Implementation of Project

Download Project Files:

Download all project files from GitHub link provided.

extract and Open the Project in VS Code:

Launch VS Code.

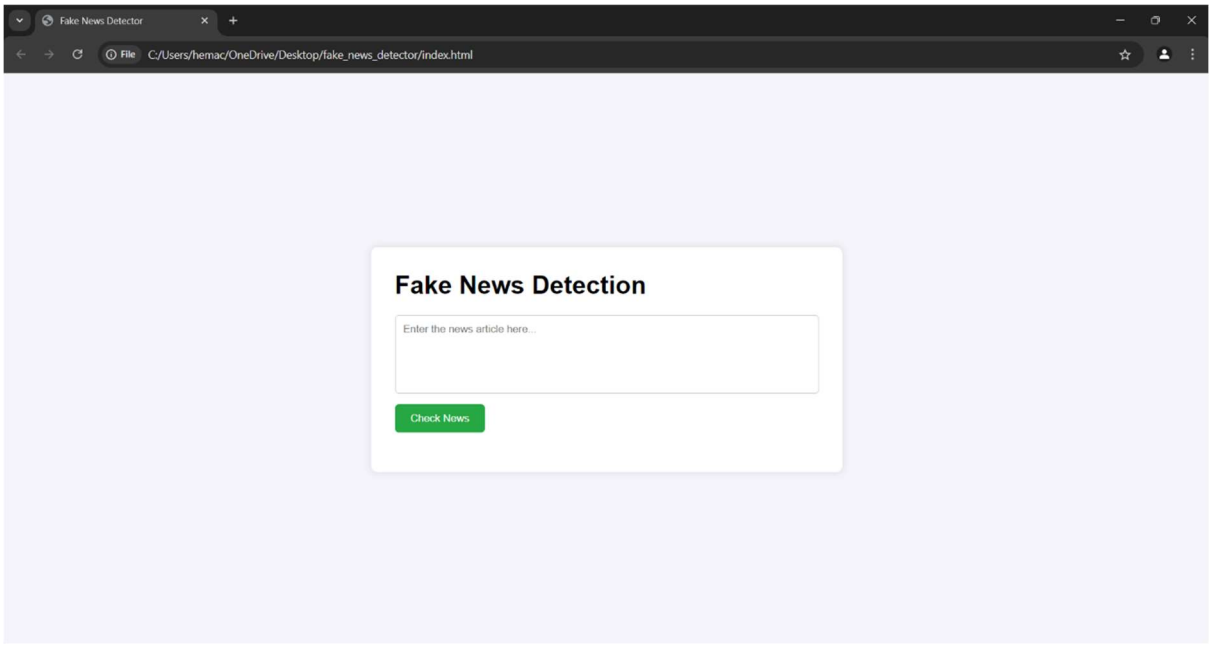
Open the project folder by selecting File > Open Folder and choosing the folder containing your downloaded files.

Open index.html in VS Code:

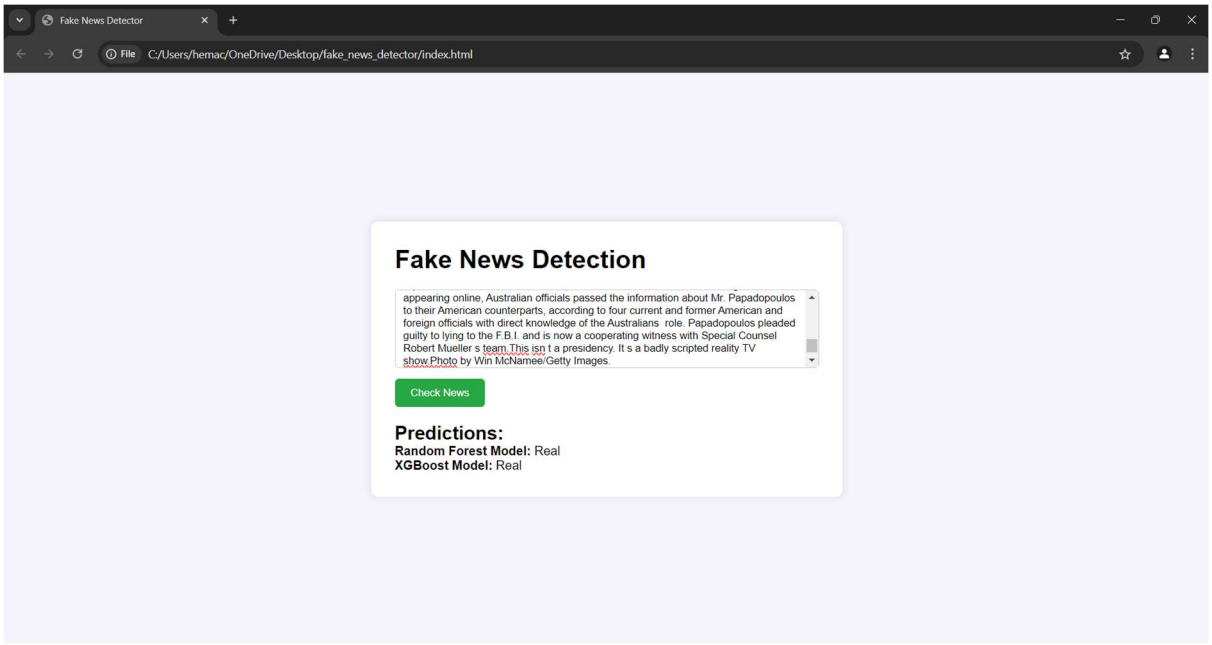
Locate the index.html file in the VS Code file explorer.

Right-click on index.html and Run it by clicking start debugging it will open in a web browser and you can take any news article from the True (1) and Fake (1) CSV files and paste it, it will display the output.

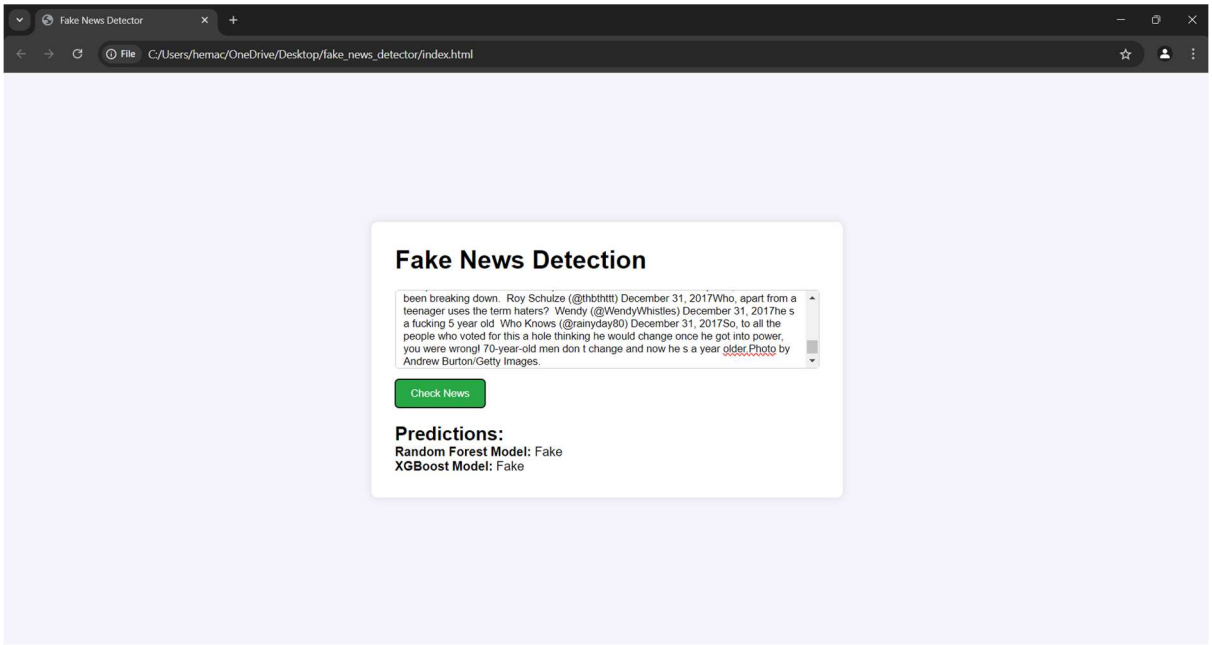
here is the outputs for how the project works:



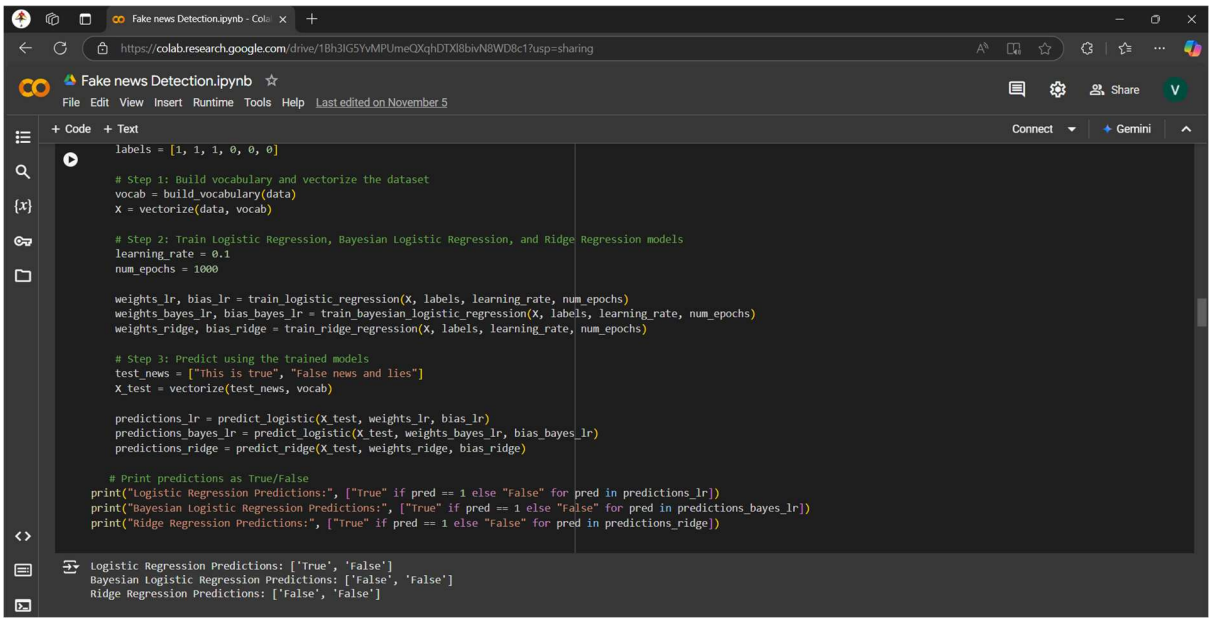
Prediction for Real news using Random Forest Model and XGBoost Model:



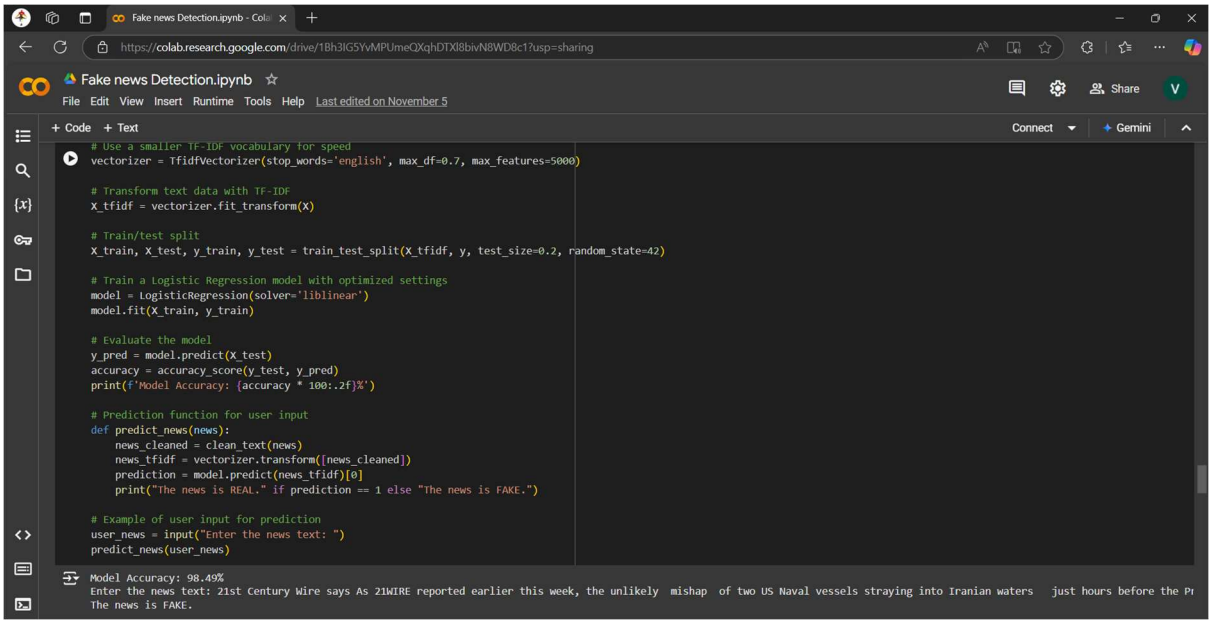
Prediction for Fake news using Random Forest Model and XGBoost Model:



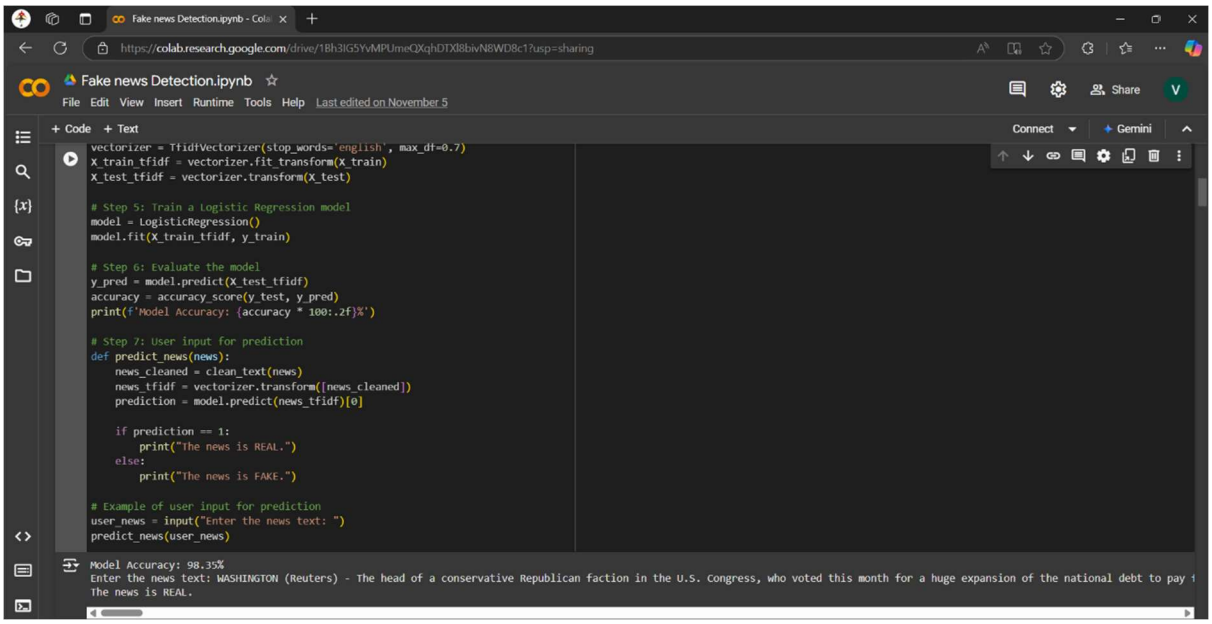
Prediction for Fake news using Logistic Regression, Bayesian Logistic Regression and Ridge Regression Model in Collab : These 3 Regression models can be implemented in the google collab.



For Fake news:



For Real News:



Comparative Analysis:

1. **Algorithm Complexity**:
    - **First Code (Random Forest & XGBoost)**: Both models are complex ensemble methods suitable for high-dimensional and non-linear datasets. XGBoost is more optimized and generally more accurate but requires careful parameter tuning.
    - **Second Code (Logistic, Bayesian Logistic, Ridge Regression)**: These are simpler linear models, with Bayesian Logistic Regression providing a probabilistic framework and Ridge Regression adding regularization to standard logistic regression.
2. **Performance and Use Cases**:
    - **Random Forest & XGBoost**: Preferred for datasets where capturing complex patterns is crucial, like text data in fake news detection, where relationships between words can be non-linear.
    - **Logistic Regression**: Works well for simpler problems with linear decision boundaries and when interpretability is needed.
    - **Bayesian Logistic Regression**: Useful in situations where uncertainty quantification is essential.
    - **Ridge Regression**: A good choice for high-dimensional datasets where overfitting is a risk.
3. **Computational Efficiency**:
    - **First Code**: More computationally intensive due to ensemble models. XGBoost, while faster than traditional boosting, still requires significant resources.
    - **Second Code**: More efficient and faster to train, especially useful for quick experimentation or when deploying on limited hardware.

4. **Interpretability**:
    - **Random Forest & XGBoost**: Less interpretable, though feature importance metrics can provide some insights.
    - **Logistic Regression & Ridge Regression**: Highly interpretable, allowing an understanding of feature impacts.
    - **Bayesian Logistic Regression**: Provides additional probabilistic interpretation, valuable for decision-making under uncertainty.
5. **Flexibility**:
    - **First Code**: Offers flexibility in capturing complex, non-linear relationships.
    - **Second Code**: Simpler models that are easier to interpret but less flexible for complex data patterns.

Summary:

- **If your goal** is to achieve high accuracy and you have sufficient computational resources, the **ensemble models (Random Forest & XGBoost)** in the first code are more suitable.
- **For simpler, faster, and more interpretable models**, the **regression models (Logistic, Bayesian, Ridge)** in the second code are preferable.
- **Bayesian Logistic Regression** adds value when understanding uncertainty is essential, while **Ridge Regression** is beneficial for high-dimensional datasets prone to overfitting.