News Text Summarization Application using T5 Transformer

Ajit Jadhav

Karan Ajay Pisay

Shree Sharma

Srashti Soni

MPS in Data Science, University of Maryland, Baltimore County

DATA 690: Special Topics in Data Science: Introduction to NLP

Dr. Antonio Diana

November 27, 2022

Abstract

Today, users are shifting from consuming news on traditional platforms like television and print media to digital news platforms and social media. As the amount of news available on digital platforms is increasing rapidly, summarization of news has become essential. News summary presents the users with easy-to-understand text while maintaining the context of the information, and this allows users to read the news and understand the context of the situation in a short duration. In this work, we have built an abstractive news summarization application on Streamlit using a fine-tuned T5 transformer model. The T5 transformer model has been fine-tuned on the CNN-Dailymail dataset which contains more than 300K news articles and their summaries before developing the summarization application.

Introduction

The widespread adoption of the internet has led to exponential growth in the amount of information present on the web. However, because of the hectic schedule of people and the immense number of options available, there is a growing need for information summarization. The idea is to give the user a condensed version of the original article that only contains the most important details and which speeds up their comprehension of the text. Condensing documents or reports into a more manageable size while preserving critical information is the aim of automatic text summarization (Sethi et al.,2017).

In general, automatic text summarization methods fall into two categories:

- Extractive: Here we identify key phrases or sentences in the original text and extract only a few pertinent phrases to form the summary.

- Abstractive: Here a condensed and understandable summary of the source text is created using advanced Natural Language Processing (NLP) techniques.

Abstractive summaries are typically more human-like in their interpretation in comparison to extractive summaries (Batra et al., 2021).

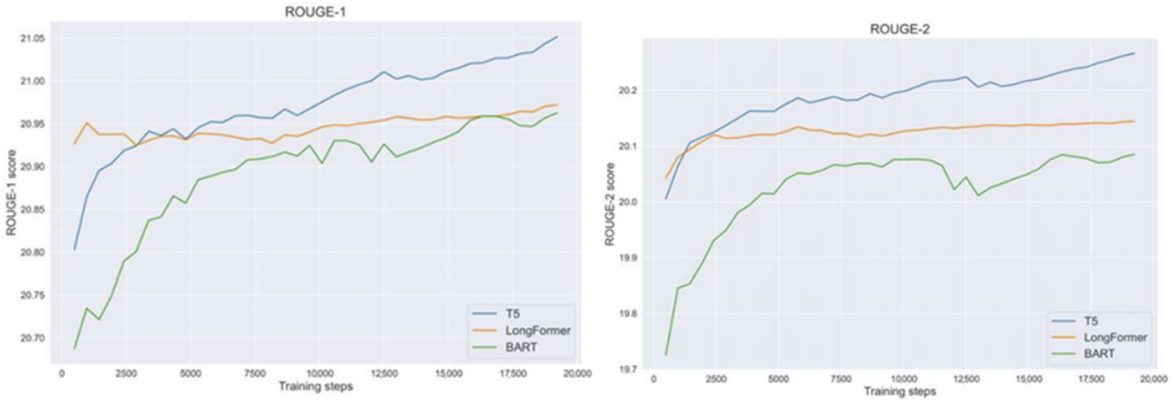
In this research, we have developed 'News Shack' an abstractive news summarization web application using Streamlit. This application allows users to gather summaries of the latest news from Google RSS feed based on different categories. It also allows users to fetch news summaries of their favorite topics. Furthermore, it provides users with an option to select the number of articles they would like to summarize. Hence, this application can be extremely valuable for users who consume news through digital platforms but at the same time, want to read news from reliable sources.

Literature Review

In the area of abstractive summarization, Blekanov et al. (2022) found that it has the potential of the plane models capable of truly understanding text semantics and, thus, generating a summary form of short phrases which can be seen as an idea closer to manual summarization. In these types of models, there are two components: (1) the first one is the encoder, which transforms the initial tokens into their corresponding encodings and (2) the decoder, which generates target summaries using the autoregression method (see Blekanov et al., 2022). The newer transformer architectures helped them create better quality models and have brought one of the first scalable neural text models. The transformer completely depended on the self-attention process, as opposed to classic models that utilized recurrent neural networks and its primary modification, the Long-Short Term Memory (LSTM). As a result, the model could

simulate longer sequences without experiencing data loss, which was an issue that Recurrent Neural Networks (RNN) had in the long run. In addition, that made it feasible to train sequential recurrent models simultaneously, which was previously not conceivable (see Blekanov et al., 2022).

According to Blekanova et al. (2022), modern pre-trained models like BART (Bidirectional Auto-Regressive Transformer) and T5 (Text-to-Text Transfer Transformer) have demonstrated outstanding outcomes for text summarization. In order to provide summaries of news articles, they modified these models in this research with the use of transfer learning. Their findings demonstrated that the T5 model outperformed the tested models. They collected data from social media platforms like Reddit and Twitter to create an analysis that could be very useful for opinion and topic mining due to the complications that social media data create for other methods of textual analysis. The authors also argued that summarizing pools of text may bring crucial results, which can be relevant for social studies. The figure below represents the ROUGE metric scores for the BART, T5, and LongFormer models (see Figure 1).



.

Fig. 1a and Fig. 1b *The fine-tuning results, as evaluated by the ROUGE metrics*

Fig. 2 *ROUGE-L score comparison for BART, T5 and LongFormer model*

After reviewing several applications, Garg et al. (2020) suggested that pre-trained language models have made significant advancements in several NLP applications, such as text summarization. Their research was primarily based on news articles related to politics, business, and commodities. They concluded that the T5 pre-trained model turned out to be very accurate and achieved the best scores, as shown in the figure below (see Table 1).

Table 1 *ROUGE scores for all tested models*

S. No.	Transfer learning	BART	T5
1	0.63	0.58	0.94
2	1.0	0.12	1.0
3	0.67	0.42	0.62
4	0.16	0.21	0.19
5	0.2	0.32	0.18

Dataset Description

Table 2 presents the data fields of the CNN/Daily Mail dataset obtained from the Hugging Face website for fine-tuning the pre-trained T5 small transformer model. It is an English language dataset containing just over 300 thousand unique news articles written by journalists at CNN and the Daily Mail.

Table 2 *Data fields of the CNN/Daily Mail dataset*

Column Names	Description
id	a string containing the heximal formatted SHA1 hash of the URL where the story was retrieved from
article	a string containing the body of the news article
highlights	a string containing the highlight of the article as written by the article author

The CNN/DailyMail dataset has three splits: train, validation, and test. Table 2 presents the statistics for version 3.0.0 of the dataset.

Table 3 *Dataset Split of the CNN/Daily Mail dataset*

Dataset Split	Number of Instances in Split
Train	287,113
Validation	13,368
Test	11,490

For our Streamlit application, we used live news feed from Google API. The fine-tuned T5 model helped summarize the articles extracted using the API. The RSS feed helped the model to extract the most up-to-date information and feed it to the T5 model for summarization. The data is dynamic and hence scrapped by opening the URL and extracting the articles in real time. Table 4 presents the data fields of the Google News dataset.

Table 4 *Data fields of the Google News dataset*

Extracted Column Name	Description
url	It is the URL that is fetched when Google API is gathering data.
article	a string containing the body of the news article
category	This field describes the category to which the news article belongs, e.g. sports, technology, and healthcare.

Methodology

In this section, we detail the steps we followed in creating a news summarization application. The main three parts in our methodology, along with the exploratory data analysis, are data pre-processing, model fine-tuning, and building the summarization application on Streamlit.

Data Pre-processing and Model Fine-Tuning

T5 (Text-to-Text Transfer Transformer) is an encoder-decoder model pre-trained on a multi-task mixture of unsupervised and supervised tasks where each task is converted into a text-to-text format. Pre-trained T5 can be directly used for different tasks by appending the task-specific prefix to each input, e.g., 'summarize' for summarization and 'translate' for translation. We used a T5 small transformer model for our summarizer, and it was fine-tuned using the supervised method where the input and output sequences are a standard sequence-to-sequence input-output mapping. The complete news article text from our dataset is passed as the input sequence to the model and the human-generated summary from the dataset is passed as the output sequence to the model. The input and output sequence lengths are truncated/padded by defining the 'maximum input length' and the 'maximum target length.' For our model, we have set the maximum input sequence length to 512 and the maximum target sequence length to 128.

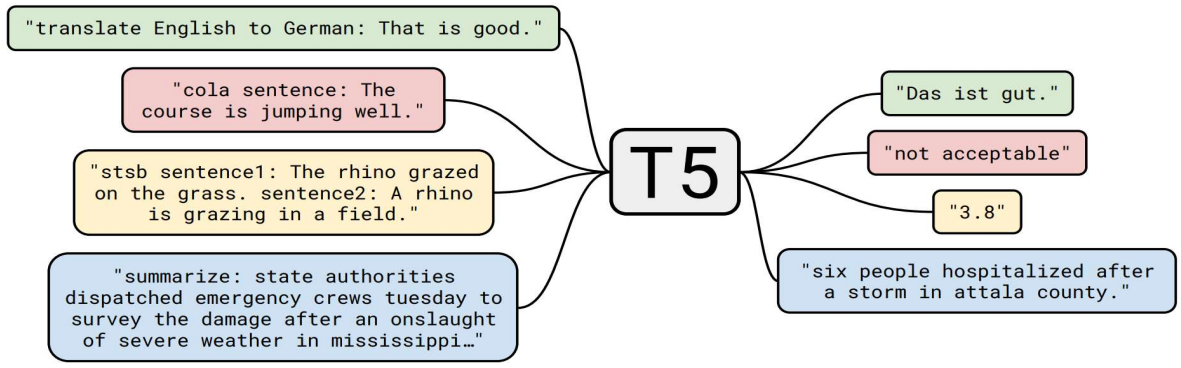


Fig. 3 *Diagram of the text-to-text framework of T5 transformer*

After pre-processing the data, the next steps in fine-tuning the model were as follows:

- Define the model and define the training arguments (see Figure 4). The Trainer only requires one argument, that is, the directory where the trained model will be saved; the rest can be left as default. But for our model fine-tuning, we used the following hyperparameters: learning rate, training batch size, evaluation batch size, optimizer, and the number of epochs.
- After the model and the training arguments are defined, they are passed to the Trainer which is then fine-tuned using the train method.
- Once the fine tuning is done, the model is saved on the Hugging Face hub, from where it can be accessed by anyone without having to fine-tune the model again.

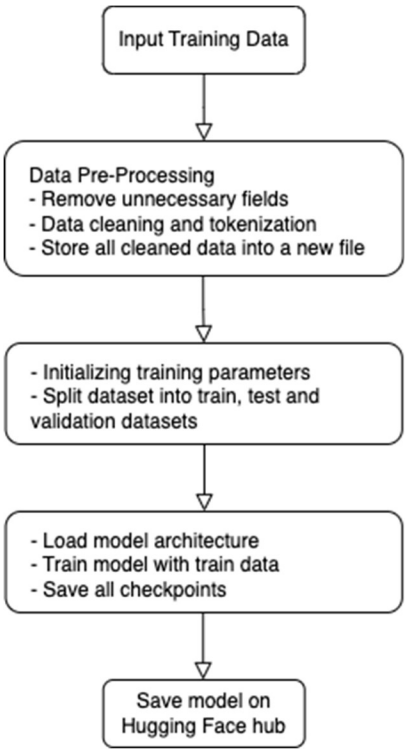


Fig. 4 *Steps followed while fine-tuning the transformer*

Streamlit News Summarization Application

Users can leverage Streamlit to create dynamic websites that take user inputs, process and carry out operations on a dynamic website, and have data science and machine learning capabilities. Furthermore, Streamlit's fundamentals are demonstrated through code while tackling NLP tasks, including Text processing and Text Summarization. In order to make the product more user-friendly and engaging, Streamlit includes a number of interactive features like sliders, radio buttons, text boxes, and buttons. ‘News Shack’ consists of various tabs and buttons which helps the user to select the desired news and categories of the news. In addition to assisting the user in locating the needed news, our application also enables the user to choose the quantity of news articles from CNN or GoogleNews that they wish to read. Additionally, the application's output provides a picture of the news incident along with the date and time it was published, as well as a synopsis of the news and a URL to the original news page if the user wants to read it later.

Results and Findings

Outcomes

We started by developing a list of keywords used across the news corpus, which provided some ideas about the source data keywords. We removed stop words to avoid any trivial conjunctions, prepositions, etc. After thoroughly cleaning the text for both the human-generated summary and the actual text, we generated the top 20 words for the news text and the summary (see Figure 5).

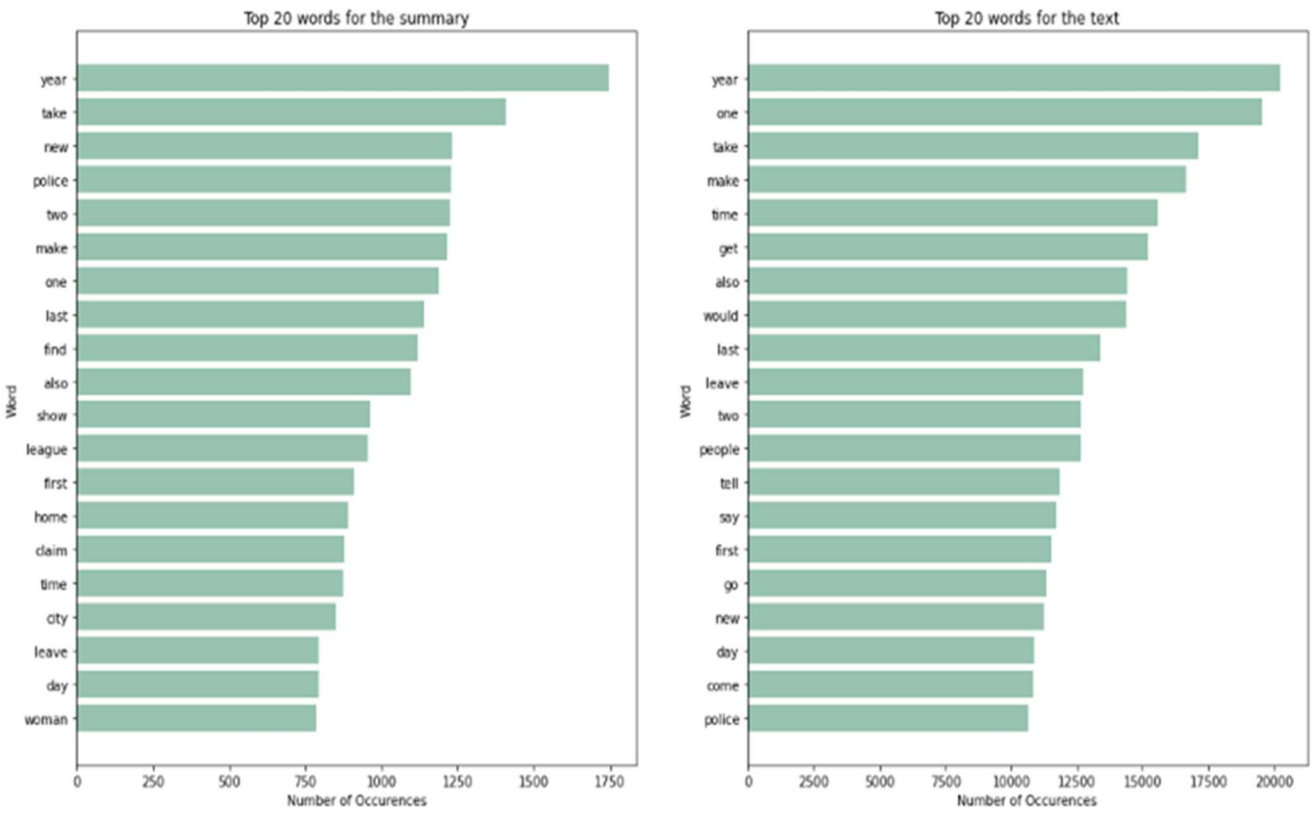


Fig. 5a and 5b *Top 20 Words for CNN/Daily Mail dataset (respectively summary and text)*

After reviewing the dataset, we determined that 'Year' and 'Take' represented the most frequent words in both cases. To plot a better visualization of this cluster of words we created a word cloud, which is a graphical representation that allows the viewer to form a quick intuitive sense of the text. In the below image, while preserving the anonymity of the subject, we can immediately see the common trends which the corpus follows (see Figure 6).

Figure. 6a and 6b *Word Clouds for CNN/Daily Mail dataset*

To further our research, we used part-of-speech tagging to understand the types of words used across the corpus. This requires first converting all the text strings to Text Blobs and calling the POS tags method on each, yielding a list of tag words for each news. The results are shown in Figure 7.

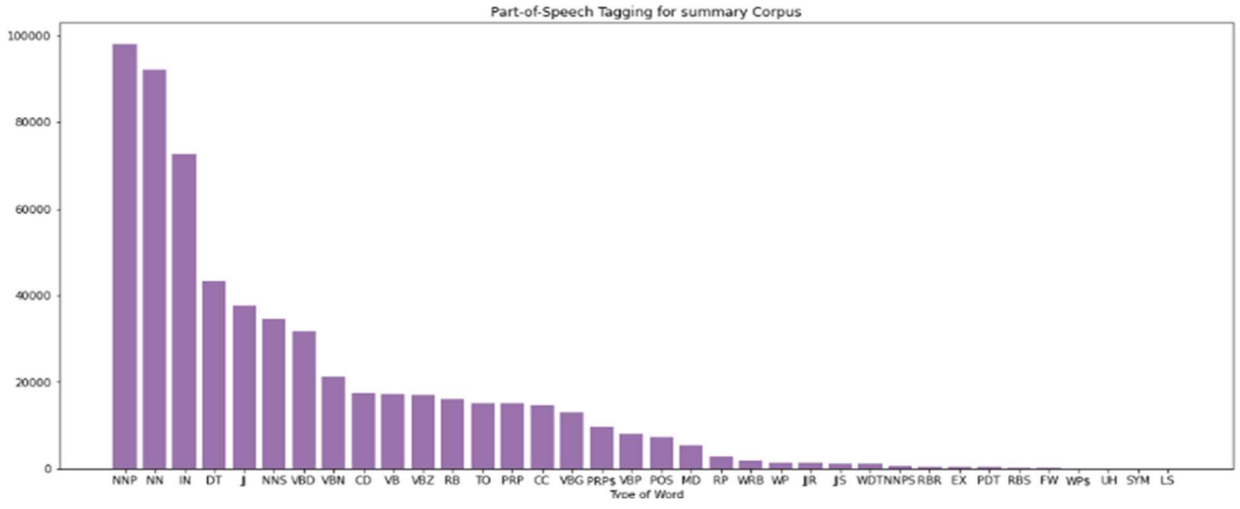


Fig. 7 *Part-of-Speech Tagging for the Corpus*

The practice of POS tagging (Parts of Speech Tagging) also known as grammatical tagging involves marking up the text for a specific section of a speech depending on its meaning and context. It aimes to give each word a particular token (a part of speech).

A popular topic modeling approach to extract themes from a corpus is Latent Dirichlet Allocation (LDA). Latent refers to the underlying meaning of a text. Following the Dirichlet distribution and method comes the Dirichlet allocation. In our case, the only feature pre-processing which is necessary is feature construction before we can represent the sample of text in a manageable feature space. This only entails transforming every string into a number vector. Utilizing the CountVectorizer object from sklearn, which produces a n x K document-term matrix makes this possible where K is the number of different words across the n news in our sample. Here, we have generated ten topics from the corpus which are illustrated in the below bar chart (see Figure 8).

According to our analysis, we have seen that most of the news in the corpus was related to the topics such as “New Year” and “Soccer Matches”. As we can see in the above bar chart there are over 4000 summaries that talk about New year and about 2000 summaries that talk about Soccer Matches.

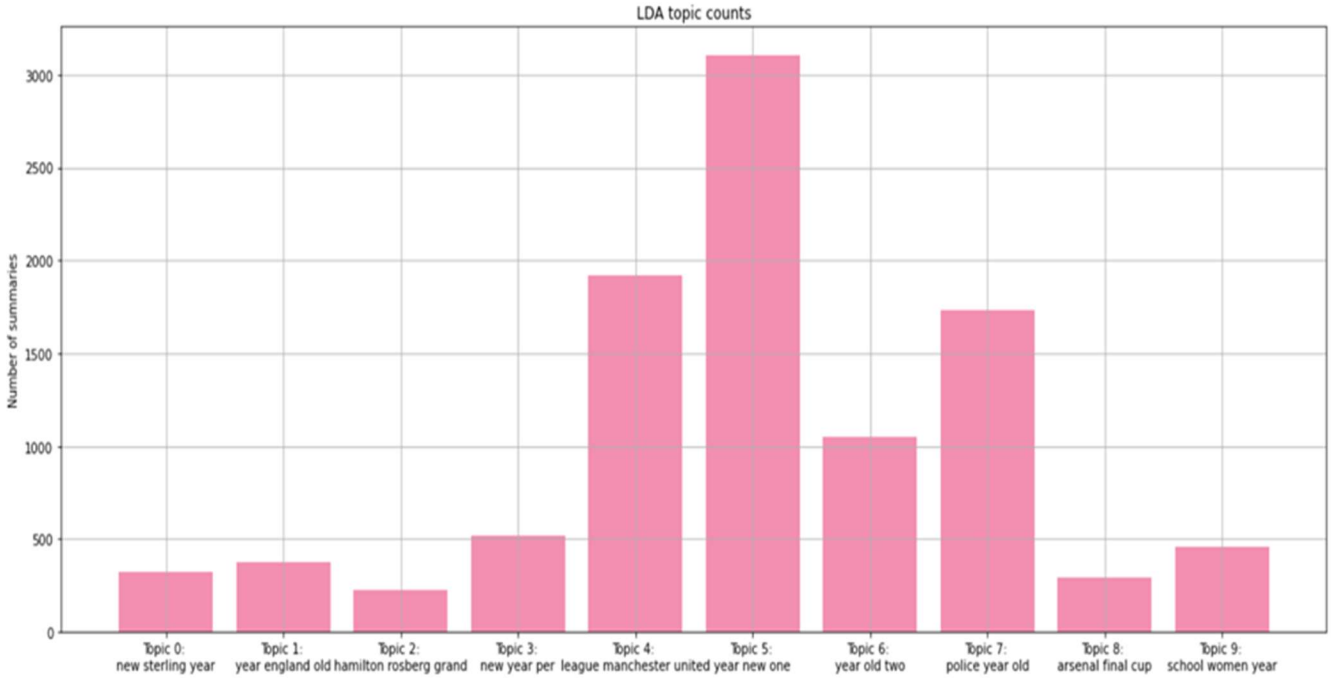


Fig. 8 *Topics generated through LDA for CNN/Daily News Dataset*

Summarization Results

We have fine-tuned the pre-trained T5 model on the CNN/Daily Mail news dataset. The CNN/Daily Mail news dataset consists of news text and human-generated summaries which are written by humans. These human-generated summaries were used for analyzing the summaries generated by our model. Table 4 shows the summary obtained for one of the news articles

Table 5 *Comparison of human-generated summary and T5-generated summary*

Source of summary	Summary
Summary generated by a human	Simpson is serving a 33-year term for robbery, kidnapping and assault. He's scheduled to be back in court on Monday to seek a new trial. He says he got poor advice from his lawyer. Prosecutors say there's no merit to the claim.
Summary generated by T5 model	O.J. Simpson is scheduled to return to a Las Vegas courtroom Monday. He's arguing bad legal advice led to his arrest and conviction in 2007 confrontation with sports memorabilia dealers, his new lawyers say. The former Buffalo Bills halfback was convicted of leading associates into room and using threats, guns and force to take back items from the two dealers.

Table 6 *Evaluation of ROUGE score*

	ROUGE-1	**ROUGE-2**	**ROUGE-L**
Precision	0.306	0.280	0.193
Recall	0.422	0.378	0.266
F-measure	0.355	0.341	0.224

Evaluating summarization tasks is difficult as there are many criteria such as information satisfaction, coverage, fluency, and concision. Table 5 presents the ROUGE score for the summary mentioned in table 4. ROUGE (Recall Oriented Understudy for Gisting Evaluation) is a set of metrics used for the evaluation of automatic text summarization and machine translations. It compares automatically generated summary with reference summary or multiple reference summaries. There are five evaluation metrics out of which we have used ROUGE-1,

ROUGE-2 and ROUGE-L for evaluating our summary. The metrics mentioned above can be interpreted as follows:

1. ROUGE Recall: 42.2 percent means that 42.2 percent of the n-grams in the reference summary is also present in the summary generated by the T5 model.
2. ROUGE Precision: 30.6 percent means that 30.6 percent of the n-grams in the summary generated by the T5 model is also present in the reference summary.
3. ROUGE F-measure: It is a measure of a test's accuracy.

Conclusion

We have fine-tuned a pre-trained T5 transformer model on a large corpus of news articles for abstractive news text summarization. This fine-tuned model has been saved on the Hugging Face hub from where others can directly use it without the need to train it again. The performance of this model has been evaluated by computing the ROUGE score for the generated summaries. We also computed the Flesch Reading Ease test score of the generated summary to ensure that we have generated easy-to-understand summaries for the user.

Our application allows users to read summaries of the latest news articles. It also allows users to filter news summaries of a category of their choice. Furthermore, our application has a feature that makes it possible for users to paste news article text as input and generate summaries at the click of a button. In the future, this Streamlit application can be developed further into a mobile application. Eventually, it can be developed to generate summaries where users directly enter the URL of the news article instead of pasting the text as input.

References

Blekanov, I. S., Tarasov, N., & Bodrunova, S. S. (2022). Transformer-Based Abstractive Summarization for Reddit and Twitter: Single Posts vs. Comment Pools in Three Languages. *Future Internet*, *14*(3), 69.

Garg, A., Adusumilli, S., Yenneti, S., Badal, T., Garg, D., Pandey, V., ... & Agarwal, R. (2020, December). NEWS Article Summarization with Pretrained Transformer. In *International Advanced Computing Conference* (pp. 203-211). Springer, Singapore.

Gupta, A., Chugh, D., Anjum, Katarya, R. (2022). Automated News Summarization Using Transformers. In: Aurelia, S., Hiremath, S.S., Subramanian, K., Biswas, S.K. (eds) Sustainable Advanced Computing. Lecture Notes in Electrical Engineering, vol 840. Springer, Singapore. https://doi.org/10.1007/978-981-16-9012-9_21

H. Batra et al., "CoVShorts: News Summarization Application Based on Deep NLP Transformers for SARS-CoV-2," 2021 9th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), 2021, pp. 1-6, doi: 10.1109/ICRITO51393.2021.9596520.

P. Sethi, S. Sonawane, S. Khanwalker and R. B. Keskar, "Automatic text summarization of news articles," 2017 International Conference on Big Data, IoT and Data Science (BID), 2017, pp. 23-29, doi: 10.1109/BID.2017.8336568.

Ramesh, G.S., Vamsi Manyam, Vijoosh Mandula, Pavan Myana, Sathvika Macha, Suprith Reddy (2022). Abstractive Text Summarization Using T5 Architecture. In: Reddy, A.B., Kiranmayee, B., Mukkamala, R.R., Srujan Raju, K. (eds) Proceedings of Second

International Conference on Advances in Computer Engineering and Communication Systems. Algorithms for Intelligent Systems. Springer, Singapore. https://doi.org/10.1007/978-981-16-7389-4_52