# Fake News Detection Model Documentation

## Table of Contents


**1. Project Overview**
The Fake News Detection Model is designed to classify news articles as either "real" or "fake" based on their textual content and additional features. This project provides a brief overview of the model, its objectives, and how to use it.

**2. Problem Statement**
The primary problem statement of this project is to develop a model capable of distinguishing between real and fake news articles. The model analyzes the textual content of articles to make this classification.

**3. Objectives**
- Acquire a labeled dataset of news articles.
- Preprocess the data to prepare it for analysis.
- Extract relevant features from the text using techniques such as TF-IDF.
- Use sentiment analysis as an additional feature.
- Train a machine learning classification model.
- Evaluate the model's performance using multiple metrics.
- Identify the limitations of the model.

**4. Features and Techniques**
- Text Preprocessing: The model performs text preprocessing steps, including lowercasing and stopword removal.
- Sentiment Analysis: It uses VADER SentimentIntensityAnalyzer to analyze the sentiment of the article's text.
- Additional Feature: The model also considers the length of the text as an additional feature.

**5. Requirements**
- Python 3
- Required Python libraries mentioned in the code

**6. Usage**
1. Clone this repository:
```bash
git clone <repository-url>
cd fake-news-detection-model

**Run the model:**

python fake_news_detector.py

**7. Example**
Consider the following example sentences:
True News Sentence:
"Scientists have discovered a new species of butterfly in the Amazon rainforest."

Fake News Sentence:
"Aliens have landed in New York City and are controlling our minds with their advanced technology."

Running the model with these sentences will classify the first as "real" and the second as "fake."

**8. Model Details**
This model is a simplified example and may not be suitable for production use. Further fine-tuning and extensive data are required for a robust solution.
The model is designed for classification and may not verify the recency of news articles. Additional checks are needed for real-time news validation.

**9. Results and Evaluation**
Include details of the model's performance and evaluation metrics, along with any insights gained from the results.

**10. Model Limitations**
Identify any limitations or constraints of the model, such as data limitations, performance issues, or areas for improvement.
```