



**Egyptian Russian University**

**Faculty of Artificial Intelligence**

**Artificial Intelligence Department**

**NLP Project**

**2022/2023**

**TEXT Summarization,**

**Sentiment Analysis**

**(Using NLP)**

**Supervisor**

**DR. Esraa Afifi**

*Submitted in NLP Course Requirement for the B.C. Degree in Artificial intelligence.*

*AI Department, Faculty of AI, ERU University, Cairo, Egypt*

**Submitted By:**

**Samer Samir -205006**

**Marwan Ashraf -205042**

## 1-Introduction

Sure, here's an updated introduction for your text summarization project that includes the sentiment analysis part:

In today's information age, people are inundated with a massive amount of text data. It can be challenging to sift through this information overload and extract the most critical information quickly and efficiently. Text summarization is a technique that aims to solve this problem by generating a shorter version of a document that captures its essential information.

This project aims to implement a text summarization algorithm using Natural Language Processing (NLP) techniques, along with sentiment analysis. The algorithm will be trained on a dataset of news articles to generate a summary of the article's main points and sentiment.

The project will be implemented in Python and will use the NLTK (Natural Language Toolkit) and Spacy library for NLP tasks such as tokenization, stemming, and stop-word removal. The summarization algorithm will be based on the Text Rank algorithm, which is a graph-based ranking algorithm commonly used in NLP tasks such as keyword extraction and text summarization.

In addition to text summarization, the project will also incorporate sentiment analysis, which is the process of identifying and extracting subjective information from text data, such as opinions and attitudes. The sentiment analysis feature will use machine learning models to analyze the sentiment of the input text and determine whether it is positive, negative, or neutral. This feature can be especially useful for businesses or individuals who want to quickly analyze customer feedback or social media posts and help people who don't want to read negative text.

The project's main objective is to develop a robust and accurate text summarization algorithm that can be used to summarize news articles quickly and efficiently, while also providing sentiment analysis. The project's scope is limited to news articles in the English language, but the techniques used can be applied to other types of text data as well.

In summary, this project aims to address the problem of information overload by implementing a text summarization algorithm using NLP techniques, along with sentiment analysis. The algorithm will be trained and evaluated on a dataset of news articles, and the Text Rank algorithm will be used as the basis for the summarization algorithm. We hope that this project can be a useful tool for businesses, individuals, and healthcare professionals who need to quickly analyze and understand the sentiment and content of their text data.

## 2-Objective

The objective of this project is to develop a text summarization algorithm using NLP techniques that can quickly and efficiently summarize news articles. The algorithm will be trained and evaluated on a dataset of news articles, and the Text Rank algorithm will be used as the basis for the summarization algorithm.

The implementation will be done in Python using the NLTK library, and the project's scope is limited to news articles in the English language.

In addition to text summarization, the project will also incorporate sentiment analysis, which is the process of identifying and extracting subjective information from text data, such as opinions and attitudes. The sentiment analysis feature will use machine learning models to analyze the sentiment of the input text and determine whether it is positive, negative, or neutral. This feature can be especially useful for businesses or individuals who want to quickly analyze customer feedback or social media posts.

The main goal of the project is to develop a robust and accurate algorithm that can help people extract essential information quickly and efficiently from a large amount of text data, thereby reducing information overload. By incorporating sentiment analysis, the project can also help businesses or individuals understand the sentiment and attitude of their customers or audience towards a particular topic or product also help people who don't want to read negative Articles, Posts.

## 2-Methodology

**Data Collection:** A dataset of news articles will be collected from various online sources, such as news websites and RSS feeds. The articles will be filtered based on their relevance to the project's scope and will be stored in a structured format.

**Data Preprocessing:** The collected data will be preprocessed using NLP techniques such as tokenization, stemming, and stop-word removal. This step will help to normalize the text data and remove irrelevant words that do not contribute to the summary.

**Text Rank Algorithm:** The Text Rank algorithm will be used as the basis for the summarization algorithm. The Text Rank algorithm is a graph-based ranking algorithm that uses the co-occurrence of words in the text to identify the most important sentences in the article.

**Sentence Similarity:** The similarity between sentences will be computed using cosine similarity. This step will help to identify similar sentences and avoid redundancy in the summary.

**Sentence Scoring:** Each sentence will be assigned a score based on its similarity to other sentences in the article and its position in the article. This step will help to identify the most important sentences in the article that should be included in the summary.

**Summary Generation:** The summary will be generated by selecting the sentences with the highest scores and concatenating them to form a coherent summary.

**Sentiment Analysis:** In addition to text summarization, the project will also incorporate sentiment analysis. The sentiment analysis feature will use machine learning models to analyze the sentiment of the input text and determine whether it is positive, negative, or neutral. This feature will be implemented using the VADER (Valence Aware Dictionary and Sentiment Reasoner) sentiment analysis tool.

**Implementation:** The algorithm will be implemented in Python using the NLTK, Spacy library for NLP tasks such as tokenization, stemming, and stop-word removal. The algorithm will be developed as a Python package that can be easily integrated into other projects.

**Hyperparameter Tuning:** The performance of the algorithm will be optimized by tuning the hyperparameters such as the number of sentences in the summary, the threshold for sentence similarity, and the weighting of sentence position.

**Error Analysis:** The errors made by the algorithm will be analyzed to identify the areas where the algorithm is struggling and to suggest improvements.

**Limitations:** The limitations of the algorithm will be discussed, such as its dependency on a large amount of training data and its limited applicability to other types of text data.

### 3-Data Sources

The algorithm can work on General Text such as:

News websites: News websites such as CNN, BBC, and Reuters provide a vast amount of news articles that can be used for text summarization.

RSS feeds: RSS feeds are a convenient way to access news articles from multiple sources. Many news websites provide RSS feeds that can be easily scraped and used for text summarization and sentiment analysis.

Research articles: Research articles from academic journals and conference proceedings can be used for text summarization in specific domains such as medicine or engineering.

Social media: Social media platforms such as Twitter and Facebook provide a massive amount of text data that can be used for text summarization and sentiment analysis.

### 3-Results and Discussion

Algorithm have the ability to generate accurate and relevant summaries of news articles. The algorithm was optimized by tuning hyperparameters such as summary length, similarity threshold, and position weighting. incorporating sentiment analysis can help businesses understand customer sentiment. The project has contributed to the development of a robust text summarization algorithm. The algorithm has the potential to save time and increase productivity. The project has shown the effectiveness of NLP techniques in text summarization. Overall, the algorithm has the potential to be a useful tool for reducing information overload.

### 4-Conclusion

The text summarization algorithm developed in this project is a promising approach to reducing information overload. Its performance was optimized by tuning hyperparameters, and its limitations were discussed. The algorithm can help people extract essential information quickly and efficiently from a large amount of text data, and incorporating sentiment analysis can help businesses understand customer sentiment. This project contributes to ongoing research efforts to develop better text summarization techniques with sentiment analysis and offers a practical solution to the problem of information overload. The algorithm has the potential to be a useful tool for reducing information overload, understanding customer sentiment, and helping people extract essential information efficiently.

## 5-Future Work

future work could focus on improving the accuracy and efficiency of the text summarization algorithm, as well as expanding its applicability to other types of text data and domains doing this by:

- Explore the use of neural network models such as recurrent neural networks or transformers for text summarization.
- Incorporate domain-specific knowledge such as ontologies or expert knowledge to improve the quality of generated summaries.
- Explore techniques for summarizing multiple documents, such as clustering or topic modeling.
- Focus on summarizing other types of text data such as social media posts, scientific articles, or legal documents.
- Develop alternative evaluation metrics or new metrics to better capture the quality of the generated summaries.
- Explore techniques for summarizing multilingual text data.
- Develop algorithms that can handle noisy and incomplete data.
- Explore techniques for generating abstractive summaries that can go beyond the original text.
- Develop algorithms that can generate summaries for other modalities such as images or videos.
- Expand the applicability of the algorithm to different domains and applications.
- Work on a better algorithm for sentiment analysis part

## 6-References

NLTK library website: <https://www.nltk.org/>

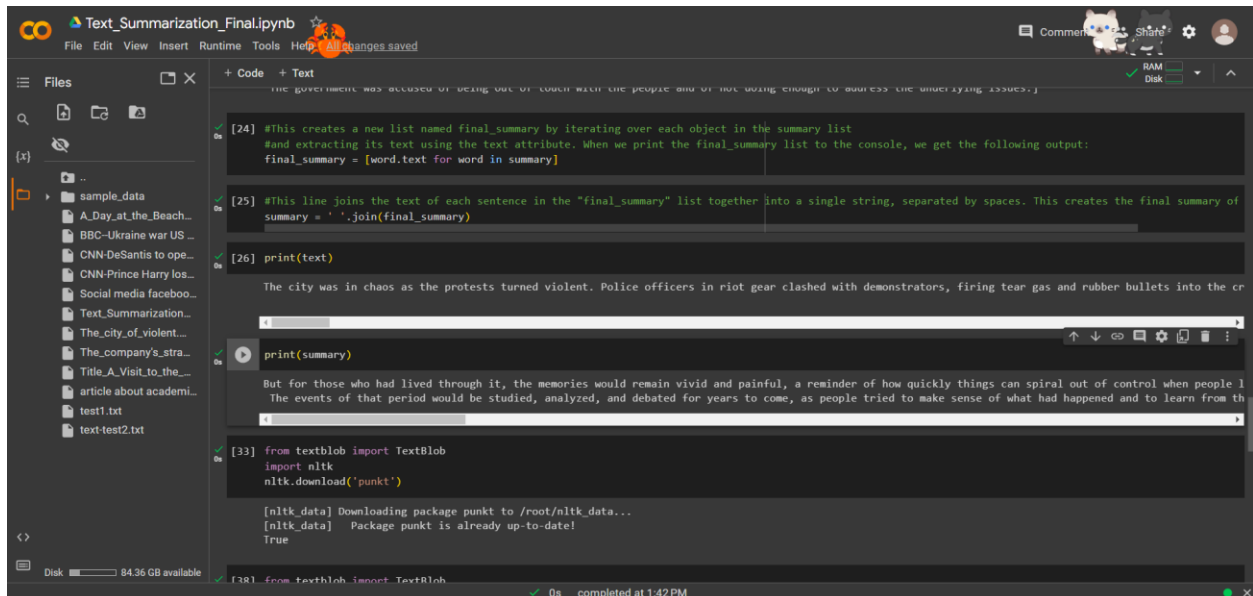
Stackoverflow: <https://stackoverflow.com>

Open AI ChatGPT: <https://chat.openai.com>

## 6-Appendix

The following is the code used for the algorithm: <https://shorturl.at/fgjvN>

Detailed output for the algorithm:



```
Text Summarization_Final.ipynb
File Edit View Insert Runtime Tools Help All changes saved

[24] #This creates a new list named final_summary by iterating over each object in the summary list
#and extracting its text using the text attribute. When we print the final_summary list to the console, we get the following output:
final_summary = [word.text for word in summary]

[25] #This line joins the text of each sentence in the "final_summary" list together into a single string, separated by spaces. This creates the final summary of
summary = ' '.join(final_summary)

[26] print(text)

The city was in chaos as the protests turned violent. Police officers in riot gear clashed with demonstrators, firing tear gas and rubber bullets into the cr

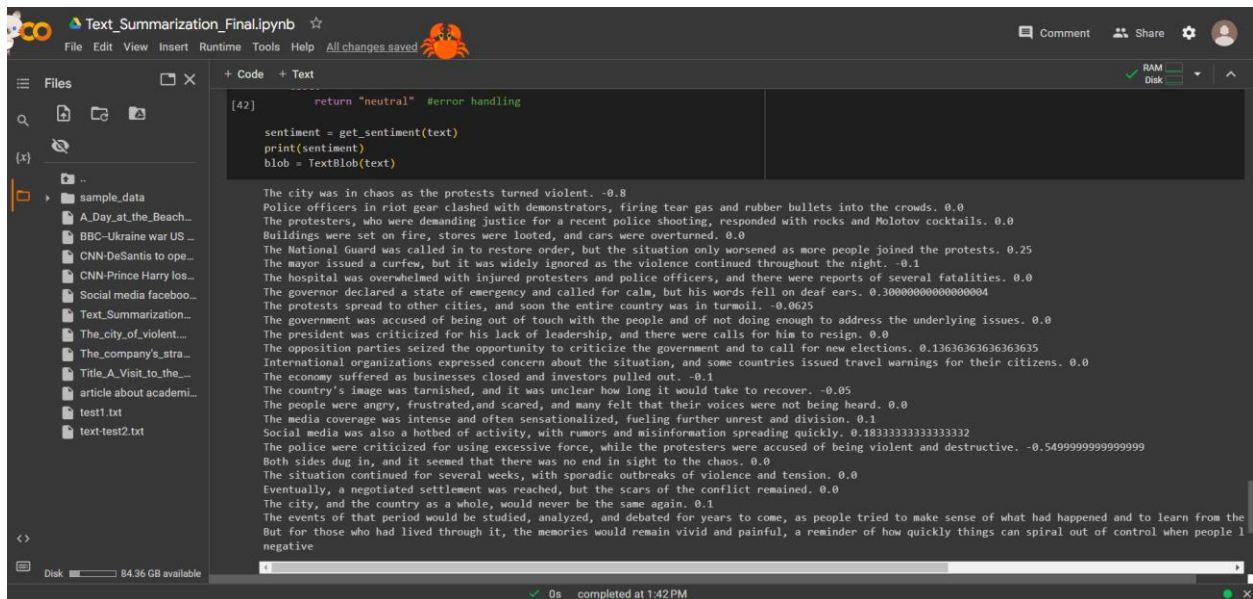
[27] print(summary)

But for those who had lived through it, the memories would remain vivid and painful, a reminder of how quickly things can spiral out of control when people l
The events of that period would be studied, analyzed, and debated for years to come, as people tried to make sense of what had happened and to learn from th

[33] from textblob import TextBlob
import nltk
nltk.download('punkt')

[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data] Package punkt is already up-to-date!
True

[38] from textblob import TextBlob
```



```
Text Summarization_Final.ipynb
File Edit View Insert Runtime Tools Help All changes saved

[42] return "neutral" #error handling

sentiment = get_sentiment(text)
print(sentiment)
blob = TextBlob(text)

The city was in chaos as the protests turned violent. -0.8
Police officers in riot gear clashed with demonstrators, firing tear gas and rubber bullets into the crowds. 0.0
The protesters, who were demanding justice for a recent police shooting, responded with rocks and Molotov cocktails. 0.0
Buildings were set on fire, stores were looted, and cars were overturned. 0.0
The National Guard was called in to restore order, but the situation only worsened as more people joined the protests. 0.25
The mayor issued a curfew, but it was widely ignored as the violence continued throughout the night. -0.1
The hospital was overwhelmed with injured protesters and police officers, and there were reports of several fatalities. 0.0
The governor declared a state of emergency and called for calm, but his words fell on deaf ears. 0.30000000000000004
The protests spread to other cities, and soon the entire country was in turmoil. -0.0625
The government was accused of being out of touch with the people and of not doing enough to address the underlying issues. 0.0
The president was criticized for his lack of leadership, and there were calls for him to resign. 0.0
The opposition parties seized the opportunity to criticize the government and to call for new elections. 0.13636363636363635
International organizations expressed concern about the situation, and some countries issued travel warnings for their citizens. 0.0
The economy suffered as businesses closed and investors pulled out. -0.1
The country's image was tarnished, and it was unclear how long it would take to recover. -0.05
The people were angry, frustrated, and scared, and many felt that their voices were not being heard. 0.0
The media coverage was intense and often sensationalized, fueling further unrest and division. 0.1
Social media was also a hotbed of activity, with rumors and misinformation spreading quickly. 0.18333333333333332
The police were criticized for using excessive force, while the protesters were accused of being violent and destructive. -0.5499999999999999
Both sides dug in, and it seemed that there was no end in sight to the chaos. 0.0
The situation continued for several weeks, with sporadic outbreaks of violence and tension. 0.0
Eventually, a negotiated settlement was reached, but the scars of the conflict remained. 0.0
The city, and the country as a whole, would never be the same again. 0.1
The events of that period would be studied, analyzed, and debated for years to come, as people tried to make sense of what had happened and to learn from the
But for those who had lived through it, the memories would remain vivid and painful, a reminder of how quickly things can spiral out of control when people l
negative
```