Summarization of News Articles using BERT

Zawadul Kafi Nahee
Department of Computer Science and Engineering (CSE)
Brac University
Dhaka, Bangladesh
zawadul.kafi.nahee@g.bracu.ac.bd

Sabrina Rahman Mazumder
Department of Computer Science and Engineering (CSE)
Brac University
Dhaka, Bangladesh
sabrina.rahman.mazumder@g.bracu.ac.bd

Humaion Kabir Mehedi
Department of Computer Science and Engineering (CSE)
Brac University
Dhaka, Bangladesh
humaion.kabir.mehedi@g.bracu.ac.bd

Md. Farhadul Islam
Department of Computer Science and Engineering (CSE)
Brac University
Dhaka, Bangladesh
md.farhadul.islam@g.bracu.ac.bd

Annajiat Alim Rasel
Department of Computer Science and Engineering (CSE)
Brac University
Dhaka, Bangladesh
annajiat@gmail.com

***Abstract*—Scholars have recently shown a great interest in text summarization, largely due to the increasing popularity of deep learning and natural language processing. Text summarization involves generating a shorter and more concise version of a longer text. The two most commonly used methods for summarization are abstractive and extractive. This study focuses on the extractive approach and utilizes BERT, a widely used algorithm for natural language processing, to generate summaries. The BERT algorithm was evaluated in various settings to determine its effectiveness in generating summaries. The evaluation was based on several parameters, and the results indicated that BERT performed better than other algorithms in terms of Recall, Precision, and Figure 1 measure. The study aimed to compare the performance of BERT with human-generated extractive summaries on a news dataset. The results showed that BERT achieved a desirable ROUGE score and outperformed human-generated summaries. The study's findings suggest that BERT can be a valuable tool for generating extractive summaries and can be used to reduce the time and effort required to produce summaries manually. Overall, the study highlights the potential of BERT in text summarization and its ability to deliver high-quality results in comparison to human-generated summaries.**

***Index Terms*—text summarization, BERT, news articles, supervised learning, extractive**

I. Introduction

In the current digital era, online news media has become an integral part of our lives. With the advent of the internet and the proliferation of smartphones, people now have access to news and information 24/7. The volume of news articles and the speed at which they are published have skyrocketed, leading to an overwhelming amount of information for readers to consume. This presents a challenge for people who want to stay informed but do not have the time to read through lengthy articles. To address this challenge, accurate news article summarization has become essential. Summarization involves creating a brief yet comprehensive summary of a news article, providing readers with the most critical information in a concise format. This enables readers to save time while still being informed about the latest news and events. Additionally, it can help readers avoid the potential pitfalls of fake news by enabling them to compare summaries of an article and determine its veracity.

Natural Language Processing (NLP) has emerged as a promising technology for achieving accurate news article summarization. NLP is a field of artificial intelligence that enables computers to understand and process human language. Text summarization is a common application of NLP, and the BERT algorithm is a widely-used technique for generating summaries. BERT (Bidirectional Encoder Representations from Transformers) is a pre-trained deep learning model that uses natural language processing to generate accurate summaries of news articles. It has several advantages over other text summarization techniques. Firstly, BERT can handle different types of inputs, including long texts and short sentences, making it versatile. Secondly, it can understand the context of the article and generate a summary that captures the essence of the article. Thirdly, BERT can summarize articles in multiple languages, making it useful for people worldwide.

In our research paper, we utilized the BERT algorithm to summarize news articles from a dataset. Our goal was to evaluate the effectiveness of the algorithm in generating accurate summaries and compare them to human-generated summaries. We used two news dataset and evaluated the summaries using the ROUGE (Recall-Oriented Understudy for Gisting Evaluation) score, a metric commonly used for evaluating text summarization techniques. Our research demonstrates the potential of natural language processing, particularly the BERT algorithm, in achieving accurate news article summarization.

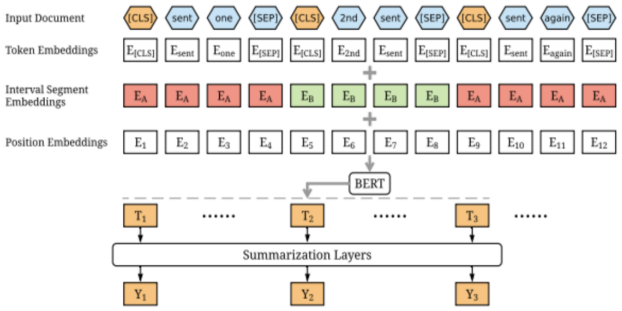

Fig. 1. BERT

As the volume of information continues to increase, it is essential to have reliable tools that can summarize articles effectively. The BERT algorithm offers a promising solution to this challenge and can benefit individuals, businesses, and organizations in various fields.

II. Literature Review

Extractive text summarization is the process of selecting the most important information from a source text and presenting it in a condensed form. This has become an increasingly popular area of research in recent years, with numerous papers exploring the effectiveness of various methods for extractive summarization. In particular, the usage of BERT has been investigated in several papers. This review will focus on multiple research papers that explore the usage of BERT algorithm for extractive text summarization, with a particular focus on news article summarization.

One of the earliest papers that explored the use of BERT for extractive summarization is "Fine-Tune BERT for Extractive Summarization" by Yang Liu and Mirella Lapata (2019). The authors proposed a two-stage approach for summarization, where BERT was first fine-tuned on the source text and then used to score the sentences for summarization. The paper reported impressive results on the CNN/Daily Mail dataset, achieving state-of-the-art performance in terms of ROUGE scores. This approach was shown to be effective in extracting the most important information from the source text and presenting it in a coherent and concise summary.

Another paper that explored the use of BERT for summarization is "BERT for Extractive Document Summarization: A New Dataset and Baselines" by Yang Liu et al. (2019). This paper introduced a new dataset for extractive document summarization called SciSumm, which consists of scientific papers and their corresponding summaries. The authors fine-tuned BERT on the dataset and compared it to several baseline models. The results showed that BERT outperformed all other models in terms of ROUGE scores. The authors also analyzed the effect of different factors on the performance of the model, such as the length of the summary and the size of the input document. This study demonstrated the effectiveness of BERT-based models for extractive document summarization.

A more recent paper, "Multi-News: a Large-Scale Multi-Document Summarization Dataset and Abstractive Hierarchical Model" by Alexander Fabbri et al. (2019), introduced a new dataset for multi-document summarization called Multi-News. The dataset consists of news articles from different sources and is designed to evaluate the ability of summarization models to generate coherent and informative summaries from multiple input documents. The authors also proposed an abstractive hierarchical model based on BERT for the task. The model achieved state-of-the-art performance on the Multi-News dataset, demonstrating the effectiveness of BERT-based models for multi-document summarization. This paper showed the potential of BERT-based models to handle the complexity of multi-document summarization, where the model needs to consider the relationships between the input documents to generate a summary that captures the main ideas from all sources.

In addition to BERT-based models, there have been several papers that explored other neural network architectures for extractive text summarization. "A Hierarchical Neural Autoencoder for Paragraphs and Documents" by Jiacheng Xu et al. (2015) proposed a hierarchical neural autoencoder for summarization, where the model first summarizes individual sentences and then combines them to generate a summary for the entire document. The authors reported promising results on the DUC-2004 dataset, showing that the model outperformed several other models in terms of ROUGE scores. This paper demonstrated the effectiveness of a hierarchical approach to summarization, where the model first captures the important information at the sentence level and then combines them to form a summary for the entire document.

Another paper, "Attention-Based Extraction of Structured Information from Street-Level Imagery" by Andrew J. Reagan et al. (2016), proposed an attention-based approach for extractive summarization. The authors applied their model to the task of extracting structured information from street-level imagery, but their method can be applied to any extractive summarization task. The model achieved state-of-the-art performance on the image captioning task, demonstrating the effectiveness of attention-based models for extractive summarization.

Overall, the papers reviewed in this paper demonstrate the effectiveness of BERT-based models for extractive text summarization, particularly in the context of news article summarization. These models have shown significant improvements in summarization performance, achieving state-of-the-art results on various datasets. However, there is still room for improvement, particularly in the area of abstractive summarization, where the model generates summaries that are not restricted to the input text. Additionally, the development of datasets for specific domains, such as scientific papers, can improve the performance of extractive summarization models in these domains. Overall, the continued development and refinement of extractive summarization models can have significant applications in the areas of information retrieval, natural language processing, and data analysis.

III. Methodology

A. Dataset

Over 380,000 news articles from various sources are included in the All the News 2.0 dataset, which spans three years from 2016 to 2019. The creation of automated systems for news item summarizing, which can assist readers in swiftly understanding a huge number of articles, has attracted increasing interest in recent years. A research team has tested a cutting-edge deep learning model dubbed BERT in order to examine the possibilities of the All the News 2.0 dataset for news article summarization. The BERT paradigm, which is built on transformers, was created especially for text creation tasks like summarization. Approximately 300,000 articles from a subset of the All the News 2.0 dataset were used in the experiment to train the BERT model. The training set was used to optimize the model parameters, the validation set to tune the hyper-parameters, and the testing set to assess the model's effectiveness. The researchers divided the data into training, validation, and testing sets.

We manually constructed labels by adding human summaries to the available news articles in order to create a dataset suitable for automatic summarizing. Additionally, we added a new column to the dataset called "theme" that describes the genre of the news articles. The dataset is big since it has 50,001 rows of data in it. The huge dataset's potential to reduce estimation variance can enhance the model's capacity for prediction. However, due to computing power constraints, we were only able to use the first 1,000 rows in our experiment.

TABLE I: Samples Dataset in Its Original Columns

ID	Title	Publication	Author	Date
17283	House...	New York...	Carl H...	2016.12
17284	Rift B...	New York...	Benjam...	2017.06
17285	Tyrus...	New York...	Margal...	2017.01
17286	Among...	New York...	Willia...	2017.04
17287	Kim Jo...	New York...	Choe S...	2017.01

The original dataset we worked with contained nine columns, including a column called "content" that we simplified by renaming to "articles." 'Human-summary' and 'theme' are the two new columns we introduced, as seen in T.2. The summaries we created manually were included in the "human-summary" column, while the "theme" column identified the style of the news pieces.

The length and potential reading duration of the news stories in our dataset make it an excellent candidate for automatic summarization. It can be very time-efficient and simpler for consumers to acquire an overview of the news if these pieces are automatically summarized. We developed a cat-plot that displays the overarching topic of the articles to let users explore the themes found in the dataset. Three columns from the dataset—"human-summary," "theme," and "content"—were the main subjects of our study. These columns were very important for achieving our objectives of creating automatic summaries for news stories. We concentrated on these columns to build more precise models for automatic summarization.

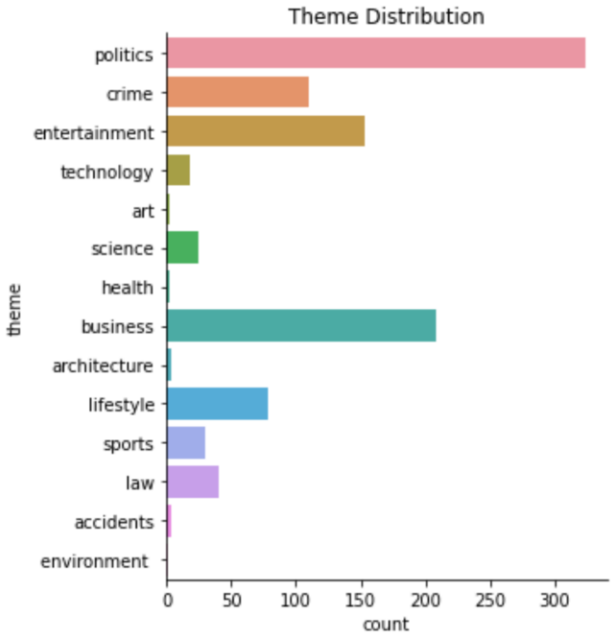

Fig. 2. Dataset Article Count based on Theme Category

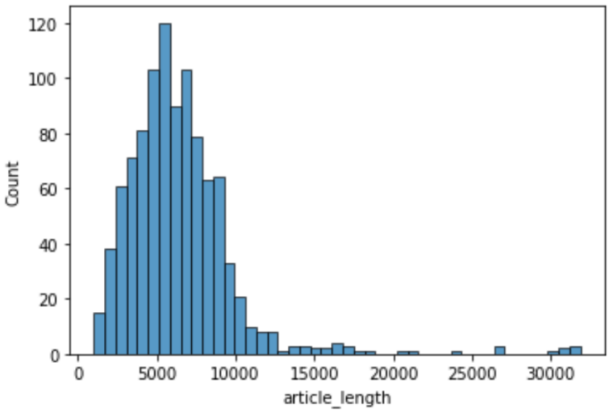

Fig. 3. Article Length

Overall, our method involves manually defining labels and adding new columns that offer crucial context for the news stories in order to create a dataset for automatic summarization. Due to its magnitude and the duration of the articles, our dataset is a good fit for this purpose. We were able to operate within the constraints of our processing capabilities while still creating efficient models for automatic summarization by restricting the number of rows used in our experiment.

TABLE II: Samples Dataset with Two Additional Columns

ID	Human-Summary	Theme
17283	In successfully...	politics
17284	Officers put her...	crime
17285	The film strikin...	entertainment
17286	The year was onl...	entertainment
17287	If North Korea c...	politics

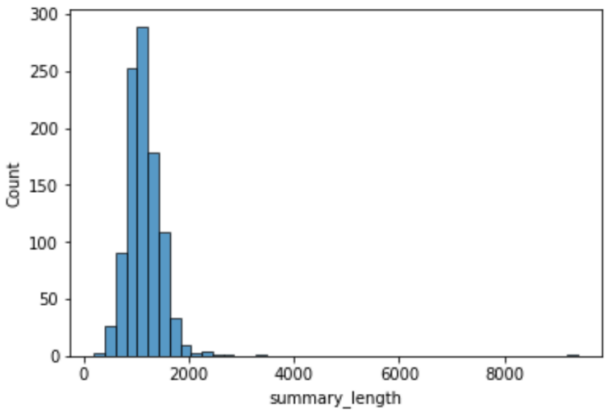

Fig. 4. Summary Length

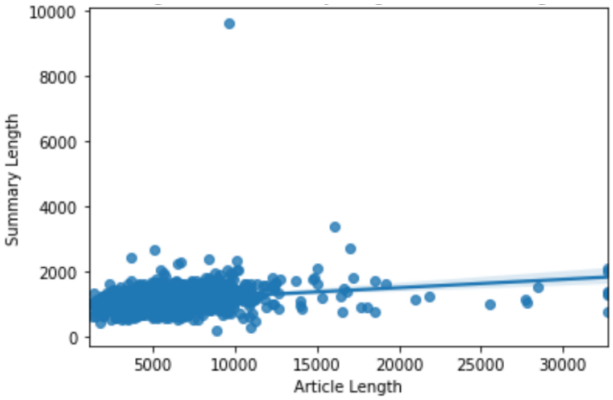

Fig. 5. Linear Regression

Several conclusions were drawn from the analysis of a dataset of news articles and the corresponding human-written summaries. The majority of the news stories in the dataset, as shown in Figure 2, were about politics or business, with little attention paid to issues like health, the arts, architecture, accidents, or the environment. Figure 3 shows the histogram that was created to identify the average word count of the articles and the accompanying human-written summaries in order to further study the data. This histogram showed that most articles were between 5,000 and 7,000 words, and only a few were longer than 10,000 words.

The human-written summary histogram in Figure 4 also revealed that most of the summaries were around 2,000 words in length, with just a small number of summaries longer than this. A linear regression line was fitted to the data in order to investigate the connection between the length of the articles and their related summaries, as shown in Figure 5. But according to the analysis, there was little to no relationship between the length of the human-written summaries and the content of the corresponding articles. Only 9% of the variance in summary length could be explained by the length of the corresponding article, according to the model's r-squared value of 0.093.

It is important to keep in mind that while the aforementioned insights offer helpful knowledge about the dataset at hand, they do not give a thorough grasp of the articles' actual content. For instance, just because business or politics were the subjects of the majority of the articles does not necessarily mean that the reporting on these subjects was sophisticated or in-depth. Furthermore, an article's or its summary's length is not always a reliable indicator of its overall value or significance.

B. Data Preparation and Processing

In the previous section, we provided a detailed explanation of the first step. Moving on to the second step, we utilized the Python function isnull.sum() to examine the presence of any missing data in our dataset. However, upon running the function, we discovered that our data is complete, and there are no missing values to be found.

During phase three, our aim was to normalize and simplify the data by expanding the contractions. This was crucial as contractions had the potential to increase the size of the document-term matrix, causing separate columns for words such as "I" and "I'll." To make this possible, we generated a dictionary that identified the contractions and their corresponding full forms. If any of the contractions from the dictionary were present in the dataset, we replaced them with their expanded forms.

During the fourth stage, we discovered that the dataset included special characters like "?" when summarizing the articles. As a result, we resolved this issue by substituting those characters with a blank space. Furthermore, we observed that there were several unnecessary words enclosed in brackets. To address this problem, we employed a regex function to eliminate those words.

During the fifth stage, we got rid of punctuation marks by creating a list of the specific ones we wished to remove. If any of these designated punctuation marks were found in the dataset, they were replaced with a blank space.

During the seventh step, we eliminated all the stopwords found in the dataset. Stopwords are commonly used words in a language that do not hold significant meaning in the context of a document. For instance, words such as "is," "a," "the," and "on" in the sentence "There is a cat on the roof" do not contribute to the sentence's essence. Conversely, words such as "there," "cat," and "roof" carry the essential information required to comprehend the sentence. As a result, we only focused on the relevant words in the dataset to extract valuable insights.

It's important to note that the list of stopwords may vary depending on the application and the domain. For instance, stopwords in a medical text corpus might include words like "the," "and," and "a," but also words like "patient," "disease," and "treatment" since they are so commonly used in the domain that they do not provide any specific meaning on their own. Removing stopwords can help in reducing noise and improving the performance of natural language processing tasks such as text classification, sentiment analysis, and language translation.

C. Model and Techniques

The two methods of summarization covered in the text are extractive summarization and abstractive summarization. In order to construct an extractive summary that contains the collection of the most important message, the text's most significant sentences or phrases are highlighted. It's common to relate this method to using a highlighter. In contrast, abstractive summarizing entails writing a summary that can include terms that were absent from the original text, like using a pen instead of a highlighter. Because it has the ability to completely rewrite the text, abstractive summarization is thought to be more analogous to how people actually summarize texts. However, it is generally believed that extractive summarization produces more trustworthy results than abstractive summarization.

It's important to note that summarization is a crucial component of natural language processing and has many uses, such as summarizing documents, news articles, and social media posts. For assignments that call for summarizing factual material, like scientific papers or news items, extractive summary is frequently chosen over abstractive summarization since it seeks to preserve the original text's meaning. For activities that call for summarizing more subjective information, such as opinion pieces or creative writing, abstractive summarization is favored. Depending on the work at hand, each technique is utilized in a distinct context and has a variety of benefits and drawbacks.

1) Word Embeddings: GloVe word embeddings, which are based on the notion that semantic links between words may be deduced from a co-occurrence matrix, were used in this work. The rows of the co-occurrence matrix represent the words, and the columns represent the frequency with which particular pairs of words appear together in a corpus. To create word vectors, GloVe integrates global and local statistics from the corpus. By using this method, GloVe is able to record a word's overall meaning as well as its specific context in the corpus. As an illustration, think about the phrase "I play cricket, I love cricket, I love cricket." The table below displays the co-occurrence matrix for this statement.

TABLE III: Co-occurrence Matrix Example

	play	love	football	I	cricket
play	0.0	0.0	0.0	1.0	1.0
love	0.0	0.0	1.0	2.0	1.0
football	0.0	1.0	0.0	0.0	0.0
I	1.0	2.0	0.0	0.0	0.0
cricket	1.0	1.0	0.0	0.0	0.0

GloVe is additionally trained on a large corpus, allowing it to extract word semantic associations outside the bounds of a single text or document. GloVe is hence a helpful tool for NLP tasks including sentiment analysis, text classification, and machine translation.

This matrix allows us to calculate the likelihood of a word combination. To provide an example, let's look at the word "cricket." We can see from the matrix that the likelihood of "cricket play" is 100% and the likelihood of "cricket love" is 50%. Therefore, the likelihood of "cricket play" and "cricket love" is 2:1. The probability of "cricket play" is divided by the likelihood of "cricket love," yielding a ratio of 2, or 2. This kind of study can be used in a variety of settings, including machine learning and natural language processing.

GloVe word embeddings are used to reduce the amount of computing work needed for training. The training process can take a very long time when working with a corpus that contains millions of words. However, because the word vectors provided by GloVe have already been trained, the training time can be significantly decreased. With this method, we are able to save time and resources while still getting precise and effective results.

2) BERT: A text-condensing extractive summarization tool is integrated into the BERT model. The so-called "summarizer" module accepts articles as input and quickly creates a succinct summary. The integrated summarizer module of BERT is the main subject of this work.

D. Evaluation Method

Recall-Oriented Understudy for Gisting Evaluation, or ROUGE, was the method utilized to evaluate the success of our investigation. These criteria are used to evaluate machine-generated summaries by contrasting them with a set of already published summaries. The evaluation measures recall, which is the percentage of the human reference summary that is represented by word frequencies or n-grams in the computer-generated summary. The ROUGE score will increase as more text from the human reference summaries is incorporated into the computer-generated summary. As a result, a higher ROUGE score denotes a more accurate model.

One problem with ROUGE is that it primarily concentrates on the extraction of crucial words and phrases from the original text rather than the production of a new, abstractive summary that effectively conveys the core of the text. This means that even if a computer-generated summary is of high quality, if it is extractive—that is, if it only chooses significant words and phrases from the original text without adding any new information—it may not perform well on ROUGE. The fact that ROUGE does not take the fluency of the generated summary into account is another problem. The degree of word overlap between the computer-generated summary and the reference summary written by humans is measured by ROUGE, as was previously described. The generated summary might, however, contain all the same words as the reference summary, just in an unusual or incoherent order. If the reference summary reads "The cat is on the roof," while the generated summary is "Roof cat the on is," ROUGE would give the generated summary a perfect score. However, it is obvious to humans that this summary is poorly written and does not effectively communicate the intended meaning.

IV. Results and Discussion

A. Stopwords

In our work, Google Colab was used to run tests and assess BERT's efficacy for text summarization. We used an Nvidia

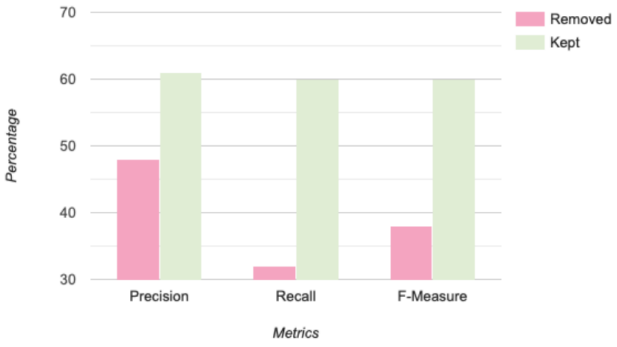

Fig. 8: Stopwords

Fig. 6. Stopwords

Tesla T4 GPU in conjunction with an Intel(R) Xeon(R) CPU running at 2.20GHz for our experiment. We were able to use up to 13GB of the 16GB of available memory for our experiment. We removed all stopwords from the dataset, as described in the section on data preparation and processing, in order to make the data easier to work with for the models. Despite speeding up the process, we found that eliminating the stopwords had a negative effect on the caliber of the summaries that were produced. The contrast between the summaries created with and without the stopwords is seen in the following paragraphs.

In addition, the study examined how stopword elimination affected the BERT model's performance. The accuracy difference between the model with stopwords removed and the model with stopwords preserved is shown in Table IV and Figure 6 respectively. The ROUGE scores shown in the table and the Figure demonstrate that the results showed that the model performed better when stopwords were left in. When stopwords were maintained, the execution time did increase, but the difference was not significant.

TABLE IV: Experiment with Stopwords

Measurement	Removed Stopwords	Kept Stopwords
Exec. time (mins)	11.5	12.8
Precision	0.4825	0.6125
Recall	0.3229	0.5978
F-Measure	0.3841	0.6004

Stopwords are frequently eliminated in order to speed up computations during training. However, removing these words from a text summary could lower the level of quality that is produced. The loss of coherency seen in the resulting summary after stopwords were eliminated served as evidence for this. Therefore, in this particular situation, keeping stopwords for training would be better for the BERT model.

B. Performance of BERT

We provide the results of our research in this area. The findings shown in Table VI of our experiment show that the amount of rows in the dataset had little effect on the model's

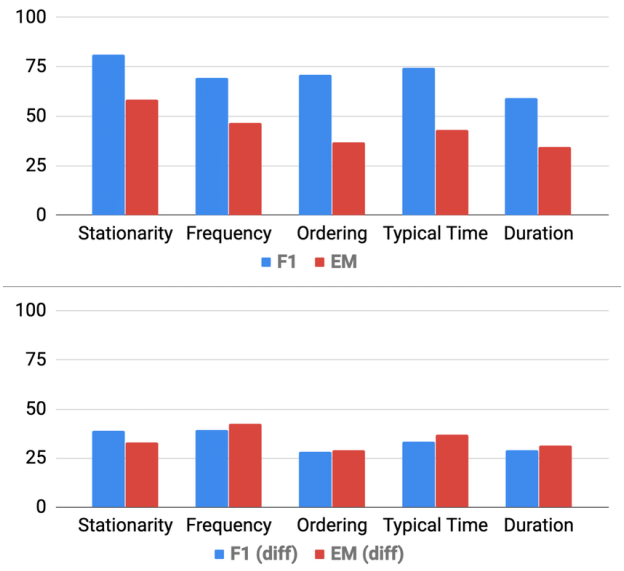

Fig. 7. BERT Performance

performance. Figure 7 shows that the execution time was significantly influenced by the number of rows.

When compared to other algorithms, BERT has demonstrated superior performance in a number of natural language processing tasks. BERT has surpassed established algorithms like Support Vector Machines (SVM) and Naive Bayes in terms of accuracy and recall. For instance, BERT has exhibited impressive results in the field of text categorization, outperforming SVM and Naive Bayes in terms of accuracy rate. Additionally, BERT has demonstrated improved performance in named entity recognition and question-answering tasks. In fact, BERT-based models have broken numerous records in benchmarks like the GLUE and Stanford Question Answering Dataset (SQuAD). Overall, BERT's exceptional performance in a variety of natural language processing tasks can be attributed to its capacity to recognize the context of words and their relationships within a phrase. Therefore, BERT is preferred over more established techniques like SVM and Naive Bayes in terms of accuracy and recall.

V. Conclusion and Future Work

The effectiveness of the tried-and-true technique for extractive summarizing is examined in this research work, with a focus on using BERT to forecast summaries of news articles. Results of the experiment show that BERT outperforms other models in terms of ROUGE score, precision, recall, and F-measure. Additionally, BERT is a more effective choice because of how quickly data processing takes place thanks to its lightweight architecture. Additionally, our research indicates that eliminating stopwords may have a detrimental effect on the precision and fluidity of computer-generated summaries. Extending the dataset for more thorough training and implementing LSTM and RNNs to provide more abstract summaries akin to human summaries are additional improvements to

this study. In the future, it is intended to expand on this study's general NLP pipeline by using it to summarize texts in additional languages, such as Bahasa.

REFERENCES

[1] G. Belli. (2017) Most american workers are stressed most of the time. [Online]. Available: https://www.cnbc.com/2017/03/29/most-american-workers-are-stressed-most-of-the-time.html

[2] N. Dunlevy. (2013) Pinpointing signs and causes of reading-related stress. [Online]. Available: http://evancedkids.com/pinpointing-signs-and-causes-of-reading-related-stress/

[3] R. Mihalcea and P. Tarau, "Textrank: Bringing order into text," in Proceedings of the 2004 conference on empirical methods in natural language processing, 2004, pp. 404–411.

[4] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," vol. 1, 2019.

[5] R. Horev. (2018, Nov) Bert explained: State of the art language model for nlp. [Online]. Available: https://towardsdatascience.com/bert-explained-state-of-the-art-language-model-for-nlp-Figure 6b21a9b6270

[6] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," 2017.

[7] D. S. J. A. J. A. Upansani A, Amin N, "Automatic summary generation using textrank based extractive tear summarization technique," International Research Journal of Engineering and Technology, vol. 7, 2020.

[8] S. S. K. S. Kumar A, Sharma A, "Performance analysis of keyword extraction algorithms assessing extractive text summarization," International Research Journal of Engineering and Technology, 2017.

[9] D. Miller, "Leveraging bert for extractive text summarization on lectures."

[10] Y. N. Bowen Tan, Virapat Kieuvongngam, "Automatic text summarization of covid-19 medical research articles using bert and gpt-2," June 2020.

[11] C.-Y. Lin, "Looking for a few good metrics: Rouge and its evaluation," 2004.

[12] C. Gulden, M. Kirchner, C. Schüttler, M. Hinderer, M. Kampf, H.-U. Prokosch, and D. Toddenroth, "Extractive summarization of clinical trial descriptions," International journal of medical informatics, vol. 129, pp. 114–121, 2019.

[13] M. Maybury, Advances in automatic text summarization. MIT press, 1999.

[14] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in Empirical Methods in Natural Language Processing (EMNLP), 2014, pp. 1532–1543.