highest similar to centroid of document is selected as best summary, as it will provide more information about given document than any other combination of sentences for given summary limit on number of sentences. In each iteration of greedy selection, new sentence is added to current summary until no sentence left in document or number of sentence limit of summary is reached.

Sentences are represented using BOW vectors with TF-IDF weighting and document centroid as sum of all sentence vectors. TF-IDF help to represent sentence with fewer words which provides more importance than naively selecting all complete BoW. We have further make use of Skip-gram and CBOW based word embedding as an alternative to TF-IDF representation of data. Cosine similarity is the dot product of two vectors divided by its magnitudes of two vectors. It is cosine of angle between two vectors and provide measure of closesness or similarity of two vectors.[1]

$$Cosine\ Similarity = sim(a, b) = a.b/|a||b| \tag{2}$$

4.4 BERT

BERT(Bidirectional Encoder Representations from Transformers) has shown great performance on many NLP tasks. In this paper, we are using BERT for extractive text summarization. In this appproach, input texts are tokenize and sentence embedding are created using BERT model. Furthermore, clustering algorithm is applied to fetch sentences have their embedding closest to the article centroid. Thus, for given input text algorithm is able to provide most relevant output sentences for summary.

In order to compare the performance of transformers for summarization, we have used PEGASUS (7) model developed by Google. This is useful for summarizing news articles, academic papers, and other lengthy documents.

4.5 Stable Diffusion

Stable Diffusion is latent diffusion model which convert text to image. It is a used in image processing task for denoising and inpainting. It is powerful algorithm and known to perform well on images to remove gaussian and other noise in image. In this paper, we are using this algorithm to provide relevant images to our predicted summaries to get idea of effectiveness of our model summaries.

5 Experiments

In this paper, we performed multiple ablation studies with different summarization approaches and transformers. For all the methods we have used BBC News dataset and split them into 80% training, test and validation dataset.

5.1 Ranking

The model was run on 20% of summaries from the BBC news summarization dataset, using articles from the "political" category. The summarization function took the following parameters:

- n: This is the percentage of the number of sentences in the original passage that need to be included in the final summary. This was set at 0.1 to reflect the sizes of the gold summaries included in the training data.

- λ: The weighting parameter that represents the importance of relevance vs diversity in the calculation of the MMR score while running the ranking algorithm.

- *Similarity Function* Two different functions were used to evaluated similarity between the two sentences. A CountVectorizer was used to measure similarity between sparse representations using word count of the two sentences. Another representation was also created using a weighted average of inverse word frequencies of dense embeddings trained on the corpus using Word2Vec. A linear interpolation of the two representations was used to determine sentence similarity.

- *Query Strategy* The query sentence taken to measure relevance of the candidate sentences were either selected as the title of the article, or simply the entire article itself.

For the baseline model, the sentence vector representations were created by using a count vectorizer on the sentences to be compared. Each word in the sentences was stemmed using the PorterStemmer class included in nltk. The query was considered as the vector created by the complete new article. 80% of the data was marked as a training set which was used to generate the dense word embeddings.

Most of the summaries provided had 10% the number of sentences that the original article contained. Therefore, the value of n was set at 0.1. The extracted summary was compared with the summary provided using the Rouge Evaluation metric.

The exploratory analysis shows that none of the hyperparameters had a very significant impact on the score of the summaries produced by the algorithm. The most significant impact was by the choice of the query sentence.

Selecting the query as the title of the summary rather than the entire summary consistently provided summaries with better f1 rouge-2 scores and was the strategy that was used while moving forward.

5.2 Centroid-based: Global Selection

BBC News Articles dataset[4] is used as input articles to summarize. Basic preprocessing is done to remove commonly found punctuations and character symbols. For baseline model, each sentence is represented in terms of significant words using TF-IDF vectorizer. Model was run on training data(80 percent of dataset) and predicted summary sentence length was kept to 10 percent of article size, as 1:10 ratio is seen between size of summary and article. Further, model is tested on test set consist of 20 percent of dataset. Final extracted predicted summaries are compared with reference summaries using rouge metric. Model is further evaluated using Skip-gram and CBOW word embedding. Histogram of precision, recall and F1 score are generated to get idea of distribution of values. Refer figure 4, 5 and 6 for Centroid-based summary performance using skip-gram embedding.

5.3 BERT

For the extractive summarization using transformers, we used pre-trained BERT for our ablation

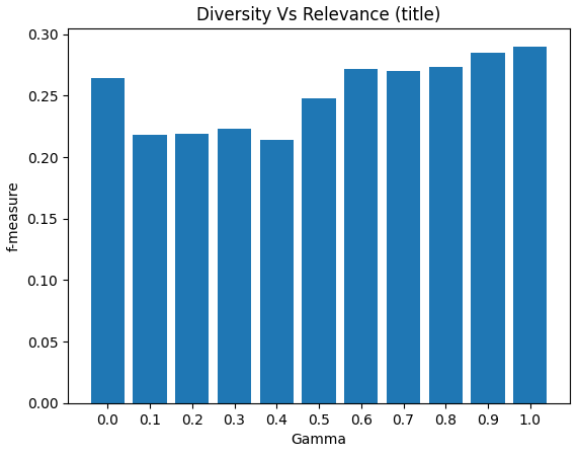

Figure 4: Performance of MMR with weightage given to relevance vs diversity using the title as query sentence

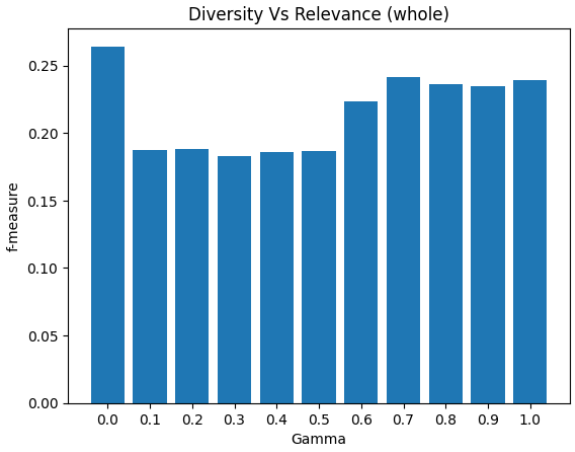

Figure 5: Performance of MMR with weightage given to relevance vs diversity using the article as query sentence

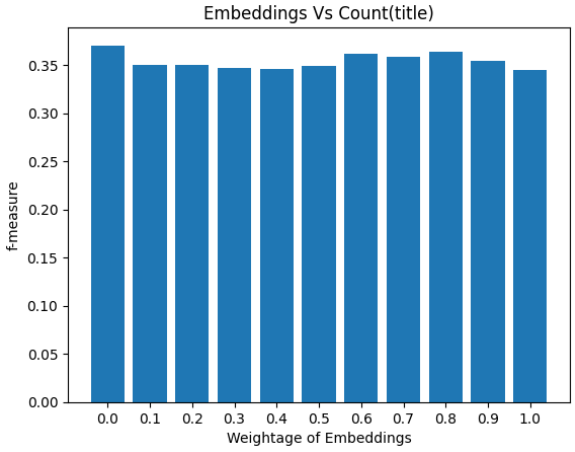

Figure 6: Performance of MMRwith similarity function weightage given to dense embeddings vs sparse vectors

studies. The extracted summaries were mainly obtained from articles of various news categories. In our baseline experiment, we compared the ROUGE score by comparing the model generated summaries against the gold summaries provided in the dataset.

5.4 PEGASUS

For further ablation studies, we used another model called PEGASUS (by Google) (7). It is useful for summarizing news articles, academic papers, and other lengthy documents. It has achieved state-of-the-art results in several NLP tasks, including text summarization and language generation. We performed transfer learning on the model and got some interesting results.

PEGASUS and BERT share some similarities in their underlying architecture, but they are fundamentally different models with different objectives.

5.5 Model Evaluation

The extracted summaries were evaluated using the ROUGE (3) evaluation metrics using N-gram co-occurence statistics. The bigram values were used to generate the valuation metrics.

$$Rouge - 2 = \frac{\sum_{bigram \in S} Count_{match}(bigram)}{\sum_{bigram \in S} Count(bigram)} \quad (3)$$

5.6 Results

Results for different model evaluation are as follows:

5.6.1 Ranking

Below results are based on Rouge2.
The following results were obtained for the baseline model with Skip-gram embeddings:
Train dataset

Precision	Recall	Fmeasure
0.129	0.477	0.200

Test dataset

Precision	Recall	Fmeasure
0.130	0.468	0.201

The following results were obtained for the baseline model with CBOW embeddings:
Train dataset

Precision	Recall	Fmeasure
0.127	0.453	0.196

Test dataset

Precision	Recall	Fmeasure
0.129	0.472	0.200

The results show that the selection or training domain of embeddings does not seem to have a major impact on the performance on the MMR algorithm as much as the variation of internal hyper-parameters.

5.6.2 Centroid-based: Global Selection

Below results are based on Rouge2.
Results for Centroid-based approach on dataset are as follows:

Training set using TF-IDF based approach:

Precision	Recall	Fmeasure
0.810	0.156	0.254

Testing set using TF-IDF based approach:

Precision	Recall	Fmeasure
0.82	0.16	0.26

Training set using Skip-gram embeddings:

Precision	Recall	Fmeasure
0.639	0.134	0.216

Testing set using Skip-gram embeddings:

Precision	Recall	Fmeasure
0.610	0.129	0.207

Training set using CBOW embeddings:

Precision	Recall	Fmeasure
0.576	0.115	0.187

Testing set using CBOW embeddings:

Precision	Recall	Fmeasure
0.555	0.113	0.1836

5.6.3 Pre-trained BERT

Results for BERT-based approach on a subset dataset are as follows:

Precision	Recall	Fmeasure
0.68	0.42	0.51

5.6.4 Transfer Learning on PEGASUS

Results for ROUGE score improving after model was trained on custom dataset for 3 epochs. The table below shows the scores:

Stage	Rouge1	Rouge2	RougeL
Pre-transfer learning	0.032	0.0014	0.030
Post-transfer learning	0.036	0.002	0.034

High Precision value is seen in centroid based approach as compare to ranking based model. Above table contains mean values of precision, recall and Fmeasure. Distribution of same are

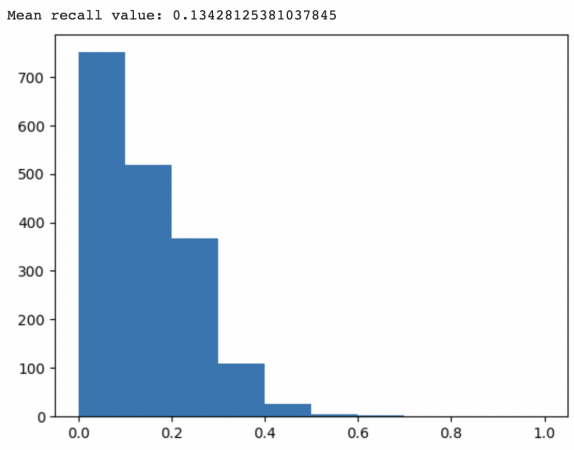

Figure 7: Recall score distribution for Skip-gram Centroid based approach

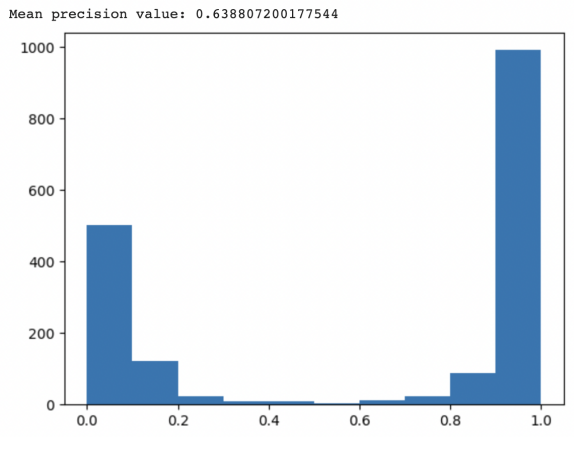

Figure 8: Precision score distribution for Skip-gram Centroid based approach

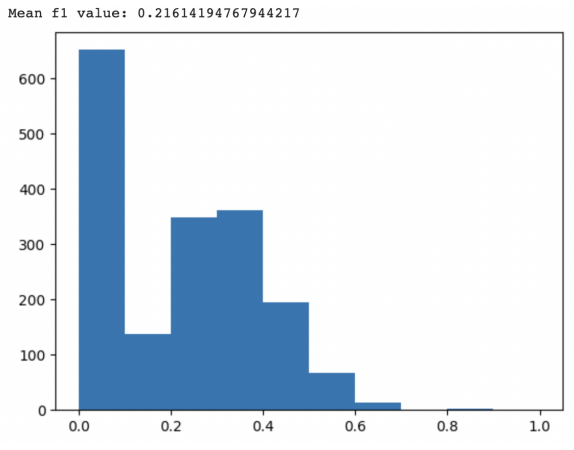

Figure 9: F1 score distribution for Skip-gram Centroid based approach

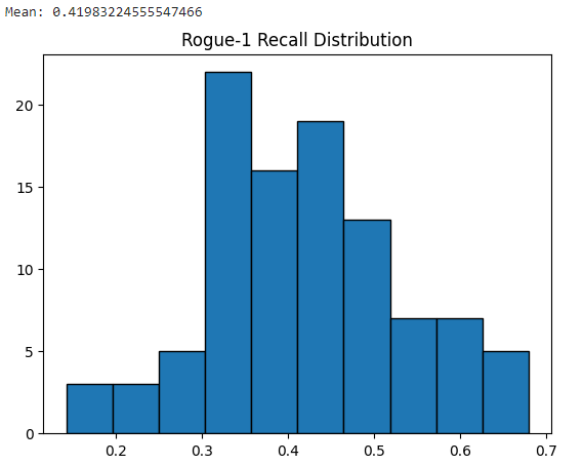

Figure 10: Recall score distribution for pre-trained BERT

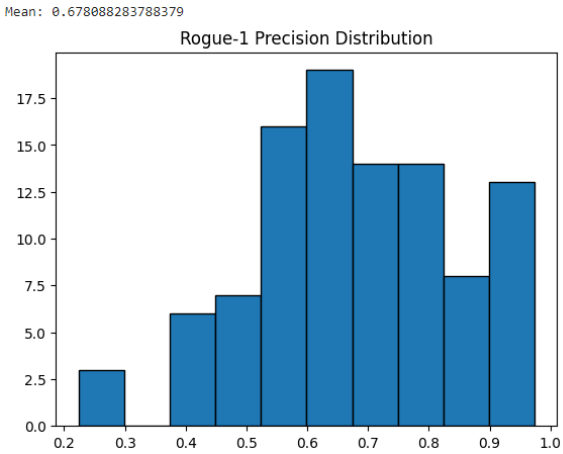

Figure 11: Precision score distribution for pre-trained BERT

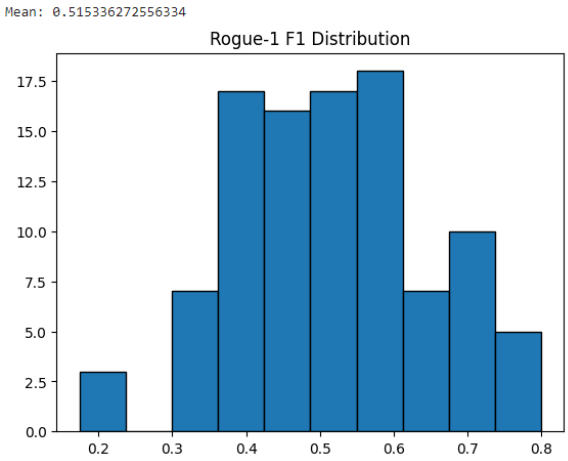

Figure 12: F1 score distribution for pre-trained BERT

Figure 13: Stable Diffusion Output for the summary: "Ferguson did not play in the FA Cup win over Colchester on Saturday despite recovering from a groin injury with Rovers boss Mark Hughes claiming it had been an emotional and difficult time for the player. Williams said We are in dialogue with Glasgow Rangers but we have no agreement."

Figure 14: Stable Diffusion Output for the summary: "The bank along with investors auditors and the groups managers wants damages for being a victim of fraud at the hands of the Italian firm."

shown in figure 4, 5 and 6. The output obtained from stable diffusion are shown in figure 13, 14 and 15.

6 Conclusions

We were successfully able to implement are baseline models for extractive text summarization using ranking and centroid based approach as well as used advance models like BERT and PEGASUS. We have also performed various experiments using different sentence representation like TF-IDF, Skip-gram and CBOW. High precision is seen in centroid based approach as compare to ranking. But advance models like BERT was able to outperform both ranking and centroid approach by huge difference.

7 Future Work

In this paper, we have not analysed all state of the art neural architectures specialized for extractive summarization, such as HiBERT (8), and GSum (9) which have additional parameters to control the sentences selected for extractive summarization. Moreover, the different variations of the BERT architecture like RoBERTA, and DeBERTa could also be analysed to see whether they have a significant impact on our findings.
To make our statistical findings more robust, we would also need to perform the experiments on datasets with a vastly different vernacular and style and see whether the same results hold.

References

[1] Ghalandari, D. G. (2017, September). Revisiting the Centroid-based Method: A Strong Baseline for Multi-Document Summarization.

[2] Jaime Carbonell and Jade Goldstein. 1998. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR '98). Association for Computing Machinery, New York, NY, USA, 335–336. https://doi.org/10.1145/290941.291025

[3] Lin, C. (2004). ROUGE: A Package for Automatic Evaluation of Summaries. Annual Meeting of the Association for Computational Linguistics.

[4] BBC News Articles Extractive Summarization dataset. https://www.kaggle.com/datasets/pariza/bbc-news-summary

[5] N. Moratanch and S. Chitrakala, "A survey on extractive text summarization," 2017 International Conference on Computer, Communication and Signal Processing (ICCCSP), Chennai, India, 2017, pp. 1-6

[6] N. Andhale and L. A. Bewoor, "An overview of Text Summarization techniques," 2016 International Conference on Computing Communication Control and automation (ICCUBEA), Pune, India, 2016, pp. 1-7, doi: 10.1109/ICCUBEA.2016.7860024.

[7] Zhang, J., Zhao, Y., Saleh, M. and Liu, P., 2020, November. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In International Conference on Machine Learning (pp. 11328-11339). PMLR.

[8] Xingxing Zhang, Furu Wei et Ming Zhou. HIBERT: Document Level Pre-training of Hierarchical Bidirectional Transformers for Document Summarization. 2019. arXiv :1905.06566

[9] Zi-Yi Dou et al. GSum: A General Framework for Guided Neural Abstractive Summarization. 2021. arXiv : 2010.08014

[10] https://github.khoury.northeastern.edu/ameyaprane/NLPProjectRepo