SRS

1. Introduction

1.1 Purpose

The purpose of a fake news detection system is to develop and implement the technology and algorithms that are used to identify and classify false information. This involves analyzing large amounts of data and using machine learning and natural language processing techniques to create a system that can accurately distinguish between credible and fake news. The designing team is responsible for ensuring that the system is reliable, efficient, and effective in detecting fake news and reducing its spread. Additionally, the design team works to continually improve the system, taking into account new trends in misinformation and evolving user needs. By using the system, the user can determine whether the news is credible or not, and make informed decisions based on accurate information. The goal is to prevent the spread of false information and promote trust in the media.

1.2 Scope

The scope of a fake news detection system project can vary depending on the specific goals and requirements of the project. The scope of our product is to detect fake news from online articles using machine learning. Our fake news detectors purely use linguistics features to detect fake news in content. By using different machine learning models, we will detect fake news for better accuracy. Fake news has impact on decision making of many people which could lead to serious mistakes.

1.3 Definitions, Acronyms and Abbreviations

Python is a programming language that is widely used for data analysis and machine learning tasks, and it can be a useful tool for developing fake news detection systems. Python has a large community of developers and a vast collection of libraries and tools for data processing, machine learning, and natural language processing, which makes it a convenient choice for fake news detection projects.

Some of the key Python libraries and tools that can be used in fake news detection systems include:

Pandas: A library for data analysis and manipulation that provides tools for reading and writing data, cleaning and transforming data, and analyzing data.

Numpy: A library for numerical computing that provides support for arrays, matrices, and mathematical operations.

Scikit-learn: A machine learning library that provides algorithms and tools for classification, regression, clustering, and other machine learning tasks.

NLTK: A natural language processing library that provides tools for tokenizing, stemming, and lemmatizing text, as well as tools for sentiment analysis, text classification, and named entity recognition.

By using these and other Python libraries and tools, developers can build and train machine learning models that can accurately detect fake news and help prevent its spread.

Abbreviations:

1. **np –** Numpy
2. **pd –** Pandas
3. **sns -** Seaborn
4. **re –** Regular expression
5. **nltk –** Natural Language Processing Library
6. **tf –** Term Frequency

1.4 References

1.https://pdfcoffee.com/software-requiement-specifications-fake-news-detector-pdf-free.html

2.https://www.studocu.com/in/document/new-horizon-college-of-engineering/computer-science/fake-news-detection-report/38642666

3. https://www.kaggle.com/c/fake-news/data

4. https://github.com/topics/fake-news-detection

5. IEEE Std 830-1998 .IEEE Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998

6.http://www.scribd.com/doc/9138468/Software-Requirement-Specification-SrsMidtems

7.https://github.com/avani17101/Real-Time-Fake-News-Detection App/blob/master/SRS.pdf

1.5 Overview

A fake news detection system is a technology-based solution aimed at identifying false or misleading information that is spread through various forms of media, including social media platforms, news websites, and messaging apps. These systems typically use a combination of machine learning algorithms and natural language processing techniques to analyze the content and context of news articles and identify signs of misinformation. Some of the most common methods used to detect fake news include fact-checking, source analysis, and sentiment analysis. The ultimate goal of a fake news detection system is to help people differentiate between real and fake news and make more informed decisions about the information they consume.

2. The Overall Description

2.1 Product Perspective

2.1.1 System Interfaces

A system interface for a fake news detection system refers to the interface between the software components of the system and the hardware components. It includes:

- Data Input/Output (I/O) Interfaces
- Hardware Interfacing
- Network Interfacing
- Database/Dataset Interfacing

2.1.2 User Interfaces

A user can interact with various interfaces in fake news detection system which includes the following elements:

- Input Field
- Verification Button
- Result display
- Source verification
- Feedback Mechanism

2.1.3 Hardware Interfaces

There are some of the interfaces which comprises in hardware interface for fake news detection:

- Input devices
- Output devices
- Server
- Dataset
- Power Supply

2.1.4 Software Interfaces

The software interface should be designed to be modular, allowing for easy integration with other systems and applications, and be easy to use and understand for developers and end-users alike. It includes:

- Jupyter Notebook (IDLE)
- Libraries Function
- Data Input/Output function
- Command Line interface (CLI)
- Web Interface (Flask)

2.1.5 Communication Interfaces

The communication interface should be designed to be secure, reliable, and scalable, allowing for seamless communication and information exchange between the fake news detection system and other systems and applications.

- Data Formats
- Network Protocols

2.1.6 Memory Constraint

Memory constraints in a fake news detection system can refer to the limitations of the system's memory in terms of available memory, memory usage, and memory efficiency.

- Data Storage
- Algorithm Complexity
- Python Memory Management
- System Hardware

2.1.7 Site Adaptation Requirement

- Data Collection
- Data Preprocessing
- Feature Engineering
- Model Selection

- Model Training
- Model Validation
- Model Deployment
- Continuous Monitoring

2.2 Product Function

- User can login on the login page so that he/she can see their data which is stored.
- User should be able to detect fake news in multiple languages.
- User will accurately get whether the news is fake or real.
- We'll provide upload button to application so that user can upload snapshot of image of news to check its authenticity.
- Application is user-friendly, with a simple interface for users to assess news article.
- User will get an explanation of why it classified a news article as real or fake, to increase transparency and trust.
- Our application should protect user privacy by not collecting or storing sensitive information.
- Our application always stays up-to-date with the latest trends in fake news and improve its accuracy over time.

2.3 User Characteristics

- User needs to enter a readable word to search on.
- User must enter the word in the right language.
- Users' technical literacy and ability to use digital tools.

2.4 Constraints

There are various constraints which need to be balanced. Balancing these constraints and finding the right trade-offs is a crucial challenge in developing a successful fake news detection system.

1.Accuracy: The system should have high accuracy in detecting fake news to avoid false negatives and false positives.

2. Scalability: The system should be able to handle a large volume of news articles and users, and be scalable to accommodate growth and changes in user behavior.

3. Privacy and Security: The system should protect the privacy of users and the security of the data, including personal information and sensitive news content.

4. Human Verification: The system should provide a mechanism for human verification of news, to ensure that the system's decisions are not solely relied upon.

5. Time and Resource Constraints: The development and deployment of the system may be limited by time and resource constraints, such as budget and manpower.

2.5 Assumptions and Dependencies

The assumptions and dependencies for a fake news detection system can vary based on the specific system and its intended purpose, but some common ones include:

- **Reliable sources:** The system needs to have access to a set of reliable sources to compare and verify information with.
- **Data Availability:** Availability of relevant data such as news articles, social media posts, and other forms of media to build and train the system.
- **Natural Language Processing (NLP) and Machine Learning (ML) algorithms:** Fake news detection systems often use NLP and ML algorithms to analyze text, image and video content, to identify patterns and make predictions.
- **Feature Engineering:** The system relies on well-designed features to detect fake news, these features can be created through techniques such as sentiment analysis, named entity recognition, and image analysis.
- **Human annotated data:** To train and fine-tune the machine learning algorithms, large amounts of annotated data are often required, where humans label news articles as "real" or "fake".
- **Computational resources:** The system needs sufficient computational resources such as memory and processing power to process large amounts of data and run complex algorithms in real-time.
- **Access to the internet:** An internet connection is required to access and compare information from various sources and to make predictions in real-time.
- **Continuous Updating:** The system should be regularly updated with new data and fine-tuned to keep pace with changes in the ways fake news is created and spread.

2.6 Apportioning of Requirements

The requirements for a fake news detection system can be apportioned into several categories, including:

- **Data collection:** This includes the requirement to gather and store large amounts of data from various sources such as news articles, social media posts, and other forms of media.
- **Natural Language Processing (NLP) and Machine Learning (ML) algorithms:** Requirements for NLP and ML algorithms to analyze text, image and video content, identify patterns, and make predictions.

- **Feature Engineering:** Requirements for well-designed features that can effectively detect fake news, such as sentiment analysis, named entity recognition, and image analysis.
- **Human annotated data:** Requirements for annotated data to train and fine-tune the machine learning algorithms, where humans label news articles as "real" or "fake".
- **User interface:** Requirements for a user-friendly interface that allows users to easily access and use the fake news detection system.
- **Performance:** Requirements for the system to be fast and accurate in detecting fake news, with low latency and minimal false positive and false negative rates.
- **Security and privacy:** Requirements for secure storage and handling of sensitive user data, and for compliance with privacy regulations.
- **Maintenance and Support:** Requirements for continuous maintenance, updates, and support to keep the system running smoothly and to address any issues that may arise.

These requirements can vary based on the specific needs and goals of the fake news detection system, but they serve as a general guide for apportioning the necessary components and resources.

3. Specification Requirements

3.1 External Interfaces

- **Web interface:** A web interface can be created for users to interact with the fake news detection system. This interface can include features such as uploading news articles, entering keywords or phrases for search, and displaying results.
- **Mobile interface:** A mobile interface can be created to allow users to access the fake news detection system from their smartphones or tablets.
- **Browser extension:** A browser extension can be developed to integrate the fake news detection system with web browsers such as Chrome or Firefox. This can allow users to easily check the authenticity of news articles they come across while browsing the internet.
- **Input devices:** The fake news detection system may require input devices, such as a keyboard or a mouse, to allow users to input news articles or interact with the system's user interface.
- **Output devices:** The fake news detection system may require output devices, such as a monitor or a printer, to display analysis results or output reports.

- **Feedback mechanism:** The external interface should also include a mechanism for users to provide feedback on the analysis. This could include a simple "thumbs up" or "thumbs down" button to indicate whether the user agrees with the system's assessment of the article.

3.2 Functions

- **Data collection:** The system must collect data in the form of news articles from various sources, such as news websites or social media platforms.
- **Data preprocessing:** The system must preprocess the collected data to clean and normalize the text, and extract relevant features from the articles.
- **Feature extraction:** The system must extract relevant features from the preprocessed data, such as the presence of certain keywords or the sentiment expressed in the article.
- **Machine learning model training**: The system must train a machine learning model using the extracted features and a labeled dataset of genuine and fake news articles.
- **Prediction:** The trained machine learning model is then used to make predictions on new, unseen articles. The system must evaluate the predictions and assign a score or confidence level indicating the likelihood that the article is fake.
- **Visualization and interpretation:** The system must visualize and interpret the analysis results to help users understand the factors that contributed to the classification decision.
- **Feedback and improvement:** The system must provide a feedback mechanism to allow users to provide feedback on the analysis results, which can be used to improve the accuracy of the system over time.