**The Superior University**

**Project Title**

**News Article Summarizer**

**Project Details**

1. **Course**: Artificial Intelligence
2. **Instructor**: Sir Rasikh
3. **Semester**: 3rd
4. **Section**: 3A
5. **Submission Date**: 8 December 2024

**Abstract**

The "News Article Summarizer" is an AI-based application designed to fetch and summarize news articles from the web. It employs natural language processing (NLP) to extract the main points from lengthy articles, providing users with concise summaries. The system also performs sentiment analysis to determine the overall tone of the article (positive, negative, or neutral). Developed using Python, this project integrates the **Newspaper3k** library for web scraping and article parsing, **TextBlob** for sentiment analysis, and **Tkinter** for a graphical user interface (GUI). The tool simplifies information consumption, making it valuable for students, researchers, and professionals.

**Table of Contents**

1. **Introduction**
2. **Objectives**
3. **System Requirements**
4. **Methodology**
5. **Implementation**
6. **Challenges and Solutions**
7. **Conclusion**

**1. Introduction**

The "News Article Summarizer" project addresses the growing need for efficient information consumption in the digital era. With an overwhelming volume of news articles published daily, manually reading and extracting key points becomes challenging. This project leverages NLP techniques to automate the summarization and sentiment analysis of news articles, offering users a convenient and time-saving solution.

**2. Objectives**

* To develop an AI-based tool for summarizing news articles.
* To integrate sentiment analysis for determining the article's tone.
* To build a user-friendly GUI for seamless interaction.
* To utilize libraries like **Newspaper3k** for article parsing and **TextBlob** for NLP.

**3. System Requirements**

* **Hardware Requirements**:
  + Processor: Intel Core i3 or higher
  + RAM: 4 GB or more
  + Storage: 500 MB free space
* **Software Requirements**:
  + Programming Language: Python 3.x
  + Libraries: Newspaper3k, TextBlob, Tkinter
  + Operating System: Windows/Linux/MacOS

**4. Methodology**

1. **Design Approach**:
   * The tool accepts a URL as input, downloads the article, parses its content, and summarizes it.
   * Sentiment analysis is performed on the full article text to provide insights into the article's tone.
2. **Workflow**:
   * **Input**: User provides the URL of a news article.
   * **Processing**: Article is fetched and processed using **Newspaper3k**.
     + Summary is generated using NLP techniques.
     + Sentiment analysis is performed using **TextBlob**.
   * **Output**: Title, author, publication date, summary, and sentiment are displayed in the GUI.
3. **Key Algorithm**:
   * NLP-based summarization and polarity scoring for sentiment analysis.

**5. Implementation**

* **Graphical User Interface (GUI)**:  
  The interface is developed using **Tkinter**, featuring text boxes for displaying the title, author, publication date, summary, and sentiment analysis.
* **Core Functions**:
  + summarize(): Fetches, parses, and processes the article for summarization and sentiment analysis.
  + GUI elements for user interaction and displaying results.
* **Sample Code Snippet**:

def summarize():

url = url\_text.get().strip()

article = Article(url)

article.download()

article.parse()

article.nlp()

title.delete('1.0', 'end')

title.insert('1.0', article.title)

summary.delete('1.0', 'end')

summary.insert('1.0', article.summary)

analysis = TextBlob(article.text)

sentiment.delete('1.0', 'end')

sentiment.insert('1.0', f"Polarity: {analysis.polarity}")

**6. Challenges and Solutions**

1. **Challenge**: Handling poorly structured articles.  
   **Solution**: Integrated error handling for missing metadata and unstructured content.
2. **Challenge**: Generating accurate summaries.  
   **Solution**: Utilized **Newspaper3k**’s NLP pipeline for reliable summarization.

**7. Conclusion**

The "News Article Summarizer" successfully demonstrates the use of AI and NLP for automating article summarization and sentiment analysis. Future enhancements could include multilingual support, advanced sentiment classification, and browser extensions for real-time summarization.