

# Documentation for Web app

## Objective

This project builds a machine learning web application that classifies tweets or headlines as Fake or Real news using NLP and a trained classifier.

Folder Structure:

```
Web app/
├── app.py
├── model.pkl
├── vectorizer.pkl
├── requirements.txt
├── templates/
│   └── model_web.html
```

.pkl will be generated after you convert trained\_model.py into trained\_model.ipynb and run the model.

## Model Summary

- Vectorizer: TF-IDF
- Classifier: Logistic Regression
- Input: text
- Output: Fake (0) or Real (1)

## How to Run Locally

1. Clone/download the project
2. Run: `pip install -r requirements.txt`
3. Start the app: `python app.py`
4. Open in browser: <http://127.0.0.1:5000/>

# Documentation for Web app

## How It Works

User submits text → TF-IDF → Logistic Regression → Output prediction

## Example Prediction

Input: "NASA confirms the sun is fake!"

Output: Real News



The screenshot shows a web browser window with the address bar displaying '127.0.0.1:5000/predict'. Below the address bar, there are several navigation links: 'Gmail', 'NotebookLM', 'Advanced Learning...', and 'Download YouTube...'. The main content area of the browser displays the title 'Fake News Detection' in a large, bold, black serif font. Below the title is a text input field containing the text 'NASA confirms the sun is fake!'. To the right of the input field, there is a small icon of a lightbulb inside a green circle, followed by a green circle containing a white 'G'. Below the input field is a button labeled 'Predict'. Below the button, the text 'Prediction: Real News' is displayed in a large, bold, black serif font.

## References

- Libraries: [Flask](#), [scikit-learn](#), [nltk](#), [pandas](#)