**FAKE NEWS DETECTION**

* **ARCITECTURE DOCUMENT**

**Contents**

Document Version Control…………………………………………………………………………...2

1. Introduction………………………………………………………………………………………....3

1.1 Project Description..................................................................................................3

1.2 Purpose of the Document.......................................................................................3

1.3 Scope of the Document…………………….…………………………………………...3

1.4 Intended Audience…………………….………………………………………………....3

2. Architecture – Fake News Framework……………………………………………………..........4

2.1 Model Pipeline WorkFlow……………………………………………………….……….4

2.2 System Architecture of proposed Fake News Detection……..……………….……...5

2.3 Web Application Framework…………………………………………………….………5

3. Server Configuration Details…………………………………………………….........................6

3.1 Model Training Server…………………………………………………………………...6

3.2 Deployment Server………………………………….………………………….…….….6

4.Dataset Details………………………………….……………………………….………….……….6

4.1 FakeNewsNet………………………………….…………………………………...….….6

4.2 ISOT………………………………….……………………………….……………………6

4.3 Fakeddit………………………………….……………………………….…….………….6

5. Technology Stack Details…………………………………...………………………….……….….7

5.1Technology Versions…………………………………...……………………………….…7

5.2 List of Python Libraries…………………...……………………………………………….7

5.3 Proposed Models……………………………………………………………………….….8

5.4 Pretrained Models………………………………………………………………………….8

**Document Version Control**

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Version | Description | Author |
|  |  |  |  |

# Introduction

## Project Description

The goal of this project is to create a graphical user interface (GUI) web application that could predict whether a news article is fake or not using trained models that have been trained over open-source datasets with the utilisation of Natural Language Processing (NLP) concepts based on the article text content.

## Purpose of the Document

The aim of this document is to provide a comprehensive architectural framework for the fake news automatic detection using machine learning and deep learning algorithms.

## Scope of the Document

This document presents the technical architecture which helps the readers to get an overall understanding of the tools and technologies, system level and web application frameworks, model workflow etc. for the development of web application.

## Intended Audience

The present document is intended to be read by the following people:

* Researchers.
* Students.
* Software Development Team.

## Architecture - Fake News Detection Framework

### MODEL PIPELINE WORKFLOW

**FAKENEWSNET**

**Text Preprocessing**

**Data Cleansing**

**FAKEDDIT**

**ISOT**

**Datasets sCollection**

**Exploratory Data Analysis**

**Train Data**

|  |
| --- |
| **Graphical user interface  Description automatically generated** |

**GUI**

**Feature Extraction**

**Model Deployment**

**Model Prediction**

**Hyperparameter Tuning**

**Evaluation**

**Test Data**

**Model Training**

**Train-Test Data Split**

### System Architecture of Proposed Fake News Detection

**Datasources**

**Model Training**

**Traditional ML Models**

1. **Random Forest**
2. **SVM**
3. **KNN**
4. **XgBoost**
5. **Naïve Bayes**
6. **Logistic Regression**

**Deep Learning Models**

1. **CNN**
2. **LSTM**
3. **Bi-LSTM**

**Fake**

**Real**

**Feature Extraction**

1. **TFIDF**
2. **Word2Vec**
3. **Glove**
4. **BERT**

**Fake**

**Text**

**Pre-Processing**

## A picture containing text Description automatically generated

**Model Deployment**

### WEB Application Framework - GUI

**Back End**

**Front End**

**Input**

**Application Logic**

Icon

Description automatically generated **A picture containing text, electronics, calculator, clipart

Description automatically generated**

**Python ,**

**Flask WebFramework**

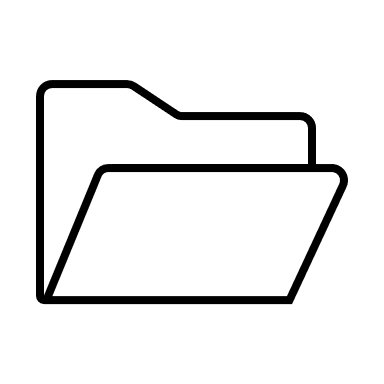
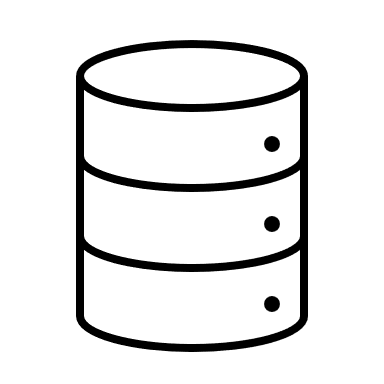
**Request**

**Users**

**Output**

**GUI / Web User Interface**

HTML, CSS

** **

**CSV Datasets**

**HTML,CSS ,Images**

**Response**

### Server Configuration Details

* 1. **Model Training Server**

|  |  |
| --- | --- |
| **Type** | **Value** |
| RAM | 50 GB |
| GPU | RTX 6000 |
| Hard Disk Space Capacity | 50 GB |

* 1. **Deployment Server**

|  |  |
| --- | --- |
| **Type** | **Value** |
| RAM | 16 GB |
| GPU | Not Needed |
| Hard Disk Space Capacity | 10 GB |

### Dataset Details

* 1. **FAKENEWSNET**

|  |  |
| --- | --- |
| **Type** | **Value** |
| Data Size | 267 MB |
| Row Records | 21k |
| Labels | Fake and Real |

* 1. **ISOT**

|  |  |
| --- | --- |
| **Type** | **Value** |
| Data Size | 110 MB |
| Row Records | 32k |
| Labels | Fake and Real |

* 1. **FAKEDDIT**

|  |  |
| --- | --- |
| **Type** | **Value** |
| Data Size | 2 GB |
| Row Records | 800k |
| Labels | Fake and Real |

### Technology Stack Details

* 1. **Technology Versions**

|  |  |
| --- | --- |
| **Technology** | **Versions** |
| Python | 3.7 |
| Flask | 2.1.2 |
| TensorFlow Keras | 2.9.0 |
| HTML | HTML5 |
| CSS | CSS3 |

* 1. **List of Python Libraries**

|  |  |
| --- | --- |
| **Libraries** | **Versions** |
| Numpy | 1.21.5 |
| Pandas | 1.3.4 |
| Matplotlib | 3.5.1 |
| Seaborn | 0.11.2 |
| sklearn | 1.0.2 |
| Xgboost | 1.6.1 |
| NLTK | 3.7 |
| Wordcloud | 1.8.1 |
| Gensim | 4.1.2 |
| bert | 0.14.9 |
| tldextract | 3.2.0 |
| BeautifulSoup | 4.11.1 |

* 1. **Proposed Models**

|  |  |
| --- | --- |
| **Machine Learning Models** | **Deep Learning Models** |
| Random Forest | 1 D Convolutional Neural Network |
| SVM | Long Short-Term Memory |
| XgBoost | Bidirectional Long Short Term Memory |
| KNN |  |
| Naïve Bayes |  |
| Logistic Regression |  |

* 1. **Pretrained Models**

|  |  |
| --- | --- |
| **Pretrained Model Name** | **Download URL** |
| bert\_en\_uncased\_L-12\_H-768\_A-12/1 | https://tfhub.dev/tensorflow/bert\_en\_uncased\_L-12\_H-768\_A-12/1 |
| Word2Vec - **GoogleNews-vectors-negative300.bin** | https://s3.amazonaws.com/dl4j-distribution/GoogleNews-vectors-negative300.bin.gz |
| Glove - **glove.6B.300d.txt** | http://nlp.stanford.edu/data/glove.6B.zip |