Fake News Detection and Analysis System

S. R. S. Report – I

Group No. 38

Rishabh Sengar

(161500451)

Prashant Raghav

(161500401)

Priyank Kaushik

(161500414)

Ram Kamra

(161500437)

**Table of Contents**

**1. Introduction 1**

1.1 Purpose 1

1.2 Scope 1

1.3 Definitions, Acronyms, and Abbreviations 2

1.4 References 2

**2. General Description 3**

2.1 Product Perspective 3

2.2 Tool Used 4

2.3 General Constraints 4

2.4 Assumptions and Dependencies 4

**3. Specific Requirements 5**

3.1 Hardware and Software Interfaces 5

3.2 Use Cases 6

3.3 Classes / Objects 6

**4. Analysis Models 7**

4.1 Sequence Diagrams 7

**A. Appendices 8**

**1. Introduction**

Social media has become a popular means for people to consume news. Meanwhile, it also enables the widespread of fake news, i.e., news with intentionally false information, which brings significant negative effects to the society. Fake news and lack of trust in the media are growing problems with huge branching in our society. Obviously, a purposely misleading story is “fake news“  but lately blathering social media’s discourse is changing its definition.  
Thus, fake news detection is attracting increasing attention. However, fake news detection is a non-trivial task, which requires multi-source information such as news content, social context, and dynamic information.

* 1. **Purpose**

The Project’s purpose is to detect the 'fake news', that is, misleading news stories that comes from the non-reputable sources. And also to detect the sentiments of the news predicted.

**1.2 Scope**

The practical part of this study, as mentioned, will focus on a linguistic text  
analysis and machine learning approach to automatically detect deception in  
news articles. There are several digital methods to detect deception and many  
different classification algorithms and text analysis approaches. In order to evaluate and motivate what method will best suit the problem and data concerned  
by this study, a presentation of a few selected techniques will have to be made.  
Thus the linguistic models and machine learning algorithms that are involved in  
this project.

The parameter that will be used to construct the linguistic models and train the classification algorithms is the raw text content of a training set of news articles. This implies an assumption that the syntactic patterns of the fake news can be distinguished from other news. The idea is to train models on a dataset and from that analyze the results and effectiveness they have on discovering deception . In reality, a wide range of other parameters can be used that are not taken into consideration.

**1.3 Definitions, Acronyms, and Abbreviations**

**Server:**A program that awaits and fulfills requests from client programs in the same or  
other computers.  
Deep Learning: A machine learning method that stimulates the neural network inhuman brain.  
**Python:**​​ One of the most commonly used programming languages.  
**Classifier*:***​ An algorithm used in Statistics and Machine Learning areas to divide the data in two or more classes.  
**IEEE:***​​* Institute of Electrical and Electronics Engineers  
**NLP*:***​​ Natural Language Processing  
**API*:***​​ Application Program Interface

**1.4 References**

Wikipedia – [www.wikipedia.com](http://www.wikipedia.com/)

Software Engineering, Seventh Edition, Ian Sommerville.

Dash – [www.plotly.com/dash](http://www.plotly.com/dash)

Datacamp – [www.datacamp.com](http://www.datacamp.com)

Researchgate – [www.researchgate.com](http://www.researchgate.com)

Kdnuggets – [www.kdnuggets.com](http://www.kdnuggets.com)

Analyticsvidhya – [www.analyticsvidhya.com](http://www.analyticsvidhya.com)

Kaggle – [www.kaggle.com](http://www.kaggle.com/)

Github – [www.github.com](http://www.github.com/)

**2. General Description**

The following section describes the various requirements and modules of the Project.

**2.1 Product Perspective**

**Existing System :**

Identifying the authenticity of the news with the help of domain experts in various fields.

**Drawbacks :**

Experts hired are expensive.

Time consuming when there are millions of articles.

**Proposed System :**

To detect the unreal news with the help of machine learning techniques.

**Our Plan :**

To predict, whether the news is real or unreal.

To analyze the news articles and detect its sentiment.

**2.2 Tool Used:**

**Anaconda :** Anaconda is a free and open source distribution of the Python and R programming languages for data science and machine learning related applications, that aims to simplify package management and deployment.

**Python :** Python is an interpreted high-level programming language for general-purpose programming. Created by Guido van Rossum and first released in 1991, Python has a design philosophy that emphasizes code readability, notably using significant whitespace.

**Dash :** Dash is a Python framework for building analytical web applications. ... Built on top of Plotly.js, React, and Flask, Dash ties modern UI elements like dropdowns, sliders, and graphs to your analytical Python code.

**Web Browser :** A web browser (commonly referred to as a browser) is a software application for accessing information on the World Wide Web. Each individual web page, image, and video is identified by a distinct URL, enabling browsers to retrieve and display them on the user's device.

**2.3 General Constraints:**

This system is dataset dependent.

GUI is only in English.

This system works on single server

**2.5 Assumptions and Dependencies**

Dataset must be in English language.

Scrapped data is from limited sources.

Operating system should be Windows or Linux.

**3. Specific Requirements**

**3.1 Hardware and Software Interfaces**

**Minimum**

|  |  |
| --- | --- |
| **HARDWARE** | **SOFTWARE** |
| 1.2 GHz processor | Windows/Linux |
| 8Gb RAM | Python 3 |
| 10 GB HDD | NLTK,Scikit-learn libraries |

**Recommended**

|  |  |
| --- | --- |
| **HARDWARE** | **SOFTWARE** |
| 2GHz Processor and above | Windows/Linux |
| NVIDIA GPU(940mx and above) | Anaconda |
| 16Gb RAM and above | Python 3 |
| 50 GB HDD and above | Tensorflow,NLTK,Scikit-learn libraries |

**3.2 USE CASE Diagram**

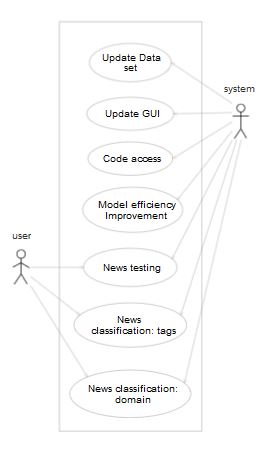
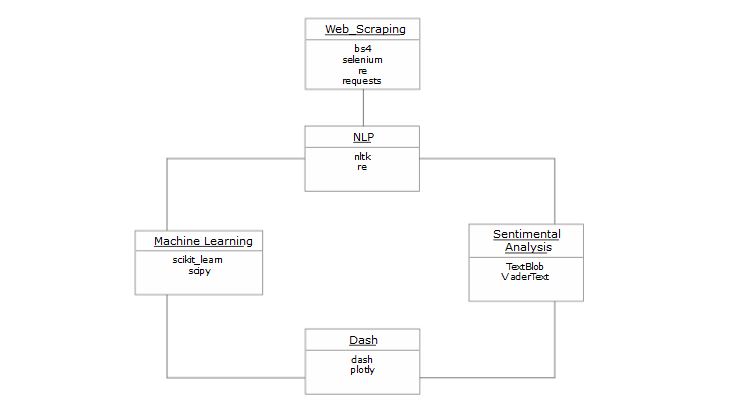
****

Fig1: Use case diagram

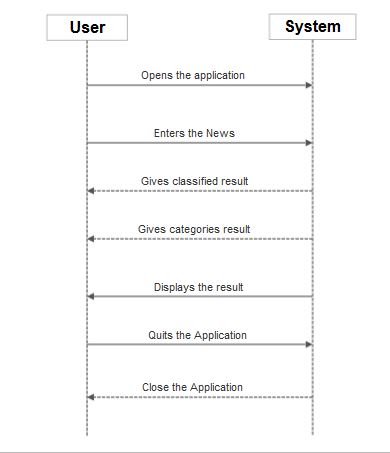
|  |  |
| --- | --- |
| **USE CASE** | **DESCRIPTION** |
| Update dataset | Admin can update dataset. |
| Update GUI | Admin can change the output patterns. |
| Code access | Admin can update/modify the code as per the requirements. |
| Model Efficiency Improvement | Admin can increase the efficiency by updating the models |
| News Testing | News provided by the user is tested by the model and gives the result as fake/real. |
| News Classification | User gets the classified news as provided. |

**3.3Class / Object Diagram**

****

|  |  |
| --- | --- |
| **Class** | **Description** |
| Web Scrapping | Data extraction from different sources to prepare dataset |
| NLP | Manipulating text data. |
| Machine learning | Data is preprocessed and model is trained on it to classify the news. |
| Sentimental Analysis | Identifying and categorizing opinions expressed in a piece of text. |

**3.4 Sequence Diagram**

****

**Appendices**

**A**

*Acronym*

*Abbreviation*

*API*

*Anaconda*

**C**

*Classification*

*Class Diagram*

*Constraints*

**D**

*Dataset*

*Dash*

*Deep Learning*

**F**

*Fake News*

**G**

*GUI*

**I**

*IEEE*

**L**

*Linux*

*Linguistic Analysis*

**M**

*Machine learning*

**N**

*NLP (Natural Language Processing)*

**P**

*Python*

**S**

*Server*

*Social Media*

*Sentiments*

*Sentimental Analysis*

*Scikit-learn*

*Sequence Diagram*

**T**

*Text analysis*

*TextBlob*

**U**

*Use Case*

**V**

*VaderText*

**W**

*Windows*

*Web Scrapping*