Project title: Fact-Checking Newspaper: A Python-Based Project

Submitted by

Shohel Rana

Roll: 25

Department of Mass Communication and Journalism

Barishal University

Submitted to:

Md. Rashid Al Asif

Assistant Professor

Dept. of Computer Science & Engineering

University of Barishal

Date of Submission: 16 February 2025

Project title: Fact-Checking Newspaper: A Python-Based Project

Abstract

In the era of information overload, the credibility of news articles has become a significant concern. Misinformation and fake news can spread rapidly, leading to public confusion and mistrust. This project aims to develop a Python-based fact-checking newspaper system that allows users to add, view, and verify news articles using various verification methods. The system is designed to help users assess the credibility of news articles by verifying their sources, images/videos, and text. This paper provides a detailed explanation of the project, including its design, implementation, and potential applications.

Introduction:

Background:

The proliferation of digital media has made it easier than ever to disseminate information. However, this ease of access has also led to the spread of misinformation and fake news. According to a study by the Pew Research Center, 64% of Americans say that fake news has caused "a great deal of confusion" about the basic facts of current events (Mitchell et al., 2016). This has created a pressing need for tools that can help users verify the credibility of news articles.

Objective

The primary objective of this project is to create a Python-based system that allows users to add, view, and verify news articles. The system will provide multiple verification methods, including source verification, image/video analysis, and text analysis. By using this system, users can make more informed decisions about the credibility of the news they consume.

Scope

This project focuses on developing a command-line interface (CLI) application that allows users to interact with the fact-checking system. The system is designed to be simple and user-friendly, making it accessible to a wide range of users. The project does not include advanced machine learning or natural language processing techniques but provides a foundation for future enhancements.

Literature Review

The Problem of Fake News

Fake news has become a significant issue in recent years, with social media platforms being particularly susceptible to the spread of misinformation. A study by Vosoughi et al. (2018) found that false news spreads faster and more broadly than true news on Twitter. This highlights the need for effective fact-checking tools that can help users identify and avoid fake news.

Existing Solutions

Several fact-checking tools and platforms have been developed to address the problem of fake news. For example, FactCheck.org and Snopes.com are popular websites that provide fact-checking services. However, these platforms often require manual intervention and are not always accessible to the average user. Automated fact-checking systems, such as those developed by Google and Facebook, use machine learning algorithms to identify and flag fake news. However, these systems are not without their limitations and have been criticized for their lack of transparency and potential biases.

Python in Fact-Checking

Python is a versatile programming language that is widely used in data analysis, machine learning, and web development. Its simplicity and readability make it an ideal choice for developing fact-checking tools. Python's extensive library ecosystem, including libraries such as NumPy, pandas, and scikit-learn, provides a solid foundation for building sophisticated fact-checking systems.

Methodology:

System Design:

The fact-checking newspaper system is designed as a command-line interface (CLI) application. The system allows users to perform the following actions:

1. Add News Article: Users can add a news article to the system by entering the article's text. The article is stored in a list along with its verification status.

2. Show All News: Users can view all the news articles stored in the system, along with their verification status.

3. Verify News: Users can verify a news article using one of three methods: source verification, image/video analysis, or text analysis. The verification status of the article is updated based on the results of the verification process.

4. Exit: Users can exit the system.

Implementation

The system is implemented in Python using basic data structures and control flow constructs. The following sections provide a detailed explanation of the implementation.

Data Structure

The system uses a list of dictionaries to store news articles. Each dictionary contains the following keys:

- **news:** The text of the news article.
- **Verified:** A boolean value indicating whether the article has been verified.
- **status:** A string describing the verification status of the article.

- python
- news_articles = [

```
    {"news": "Sample news article 1", "verified": False, "status": "Unverified"},

    {"news": "Sample news article 2", "verified": True, "status": "Verified by Source"},

]
```

Main Loop

The system uses a `while` loop to continuously display the main menu and handle user input. The loop runs until the user chooses to exit the system.

python

```
while True:

    print("\n===== Fact-Checking Newspaper =====")

    print("1. Add News Article")

    print("2. Show All News")

    print("3. Verify News (Source, Image/Video, Text)")

    print("4. Exit")

    choice = input("Enter your choice: ")

   if choice == '1':

    Add News Article:

    elif choice == '2':

        # Show All News

    elif choice == '3':

        # Verify News

    elif choice == '4':

        # Exit

    else:

        print("Invalid choice. Please try again.")
```

Adding News Articles

When the user chooses to add a news article, the system prompts the user to enter the article's text. The article is then added to the `news_articles` list with an initial status of "Unverified."

```
if choice == '1':
    news = input("Enter the news article: ")
    news_articles.append({"news": news, "verified": False, "status": "Unverified"})
    print("News added!")
```

Showing All News

When the user chooses to show all news articles, the system iterates through the `news_articles` list and displays each article along with its verification status.

```
elif choice == '2':
    print("\nNews Articles:")
    for index, article in enumerate(news_articles):
        print(f"{index + 1}. {article['news']} - Status: {article['status']}")
```

Verifying News

When the user chooses to verify a news article, the system prompts the user to select a verification method. The available methods are:

1. Source Verification: The user is prompted to enter the source of the news article. If the source is credible (e.g., BBC, Reuters), the article is marked as verified.

2. Image/Video Analysis: The user is asked whether the image or video associated with the article appears in multiple sources. If the answer is yes, the article is marked as verified.

3. Text Analysis: The user is asked whether the text of the article matches credible sources. If the answer is yes, the article is marked as verified.

```
elif choice == '3':
    if not news_articles:
        print("No news to verify.")
        continue

    news_index = int(input("Enter the news number to verify: ")) - 1
    if 0 <= news_index < len(news_articles):
        print("Choose verification method:")
        print("1. Source Verification")
        print("2. Image/Video Analysis")
        print("3. Text Analysis")

        method = input("Enter method number: ")

        if method == '1':
            source = input("Enter source name: ")
            if source.lower() in ["bbc", "reuters", "ap", "the guardian"]:
                news_articles[news_index]["verified"] = True
                news_articles[news_index]["status"] = "Verified by Source"
            else:
                news_articles[news_index]["status"] = "Unreliable Source"

        elif method == '2':
            image_check = input("Does the image/video appear in multiple sources? (yes/no): ")
            if image_check.lower() == "yes":
                news_articles[news_index]["verified"] = True
                news_articles[news_index]["status"] = "Verified by Image/Video"
            else:
                news_articles[news_index]["status"] = "Potentially Fake Media"

        elif method == '3':
            text_check = input("Does the text match credible sources? (yes/no): ")
            if text_check.lower() == "yes":
                news_articles[news_index]["verified"] = True
                news_articles[news_index]["status"] = "Verified by Text"
            else:
                news_articles[news_index]["status"] = "Possible Fake News"


        else:
            print("Invalid method.")
            continue


        print("News verification updated!")
    else:
        print("Invalid news number.")
```

Exiting the System

When the user chooses to exit the system, the `while` loop is terminated, and the program ends.

```
elif choice == '4':
    print("Exiting Fact-Checking Newspaper.")
    break
```

Discussion

Strengths

The fact-checking newspaper system developed in this project has several strengths:

1. Simplicity: The system is easy to use and does not require any specialized knowledge or skills. Users can quickly add, view, and verify news articles using a simple command-line interface.

2. Flexibility: The system provides multiple verification methods, allowing users to choose the most appropriate method for each news article.

3. Scalability: The system can be easily extended to include additional verification methods or more sophisticated algorithms. For example, machine learning techniques could be incorporated to automate the verification process.

Limitations

Despite its strengths, the system also has several limitations:

1. Manual Verification: The system relies on manual input from the user to verify news articles. This can be time-consuming and may not be practical for large volumes of news articles.

2. Limited Credibility Assessment: The system's credibility assessment is based on a simple set of rules (e.g., checking the source name). This may not be sufficient to accurately assess the credibility of all news articles.

3. Lack of Automation: The system does not include any automated fact-checking capabilities, such as natural language processing or image recognition. This limits its ability to handle complex verification tasks.

Future Work

There are several areas for future work that could enhance the capabilities of the fact-checking newspaper system:

1. Automated Fact-Checking: Incorporating machine learning algorithms to automate the fact-checking process. For example, natural language processing techniques could be used to compare the text of a news article with credible sources.

2. Image/Video Recognition: Adding image and video recognition capabilities to automatically verify the authenticity of multimedia content.

3. User Interface: Developing a graphical user interface (GUI) to make the system more user-friendly and accessible to a wider audience.

4. Integration with Social Media: Integrating the system with social media platforms to automatically flag and verify news articles shared on these platforms.

Conclusion:

The fact-checking newspaper system developed in this project provides a simple and effective way for users to verify the credibility of news articles. While the system has some limitations, it serves as a solid foundation for future enhancements. By incorporating more advanced techniques, such as machine learning and image recognition, the system could become a powerful tool for combating the spread of fake news.

References

Mitchell, A., Gottfried, J., Barthel, M., & Shearer, E. (2016). The Modern News Consumer. Pew Research Center. Retrieved from https://www.pewresearch.org/journalism/2016/07/07/the-modern-news-consumer/

Vosoughi, S., Roy, D., & Aral, S. (2018). The spread of true and false news online. Science, 359(6380), 1146-1151. https://doi.org/10.1126/science.aap9559

Appendix: Full Python Code

```
def main():
    news_articles = []

    while True:
        print("\n===== Fact-Checking Newspaper =====")
        print("1. Add News Article")
        print("2. Show All News")
        print("3. Verify News (Source, Image/Video, Text)")
        print("4. Exit")

        choice = input("Enter your choice: ")

        if choice == '1':
            news = input("Enter the news article: ")
            news_articles.append({"news": news, "verified": False, "status": "Unverified"})
            print("News added!")

        elif choice == '2':
            print("\nNews Articles:")
            for index, article in enumerate(news_articles):
                print(f"{index + 1}. {article['news']} - Status: {article['status']}")

        elif choice == '3':
            if not news_articles:
                print("No news to verify.")
                continue

            news_index = int(input("Enter the news number to verify: ")) - 1
            if 0 <= news_index < len(news_articles):
                print("Choose verification method:")
                print("1. Source Verification")
                print("2. Image/Video Analysis")
                print("3. Text Analysis")

                method = input("Enter method number: ")

                if method == '1':
                    source = input("Enter source name: ")
                    if source.lower() in ["bbc", "reuters", "ap", "the guardian"]:
                        news_articles[news_index]["verified"] = True
                        news_articles[news_index]["status"] = "Verified by Source"
                    else:
                        news_articles[news_index]["status"] = "Unreliable Source"

                elif method == '2':
                    image_check = input("Does the image/video appear in multiple sources? (yes/no): ")
                    if image_check.lower() == "yes":
                        news_articles[news_index]["verified"] = True
                        news_articles[news_index]["status"] = "Verified by Image/Video"
                    else:
                        news_articles[news_index]["status"] = "Potentially Fake Media"
```

~~~
                elif method == '3':
                    text_check = input("Does the text match credible sources? (yes/no): ")
                    if text_check.lower() == "yes":
                        news_articles[news_index]["verified"] = True
                        news_articles[news_index]["status"] = "Verified by Text"
                    else:
                        news_articles[news_index]["status"] = "Possible Fake News"

                else:
                    print("Invalid method.")
                    continue
```
                print("News verification updated!")
            else:
                print("Invalid news number.")

        elif choice == '4':
            print("Exiting Fact-Checking Newspaper.")
            break

        else:
            print("Invalid choice. Please try again.")

if __name__ == "__main__":
    main()
~~~