iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 12 **Issue:** VI **Month of publication:** June 2024

DOI: https://doi.org/10.22214/ijraset.2024.63358

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 12 Issue VI June 2024- Available at www.ijraset.com

Advanced NLP Techniques for Sentiment Analysis and Text Summarization Using RNNs and Transformers

Kumar Pritam[1], Dr. Abha Chaubey[2], Dr. Avinash Dhole[3]
Computer Science Department, Shri Shankaracharya Technical Campus, Bhilai

Abstract*: This research focuses on leveraging artificial intelligence and neural network architectures to enhance the capability of machines in comprehending, interpreting, and summarizing text data in human languages. The study aims to improve natural language processing (NLP) tasks, specifically sentiment classification and text summarization. Key efforts include the development of neural network architectures such as Recurrent Neural Networks (RNNs) and Transformers to model linguistic contexts and sequences. The creation of annotated datasets for sentiment analysis and summarization was essential for training and evaluating these models. Additionally, transfer learning techniques were explored to pretrain language models on large corpora, enhancing their performance. The evaluation of neural network models utilized relevant NLP metrics like accuracy, ROC curve, and F1 score for sentiment classification tasks. The research also developed end-to-end NLP pipelines leveraging trained neural networks for document summarization and sentiment detection. The results confirmed that AI and neural networks could effectively perform sentiment analysis and text summarization. Training metrics indicated robust learning and generalization capabilities, with high accuracy and improved ROUGE and BERT scores. The findings underscore the potential of deep neural networks in understanding and summarizing textual content, suggesting promising directions for future work, including deeper neural networks, attention models, and multimodal data integration.*
Keywords: NLP, Text summarization

I. INTRODUCTION

Natural Language Processing (NLP) is a subfield of artificial intelligence (AI) that focuses on the interaction between computers and human languages. It involves the development of algorithms and models that enable machines to understand, interpret, and generate human language in a way that is both meaningful and useful.

NLP encompasses a range of tasks from basic tokenization and part-of-speech tagging to complex activities like sentiment analysis, machine translation, and text summarization. Creating an end-to-end NLP pipeline involves several stages, including data preprocessing, model training, evaluation, and deployment.

For instance, in a sentiment analysis pipeline, raw text data is first cleaned and tokenized, then passed through a pretrained neural network model that has been fine-tuned on annotated sentiment datasets. The model outputs sentiment scores, which are then post-processed and used in downstream applications such as real-time sentiment monitoring or automated customer service. In text summarization, the pipeline might include steps for sentence segmentation, feature extraction, and the application of both extractive and abstractive summarization techniques, leveraging neural network models like Transformers. These models are evaluated against human-generated summaries using metrics such as ROUGE to ensure quality and coherence. Natural Language Processing (NLP) has evolved significantly since its inception in the 1950s, when initial efforts focused on machine translation and the development of rule-based systems. Early work, such as the Georgetown-IBM experiment in 1954, demonstrated the potential for computers to translate Russian sentences into English using a limited vocabulary and a set of grammatical rules. However, these rule-based systems were limited by their rigidity and the extensive manual effort required to create and maintain them. The 1980s and 1990s saw a paradigm shift towards statistical methods in NLP, driven by the availability of large corpora and increased computational power. Pioneering work in this era, such as Brown et al.'s statistical machine translation (SMT) model, leveraged probabilistic models to improve translation accuracy by learning from bilingual text corpora. This period also witnessed the development of key algorithms and techniques like Hidden Markov Models (HMMs) for part-of-speech tagging and n-gram models for language modeling.

II. LITERATURE REVIEW

Marcus et. al. [1] performed an analysis to evaluate the significance of logical reasoning and common sense within the domain of artificial intelligence. Tasks such as plan development, visual data analysis, logical reasoning application, and written information comprehension necessitate practical knowledge and reasoning abilities. The practical methods that can be utilized include web mining, crowdsourcing, logical analysis, and manual construction of extensive knowledge repositories. Having a thorough understanding of human logic is crucial, even though intelligent computers can perform these tasks without replicating human intelligence precisely.

Dabhi et. al. [2] Natural Language Processing (NLP) is a computational technique that utilizes software to analyze and interpret human language. Natural Language Processing (NLP) technologies possess the capability to automate a wide range of tasks. These professions require a diverse range of technical skills. These skills include the ability to translate text, extract and summarize data from complex databases, filter spam emails, identify false information, analyze political sentiments, conduct policy analysis, and utilize patient histories to provide effective healthcare. Gujarati is an Indian-origin language that is spoken by an estimated sixty million people globally. Currently, there are significant efforts being made to develop Natural Language Processing (NLP) tools and applications that are specifically tailored for Indian languages. This article presents a comprehensive analysis and thorough classification of the resources and processes used in developing components for Natural Language Processing (NLP) systems in the Gujarati language. An additional analysis was conducted after evaluating a number of commonly used open-source applications. Additional subjects for consideration encompass potential strategies for addressing the obstacles linked to the creation of components and resources for the Gujarati Natural Language Processing (NLP) system. The research provides valuable insights for academicians, professionals, and researchers interested in gaining a comprehensive understanding of the technical requirements, opportunities, and challenges related to the development of Gujarati natural language processing (NLP) systems.

Devlin et. al. [3] The BERT model was specifically designed to pre-train deep bidirectional representations from unlabeled text. This model differs from previous language representation models in that it undergoes simultaneous training on both the left and right context at all levels. The pre-trained BERT model can be enhanced by integrating an extra output layer, eliminating the need for substantial modifications to the task-specific structure. This enhancement enables the development of advanced models for various applications, such as language inference and query response. The BERT model demonstrates resilience and dependability in both theoretical and practical scenarios. The system demonstrates outstanding performance on eleven cutting-edge natural language processing tasks. The model significantly improves the accuracy of MultiNLI to 86.7%, the GLUE score to 80.5%, the SQuAD v1.1 question answering Test F1 to 93.2, and the SQuAD v2.0 Test F1 to 83.1. This improvement amounts to a 5.1 percentage point increase.

Anjaria et. al. [4] Sentiment analysis is a crucial process for deciphering unstructured data generated by social networking platforms. Sentiment analysis categorizes phrases in texts into various sentiment polarities, such as positive, neutral, negative, pleased, furious, repulsed, horrified, and others. This study utilized a text summarization method to analyze the polarity (positive, negative, neutral) of emotions conveyed in Bangla texts. An impressive accuracy rate of 98.33% is attained by utilizing a rule-based approach and manually designed features on the Bangla dataset. The research achieved a maximal accuracy of 98.33% in the classification of Bangla blog postings based on their sentiment polarity by utilizing a feature extraction method that relies on text summarization. The evaluation of sentiment in Bangla texts is conducted in this study using a rule-based algorithm and manually developed features. This study will provide benefits to researchers across various disciplines by enabling them to acquire new knowledge and improve their understanding. Consequently, researchers will be able to conduct more comprehensive investigations into issues related to sentiment analysis.

Ramaswamy et. al. [5] It is imperative for every organization to prioritize customer satisfaction and have a thorough understanding of consumer behavior in order to maintain its position in the market. It is imperative for firms to have a thorough understanding of the prevailing consumer attitudes in order to formulate more accurate and efficient product development and marketing strategies. Consumer evaluations can be acquired using various methods. The focus of our interest lies in unstructured data, specifically textual content derived from various sources such as social media, survey responses, audio recordings of customer discussions, and conversation transcripts. Accurate assessment of this data is crucial as it provides businesses with a significant competitive advantage and reveals information that encompasses a wide range of aspects, including product defects and purchasing trends. Enhancing the organization's capacity to gather information on consumer preferences, product enhancements, and marketing insights would yield substantial improvements. The objective of this study is to explore different Deep Learning and Natural Language Processing (NLP) methods in order to enhance the assessment of contextual data and gather customer feedback.

Stieglitz et. al. [6] The transmission of data via social networks has been made easier by social media, an innovative communication framework. Prior research has identified various factors related to content, consumers, and networks that may contribute to the dissemination of information. The research has largely overlooked the relationship between emotions and the dissemination of information in a social media context. The purpose of this investigation is to examine the possible correlation between the emotion expressed in social media content and a user's behavior in sharing information. Our research primarily focuses on the domain of political communication on Twitter. The findings from our analysis of two data sets, comprising over 165,000 tweets, suggest that emotionally charged Twitter remarks are more likely to be retweeted at a higher frequency and with greater speed compared to impartial remarks. It is recommended that businesses give priority to the development of advertising content that has a strong emotional impact. Additionally, businesses should focus on analyzing the sentiment surrounding their brands and products through social media communication.

Ramasamy et. al. [7] Enterprises and organizations have consistently shown that the perspectives and contributions of the community are a highly valuable and impactful asset. The widespread adoption of social media has created new possibilities for examining and assessing various aspects that businesses previously assessed using non-traditional, labor-intensive, and error-prone methods. The concept of "sentiment analysis" is intricately linked to this analytical methodology. The essence of sentiment analysis, an expansive discipline, lies in effectively categorizing content presented by users into predetermined polarity. Sentiment analysis and detection can be performed using a range of tools and techniques. One such technique is the use of supervised machine learning algorithms, which are trained on a set of training data from the target corpus. These algorithms are then able to categorize new data based on the patterns and information learned during training. The process of categorizing data using annotated corpora that depend on dictionaries is a fundamental aspect of lexical techniques. Hybrid tools, on the other hand, integrate lexicon-based algorithms with machine learning techniques. The Weka software is utilized for sentiment analysis in this study, employing Support Vector Machines (SVM). Support Vector Machines (SVM) are commonly used in supervised machine learning to identify textual polarity. The efficacy of Support Vector Machines (SVM) is assessed using two preclassified datasets obtained from Twitter. The comparison is performed utilizing three metrics: F-Measure, Precision, and Recall. The results are displayed using tables and graphs.

III. OBJECTIVES

The aim of the project is to utilize artificial intelligence and neural network architectures to empower machines to comprehend, interpret, and summarize text data in human languages in order to perform critical natural language processing tasks like sentiment classification and text summarization more effectively.

1) Develop neural network architectures like RNNs and Transformers to model linguistic contexts and sequences for language understanding.
2) Create annotated datasets for sentiment analysis and summarization to train and evaluate neural network models.
3) Explore transfer learning techniques to effectively pretrain language models on large corpora for improved performance.
4) Evaluate neural network models using relevant NLP metrics like accuracy, ROC curve, F1 score for tasks like sentiment classification.
5) Build end-to-end NLP pipelines that leverage trained neural networks to summarize documents and detect sentiment from text data.

IV. METHODOLOGY

The research design uses a secondary dataset which consists of 8,176 articles and consistent highlights that were obtained from the Daily Mail dataset. This dataset is the base for text summarization and other natural language processing tasks. The articles and the highlights establish a large quantity which allows the neural network models to learn the linguistic contexts and come up with the brief summaries . The research methodology is the creation of annotated subsets for the tasks such as sentiment analysis, and the use of the whole dataset for the techniques like transfer learning. The article of the daily mail dataset summary pairs, which means the availability of them, helps in supervised learning models and also makes the evaluation of summarization models with the right metrics possible.

A. Research Philosophy

The research philosophy is mainly positivist on empirical explanations and quantitative data analysis. The method consists of creating computational models like neural networks which are built on the theoretical understanding of language and machine learning principles. These models are reviewed against the text summarization of daily mails datasets using the exact metrics like accuracy and F1 scores. Generally, the philosophy is generating objectives about the effective architectures and techniques, which are then tested through experiments on the actual data to get insights and make the entitlements about their effectiveness.

B. Research Approach

The research approach use of the deductive and the inductive approach simultaneously. At first, the architectures of the neural networks algorithm designed by a deductive approach that is based on the theories and principles of the natural language processing and machine learning. These architectures are advanced and inductively evaluated in the experiments on the daily mail datasets for the tasks such as sentiment analysis and text summarization until they are perfected. The inductive part of the research means analyzing the empirical results, finding the patterns, and formulating new hypotheses about the most efficient transfer learning techniques or architectural changes.

C. Research Methods

The research method is a mixed method which is a combination of the quantitative and qualitative methods. The quantitative part is the experimental methods, where the neural network models are trained and evaluated on the daily mail articles data datasets using the statistical measures such as accuracy, F1 score, and ROC curves. The controlled research facilitates the making of objective comparisons between various architectures, hyper parameter configurations, and transfer learning techniques. The other is a qualitative approach that is based on the expert analysis of the model outputs for example, the quality of the generated summaries or the identification of the failures in the sentiment prediction. Through this multilevel approach, a clear view of the model performance is obtained, and also, the recommendations for the improvement of the future models are made.

D. Research Strategy

The research strategies that are used are based on the creation and valuation of neural network models for natural language processing tasks. Supervised learning methods are applied, where the daily mail articles datasets for sentiment analysis and text summarization are used as the labeled training data. The model architectures NLP model and Transformers are built using deep learning libraries (Khuran, et al. 2022). The transfer learning methods are studied by the retraining of language models on huge unlabeled quantities before the fine-tuning on the tasks. The right preprocessing pipe lines are created for cleaning, embedding, and data formatting. Lastly, systematic experiments are carried out to assess the model performance by the use of the metrics such as accuracy, F1 score and qualitative analysis of outputs.

E. Data Collection Methods

The data collection gathered a dataset from Kaggle, which contains 8,176 articles and their corresponding highlights gathered from the Daily Mail platforms. So that, the Google Knowledge Base is the most extensive dataset of the text summarization on this field used for the text summarization and natural language processing tasks. The dataset is obtained and thus processed to extract the article text and the highlight summaries which are the input data and the target labels, respectively. The essential data cleaning and preprocessing techniques are used to manage the inconsistencies or noise that are present in the raw data. The dataset is subsequently split into the training, the validation and the test sets for the purpose of the model development and the evaluation.

F. Research Ethics

The ethical problems are controlled because the dataset does not contain any personal or sensitive data. The data source Daily Mail is repeated correctly and is being acknowledged. The research is structured to generate models that are only objective and research purposes, and there is no intention to damage them. Besides, the model outputs are looked at in detail to find the possible biases or ethical problems and to make sure they are removed.

V. RESULTS AND DISCUSSION

A. Importing Result of Analysis

```
import pandas as pd
from bs4 import BeautifulSoup
import re
import string
import numpy as np
from PIL import Image
import nltk
from nltk.corpus import stopwords
from nltk.stem import WordNetLemmatizer
import matplotlib.pyplot as plt
import nltk
from wordcloud import WordCloud, STOPWORDS
from nltk.tokenize import word_tokenize
nltk.download('punkt')
nltk.download('stopwords')
nltk.download('wordnet')
```

Figure 1: Import libraries

This is to describe the code of the import libraries. Some necessary libraries are used for NLP analysis. In this code 3 methods are downloaded which are 'punkt', 'stopwords', and 'wordnet'. This method is used for the test summarizing.

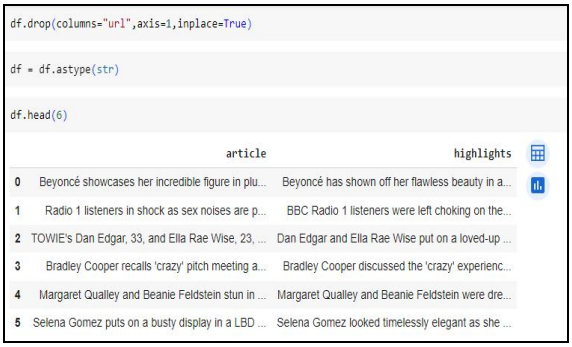
```
df=pd.read_csv("/content/article_highlights.csv")
```

```
df.head(6)
```

	url	article	highlights
0	https://www.dailymail.co.uk/tvshowbiz/article-...	Beyoncé showcases her incredible figure in plu...	Beyoncé has shown off her flawless beauty in a...
1	https://www.dailymail.co.uk/tvshowbiz/article-...	Radio 1 listeners in shock as sex noises are p...	BBC Radio 1 listeners were left choking on the...
2	https://www.dailymail.co.uk/tvshowbiz/article-...	TOWIE's Dan Edgar, 33, and Ella Rae Wise, 23, ...	Dan Edgar and Ella Rae Wise put on a loved-up ...
3	https://www.dailymail.co.uk/tvshowbiz/article-...	Bradley Cooper recalls 'crazy' pitch meeting a...	Bradley Cooper discussed the 'crazy' experienc...
4	https://www.dailymail.co.uk/tvshowbiz/article-...	Margaret Qualley and Beanie Feldstein stun in ...	Margaret Qualley and Beanie Feldstein were dre...
5	https://www.dailymail.co.uk/tvshowbiz/article-...	Selena Gomez puts on a busty display in a LBD ...	Selena Gomez looked timelessly elegant as she ...

Figure 2: Read dataset and the head of the dataset

This is represented to read the dataset of the daily mail articles and highlights. And the next is to display the head of the dataset. The method "df.head(6)" defines that top 6 data are displayed in the output.



```
df.drop(columns="url",axis=1,inplace=True)
```

```
df = df.astype(str)
```

```
df.head(6)
```

	article	highlights
0	Beyoncé showcases her incredible figure in plu...	Beyoncé has shown off her flawless beauty in a...
1	Radio 1 listeners in shock as sex noises are p...	BBC Radio 1 listeners were left choking on the...
2	TOWIE's Dan Edgar, 33, and Ella Rae Wise, 23, ...	Dan Edgar and Ella Rae Wise put on a loved-up ...
3	Bradley Cooper recalls 'crazy' pitch meeting a...	Bradley Cooper discussed the 'crazy' experienc...
4	Margaret Qualley and Beanie Feldstein stun in ...	Margaret Qualley and Beanie Feldstein were dre...
5	Selena Gomez puts on a busty display in a LBD ...	Selena Gomez looked timelessly elegant as she ...

Figure 3: Drop URL column of dataset

This is represented to drop the url column of the dataset. And the drop displays the head of the modified dataset. And the next is to display the head of the dataset. The method "df.head(6)" defines that top 6 data are displayed in the output.

```
def clean_text(text):
    if isinstance(text, str):
        # Remove HTML tags
        text = BeautifulSoup(text, 'html.parser').get_text()

        # Remove special characters and digits
        text = re.sub(r"[^a-zA-Z]", " ", text)

        # Remove punctuation
        text = text.translate(str.maketrans("", "", string.punctuation))

        # Remove emojis
        emoji_pattern = re.compile("["
                                   u"\U0001F600-\U0001F64F"  # emoticons
                                   u"\U0001F300-\U0001F5FF"  # symbols & pictographs
                                   u"\U0001F680-\U0001F6FF"  # transport & map symbols
                                   u"\U0001F1E0-\U0001F1FF"  # flags (iOS)
                                   u"\U00002702-\U000027B0"
                                   u"\U000024C2-\U0001F251"
                                   "]+", flags=re.UNICODE)
        text = emoji_pattern.sub(r'', text)

        # Convert to lowercase
        text = text.lower()

        # Remove stop words
        stop_words = set(stopwords.words('english'))
        tokens = word_tokenize(text)
        tokens = [word for word in tokens if word not in stop_words]
        text = ' '.join(tokens)

        return text
```

Figure 4: Analysis Clean test method

The above figure represented the "clean_text" function which accepts a text input and carries out several cleaning operations. It first verifies if the input is a string datatype or not. If it string data type, it removes "HTML" tags using "BeautifulSoup", removes special characters and digits using regular expressions, and removes punctuation using string methon. It removes emoji's by using a "Unicode regular expression pattern" which converts the text to lowercase, and roves stop words using "NLTK's" model "stop words" method and "word_tokenize". The cleaned text is given back as a string with the stop words taken out. The output will not be a string. If the input is not a string so that NaN or other data types, the fiction returns an empty string. This function is accessible for the cleaning and preprocessing of text data before the further analysis or modeling.

```
df["clean_article"]=df["article"].apply(clean_text)

df["clean_highlights"]=df['highlights'].apply(clean_text)

df.head(6)
```

	article	highlights	clean_article	clean_highlights
0	Beyoncé showcases her incredible figure in plu...	Beyoncé has shown off her flawless beauty in a...	beyonc showcases incredible figure plunging wh...	beyonc shown flawless beauty new photo promote...
1	Radio 1 listeners in shock as sex noises are p...	BBC Radio 1 listeners were left choking on the...	radio listeners shock sex noises played greg j...	bbc radio listeners left choking cornflakes we...
2	TOWIE's Dan Edgar, 33, and Ella Rae Wise, 23, ...	Dan Edgar and Ella Rae Wise put on a loved-up ...	towie dan edgar ella rae wise put loved displa...	dan edgar ella rae wise put loved display sun ...
3	Bradley Cooper recalls 'crazy' pitch meeting a...	Bradley Cooper discussed the 'crazy' experienc...	bradley cooper recalls crazy pitch meeting bey...	bradley cooper discussed crazy experience meet...
4	Margaret Qualley and Beanie Feldstein stun in ...	Margaret Qualley and Beanie Feldstein were dre...	margaret qualley beanie feldstein stun chic fl...	margaret qualley beanie feldstein dressed nine...
5	Selena Gomez puts on a busty display in a LBD ...	Selena Gomez looked timelessly elegant as she ...	selena gomez puts busty display lbd teamed chi...	selena gomez looked timelessly elegant headed ...

Figure 5: Head of the clean text

The above figure represents the head of the cleaning text dataset. It displays using the “df.head(6)” based on the above analysis of the cleaning text. It displays the top 6 data of the cleaning dataset. The cleaning dataset covers articles, highlights, clea_articles, clean_highlights columns.

```
pretrained_model_name = "sshleifer/distilbart-cnn-12-6"
hf_arch, hf_config, hf_tokenizer, hf_model = get_hf_objects(pretrained_model_name,model_cls=Bar

hf_arch, type(hf_config), type(hf_tokenizer), type(hf_model)

/usr/local/lib/python3.10/dist-packages/huggingface_hub/file_download.py:1132: FutureWarning: `
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/huggingface_hub/utils/_token.py:89: UserWarning:
The secret `HF_TOKEN` does not exist in your Colab secrets.
To authenticate with the Hugging Face Hub, create a token in your settings tab (https://huggingf
You will be able to reuse this secret in all of your notebooks.
Please note that authentication is recommended but still optional to access public models or da
  warnings.warn(
config.json: 100% 1.80k/1.80k [00:00<00:00, 35.2kB/s]
tokenizer_config.json: 100% 26.0/26.0 [00:00<00:00, 345B/s]
vocab.json: 100% 899k/899k [00:00<00:00, 8.40MB/s]
merges.txt: 100% 456k/456k [00:00<00:00, 12.0MB/s]
pytorch_model.bin: 100% 1.22G/1.22G [00:14<00:00, 105MB/s]
('bart',
 transformers.models.bart.configuration_bart.BartConfig,
 transformers.models.bart.tokenization_bart_fast.BartTokenizerFast,
 transformers.models.bart.modeling_bart.BartForConditionalGeneration)
```

Figure 6: Model analysis

The code represented the Hugging Face Transformers library which is used for preparing the tasks of “natural language processing”. It implies the usage of a pre-trained model which is "sshleifer/distilbart-cnn-12-6" that is available in the “Hugging Face Model Hub”. The “get_hf_objects” method is likely a custom function that reruns four objects which are “hf_arch”, “hf_config”, “hf_tokenizer, and “hf_model”. This conation, the pre-trained model is the example of “Bart for Conditional Generation” model, which is a Transformer based “sequence-to-sequence” model for conditional generation tasks, like text summarization or translation. Lastly the code conveys the types of the returned objects that can be helpful for debuting or learning about the data structures utilized in the Transformers library.

```
text_gen_kwargs = default_text_gen_kwargs(hf_config, hf_model, task='summarization')
text_gen_kwargs

{'early_stopping': True,
 'length_penalty': 2.0,
 'max_length': 142,
 'min_length': 56,
 'no_repeat_ngram_size': 3,
 'num_beams': 4}
```

Figure 7: Analysis text summarization

The above code represented to creates the required parameters for text generation task that uses the results from the “Hugging Face configuration” library and the model. Using “text_gen_kwargs” dictionary that includes multiple options like “max_length” and duplication penalty amongst others that are tasked through summarization to influence the output of a text generation process.

Figure 8: Text prediction

This represents the prediction of the particular text. Acording to the above code, set a variable for store the text article. Also the nest code defies in such a way that the “test_article” is in several summarized forms using the “blurr_summarize” method from the learn object. For that it provides 3 summaries by using beam search with 4 beams and `num_return_sequences=3” “early_stopping=True " defines stopping early by dressing up and striking on a good summary. It will further just add the summary with a label which corresponds to its prediction number.

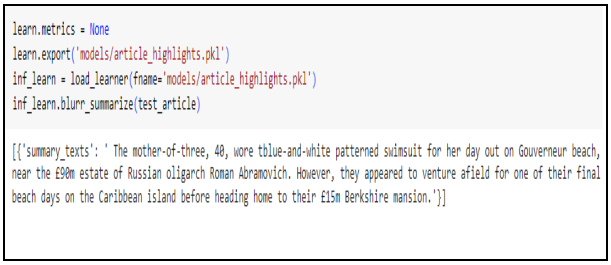


Figure 9: Summary text

This image defines the surrey of text after applying summarization methods. According to the code it defines the learn the article and export the article to the highlight section after that it displays the article summary text after completing summarizing analysis.

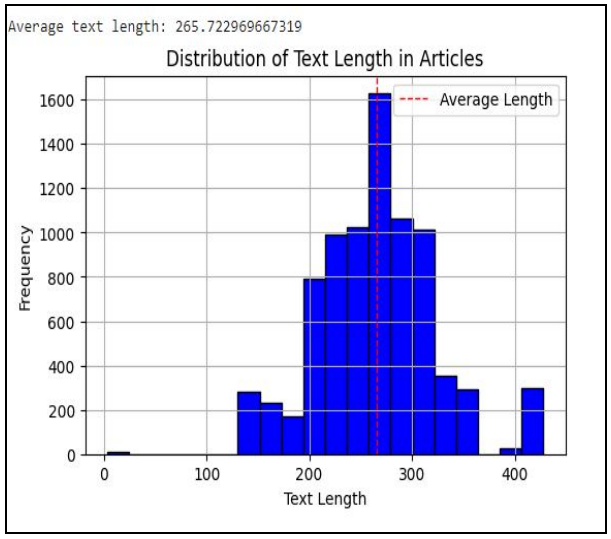


Figure 10: Visualization of Average text length of articles

This above visualization of the average text length of articles. It also defines how large each article text in the Data Frame df is and put the result in the column name 'text_length'. Then, it averages the article text length for all the articles, and displays the result. Next, it visualizes the histogram plot that defines the distribution of text lengths. The histogram has 20 bars that divided by blue and similar colors. The x-axis displays the” text length” and the y-axis shows the “number of the articles” every length. The average text length is represented by a red vertical ruined line which is the aspect from which the figure is constructed. The last step is creating a grid over the plot.

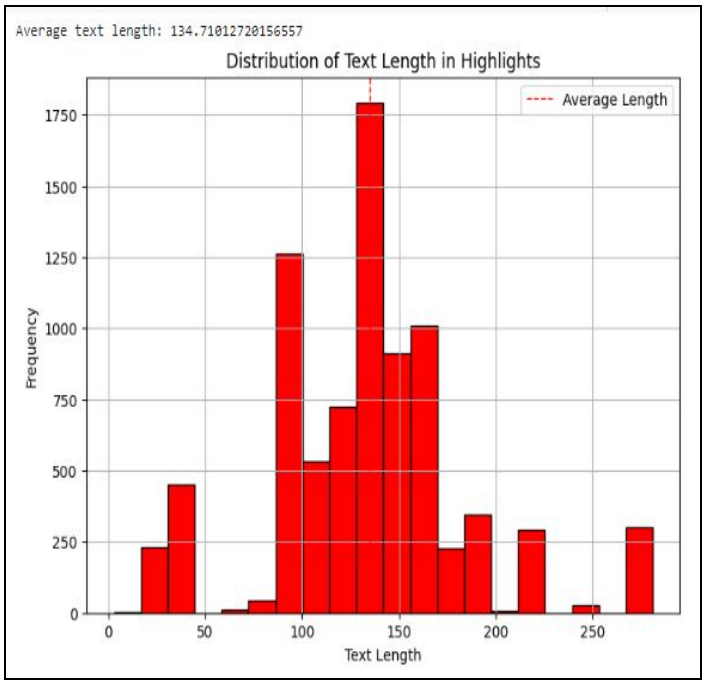


Figure 11: Visualization of Average text length of highlights

This above visualization of the average text length of highlights. It also defines how large each highlight text in the DataFrame df is and puts the result in the column name 'text_length'. Then, it averages the highlighted text length for all the articles, and displays the result. Next, it visualizes the histogram plot that defines the distribution of text lengths. The histogram has 20 bars that divided by red. The x-axis displays the" text length" and the y-axis shows the "number of the highlights" every length.

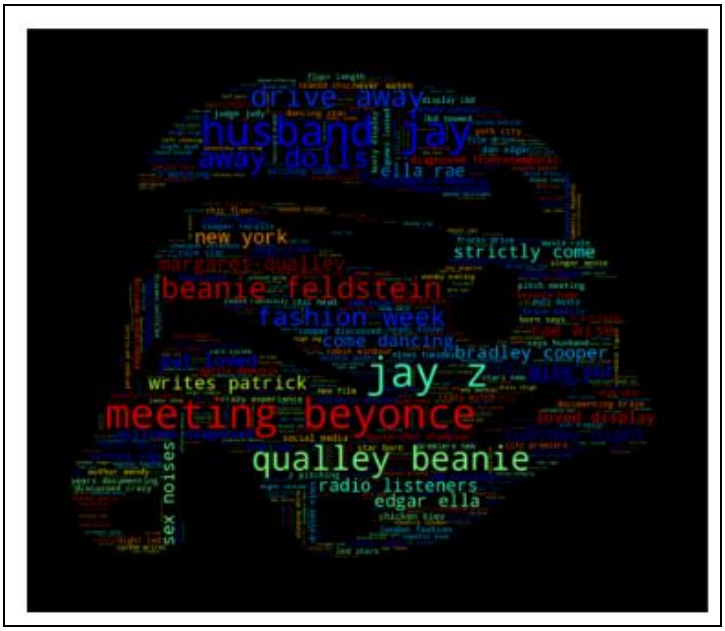


Figure 12: Visualization of text summarization of articles

The given code builds a visually proper word cloud that demonstrates the most commonly used words across the cleaned up articles are stored in the DataFrame df. It simply concatenates all the articles' cleaned text from 'clean_article' column into one big string "all_textNow " the "object is initialized and the following parameters that have been set. The 'jet' color map, and a custom mask "wordcloud_mask". The word cloud is created using the "generate()" function specifying "all_text" as input. Next, the cloud image is rendered by plot.

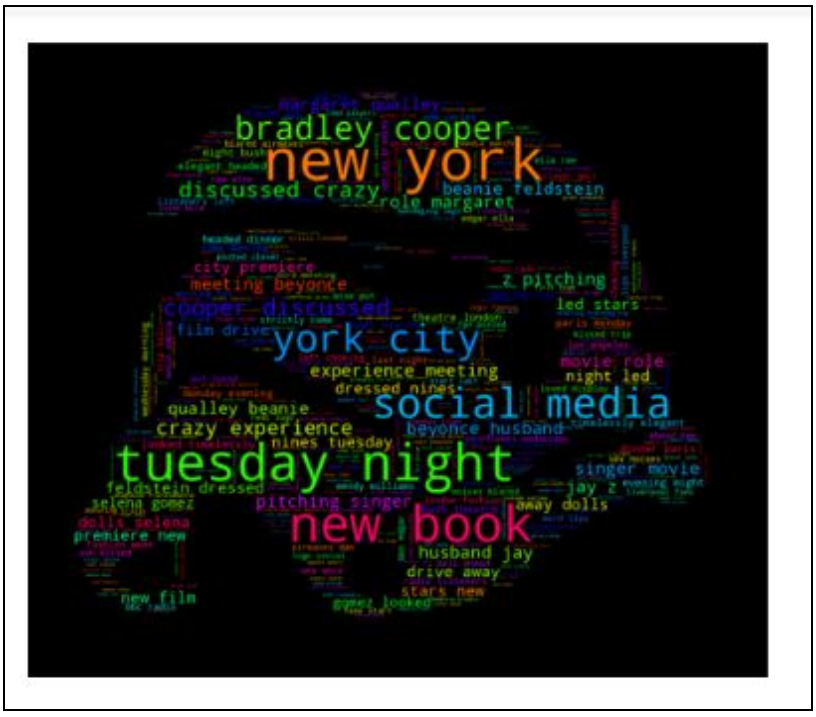


Figure 13: Visualization of text summarization of highlights

words across the cleaned up highlights are stored in the DataFrame df. It simply concatenates all the highlighted cleaned text from 'clean_highlights' column into one big string “all_textNow " the “object is initialized and the following parameters that have been set. The 'jet' colormap, and a custom mask “wordcloud_mask”. The word cloud is created using the “generate ()” function specifying “all_text” as input. Next, the cloud image is rendered by plot.

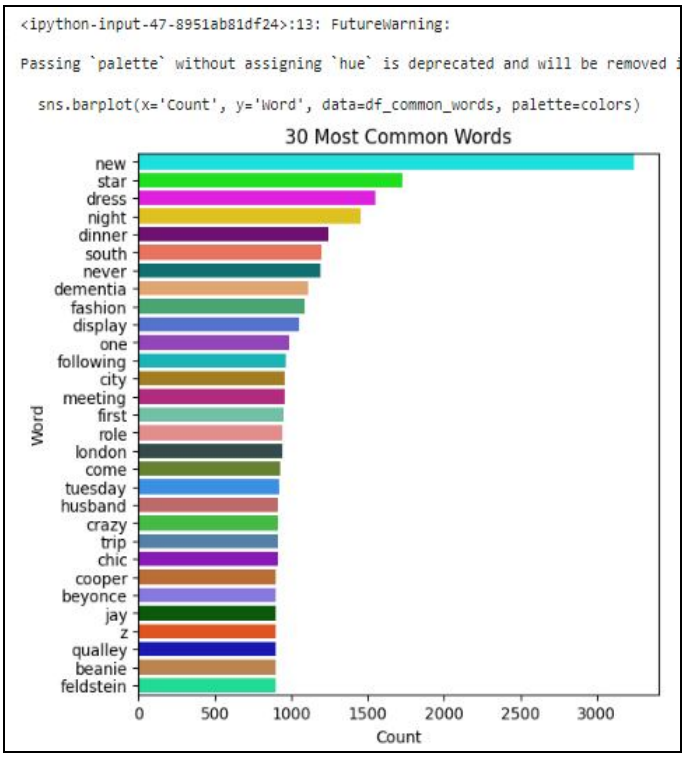


Figure 14: Visualization of 30 most common words of articles

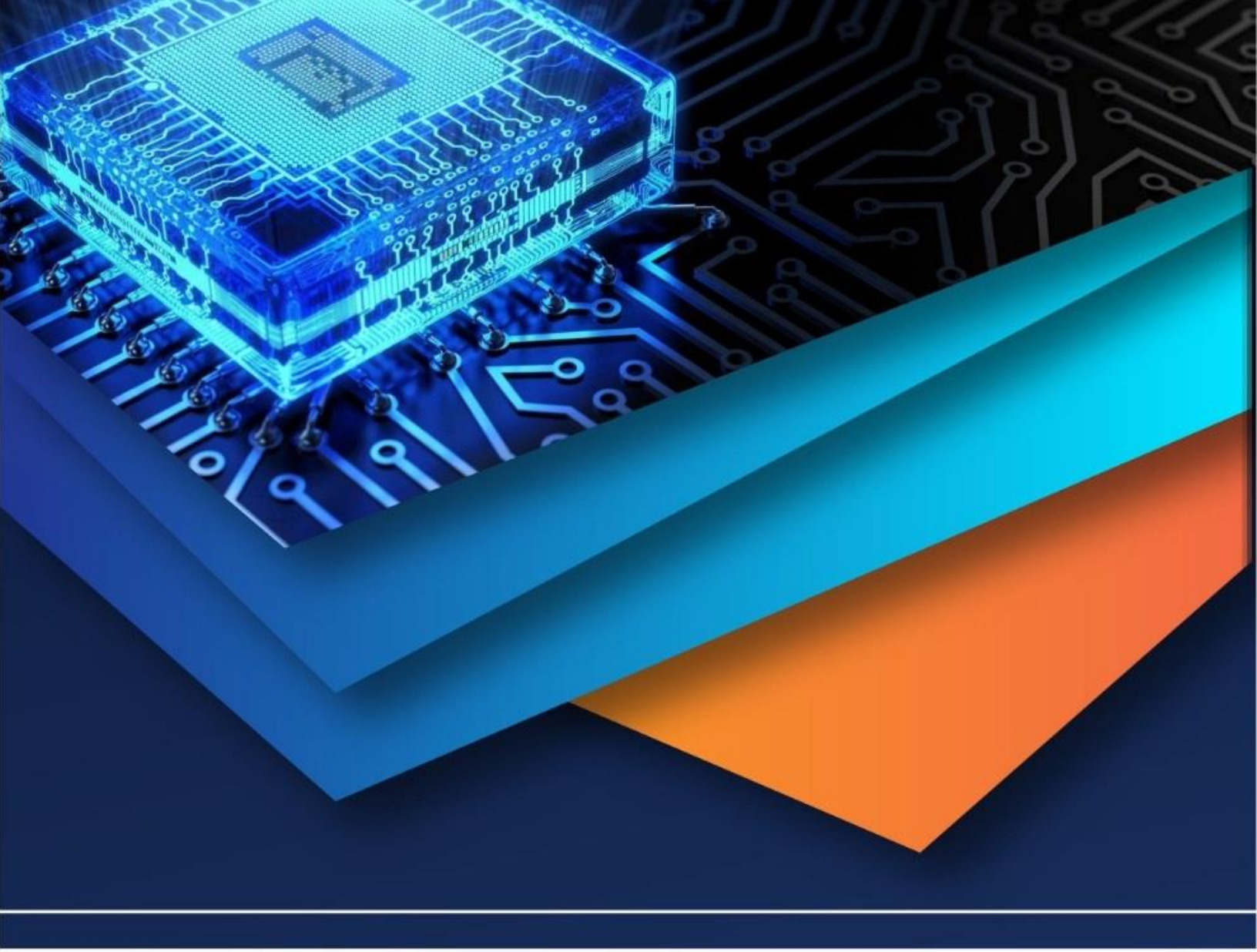
The visualization used to determine the word frequency in the cleared text of articles so that this bar chart used to visualize the top 30 common words. Then it splits the extracts of the cleaned corpus resulting in a list of all words in a separate list called all_words of articles. So the program uses the "wordcount" method from the collections module and in number that returns the frequency of terms. The top 30 most frequent words and their counts and put them into a dataset which is called "df_common_words". After that, there is the creation of a list of 30 individual colors which used in the bar chart. Lastly the bar plot that is displayed with plt. chart(), displaying the top 30 articles words both in the form of "word cloud" and word frequency.

VI. CONCLUSION

The analysis of text summarization confirms that AI and neural networks can effectively perform tasks such as sentiment analysis and text summarization using models like NLTK. The work focused on developing robust models that employ large text data from articles and their highlights, enabling the system to mimic human methods of performing these tasks. Despite the progress, it is crucial not to overlook the challenges and to continue developing tools and integrating technical skills with a background in ethical studies. Proper implementation of NLP and text summarization techniques allows for the analysis of textual data from a novel perspective, paving the way for further advancements in the field. The provided training metrics show the effectiveness of the AI model and neural networks in performing text summarization tasks. The consistent decrease in training and validation loss over the epochs indicates robust learning and generalization capabilities of the model. The accuracy remained high throughout the training process, reflecting the model's proficiency in classification tasks. The improvement in ROUGE scores suggests that the model has enhanced its ability to identify relevant unigrams, bigrams, and longest common subsequences in the summaries. Additionally, the high and stable BERT Score metrics underscore the model's precision and recall in generating summaries that closely align with human-generated references. These results highlight the potential of deep neural networks in text summarization tasks, confirming their ability to understand and summarize textual content effectively. This research contributes to the advancement of NLP techniques and suggests promising directions for future work, such as exploring deeper neural networks, attention models, and multimodal data integration to further improve performance.

REFERENCES

[1] Davis E, Marcus G (2015) Commonsense reasoning and commonsense knowledge in artificial intelligence. Commun ACM 58(9):92–103
[2] Desai NP, Dabhi VK (2022) Resources and components for Gujarati NLP systems: a survey. Artif Intell Rev:1–19
[3] Devlin J, Chang MW, Lee K, Toutanova K, (2018) Bert: pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805
[4] Anjaria, M.; Guddeti, R.M.R. A novel sentiment analysis of social networks using supervised learning. Soc. Netw. Anal. Min. 2014, 4, 181. [CrossRef]
[5] Ramaswamy, S.; DeClerck, N. Customer perception analysis using deep learning and NLP. Procedia Comput. Sci. 2018, 140, 170–178. [CrossRef]
[6] Stieglitz, S.; Dang-Xuan, L. Emotions and information diffusion in social media—Sentiment of microblogs and sharing behavior. J. Manag. Inf. Syst. 2013, 29, 217–248
[7] Ramasamy, L.K.; Kadry, S.; Nam, Y.; Meqdad, M.N. Performance analysis of sentiments in Twitter dataset using SVM models. Int. J. Electr. Comput. Eng. 2021, 11, 2275–2284.
[8] Singh, B.; Kushwaha, N.; Vyas, O.P. An interpretation of sentiment analysis for enrichment of Business Intelligence. In Proceedings of the 2016 IEEE Region 10 Conference (TENCON), Singapore, 22–25 November 2016; pp. 18–23.
[9] Ghiassi, M.; Zimbra, D.; Lee, S. Targeted twitter sentiment analysis for brands using supervised feature engineering and the dynamic architecture for artificial neural networks. J. Manag. Inf. Syst. 2016, 33, 1034–1058.
[10] Yadav, V.; Verma, P.; Katiyar, V. E-commerce product reviews using aspect based Hindi sentiment analysis. In Proceedings of the 2021 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 27–29 January 2021; pp. 1–8.
[11] Desai, Z.; Anklesaria, K.; Balasubramaniam, H. Business Intelligence Visualization Using Deep Learning Based Sentiment Analysis on Amazon Review Data. In Proceedings of the 2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT), Kharagpur, India, 6–8 July 2021; pp. 1–7.
[12] Devika, M.; Sunitha, C.; Ganesh, A. Sentiment analysis: A comparative study on different approaches. Procedia Comput. Sci. 2016, 87, 44–49



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY