



















# Fake News Detector - AI-Powered Verification System

An intelligent web application powered by **Machine Learning, Flask,** and **MongoDB** to detect and classify fake news articles in real time. The project includes a secure admin panel with OTP verification, a feedbackdriven model retraining system, and an elegant UI experience.

## Features
















-  **Real-time Prediction** of fake or real news articles
-  **Feedback Collection** for user satisfaction (Yes/No)
-  **Admin Panel** with secure OTP-based login
-  **Secure Email-based Access** for approved admin emails only
-  **Model Retraining** using approved feedback
-  Live database with **MongoDB Atlas**
-  **Email Integration** using Flask-Mail & Gmail SMTP
-  Time stamped logs of predictions with confidence score
-  Beautiful, responsive UI with animated visuals (waves, particles, fireflies)

## Tech Stack

Component	Technology
Backend	 PYTHON  FLASK
ML Model	 LOGISTIC REGRESSION  TF-IDF
Database	 MONGODB (ATLAS)
Email Service	 FLASK-MAIL (SMTP)
Frontend	 HTML5  CSS3  JAVASCRIPT

## Folder Structure

### ▼ FAKE NEWS PREDICTOR

- > \_\_pycache\_\_
- > .env
- ▼ static
  - ▼ CSS
    - # admin\_dashboard.css
    - # admin\_login.css
    - # F\_News.css
    - # index.css
    - # R\_News.css
    - # verify\_otp.css
  - ▼ Images
    - > F\_News
    - > R\_News
    -  F\_News.png
    -  LOGO.jpg
    -  Logo.png
    -  LOGO1.jpg
  - ▼ JS
    - JS admin\_dashboard.js
    - JS admin\_login.js
    - JS verify\_otp.js
  - ▼ templates
    - <> admin\_dashboard.html
    - <> admin\_login.html
    - <> F\_News.html
    - <> index.html
    - <> R\_News.html
    - <> verify\_otp.html
-  .env
-  .gitattributes
-  .gitignore
-  fake\_news\_pipeline.py
-  Fake.csv
-  Home.py
-  model.joblib
-  news\_dataset.csv
-  Readme.pdf
-  True.csv
-  vectorizer.joblib

U

## Setup Instructions

### 1. Clone the Repository

```
git clone https://github.com/atribiswas03/fake-news-detector-ai.git  
cd fake-news-detector-ai
```

### 2. Download Required Files (**Mandatory**)

Download the following CSV files and place them in the project root directory as shown in the folder structure :

- ♠ Fake.csv - [Download](#)
- ♠ True.csv - [Download](#)
- ♠ news\_dataset.csv - [Download](#)

### 3. Create a Virtual Environment

```
python -m venv venv  
venv\Scripts\activate # Windows  
# OR  
source venv/bin/activate # macOS/Linux
```

### 4. Install Dependencies

```
pip install -r requirements.txt
```

### 5. Create .env File

```
touch .env
```

Add the following :

```
MONGO_URI=your_mongo_uri  
MAIL_USERNAME=your_email@gmail.com  
MAIL_PASSWORD=your_app_password  
MAIL_DEFAULT_SENDER=your_email@gmail.com
```

## 6. Run the App

`python Home.py`

Then open your browser and visit : <http://127.0.0.1:8000>



## Admin Panel Features

- ♣ View logs of all predictions with confidence levels
- ♣ Approve or reject user feedback entries
- ♣ Automatically retrain model with approved samples
- ♣ Add/delete admin emails (admin-only)
- ♣ OTP-based secure login
- ♣ Delete single or all prediction logs



## Contact

Developed by **Atri Biswas**

♣ Email : [atribiswas2003@gmail.com](mailto:atribiswas2003@gmail.com)

♣ LinkedIn : [Atri Biswas](#)



## License

This project is open-source and available under the [MIT License](#).



## IMPORTANT STEPS

### 1. Run in VS Code Terminal

`pip install flask pymongo flask-mail python-dotenv pandas nltk scikit-learn joblib`

### 2. Delete the following two files

a. **vectorizer.joblib**

b. **model.joblib**

### 3. Run **fake\_news\_pipeline.py** in VS Code

This may take a little time. After successful execution, it will automatically generate the above two files.)

### 4. Now run **Home.py**

The model will continue to improve over time based on user feedback and admin approval.

**Note :** Proper training and consistent feedback help improve prediction accuracy.

## **Show your support**

If you like this project, don't forget to give it a **star**  and share it with others!