News Articles Share Prediction using Machine Learning

Project Overview

Business Overview

Our project aims to predict the number of shares news articles will receive on various online platforms. News agencies, bloggers, and content creators are always interested in understanding how popular their articles will be. Predicting the number of shares can help them tailor their content and promotional strategies.

Aim

To predict the number of shares news articles will receive using machine learning algorithms based on various article attributes.

Data Description

The dataset comprises news articles and their associated attributes, such as title, text, publication date, keywords, and more. These attributes can be used to predict the article's shareability.

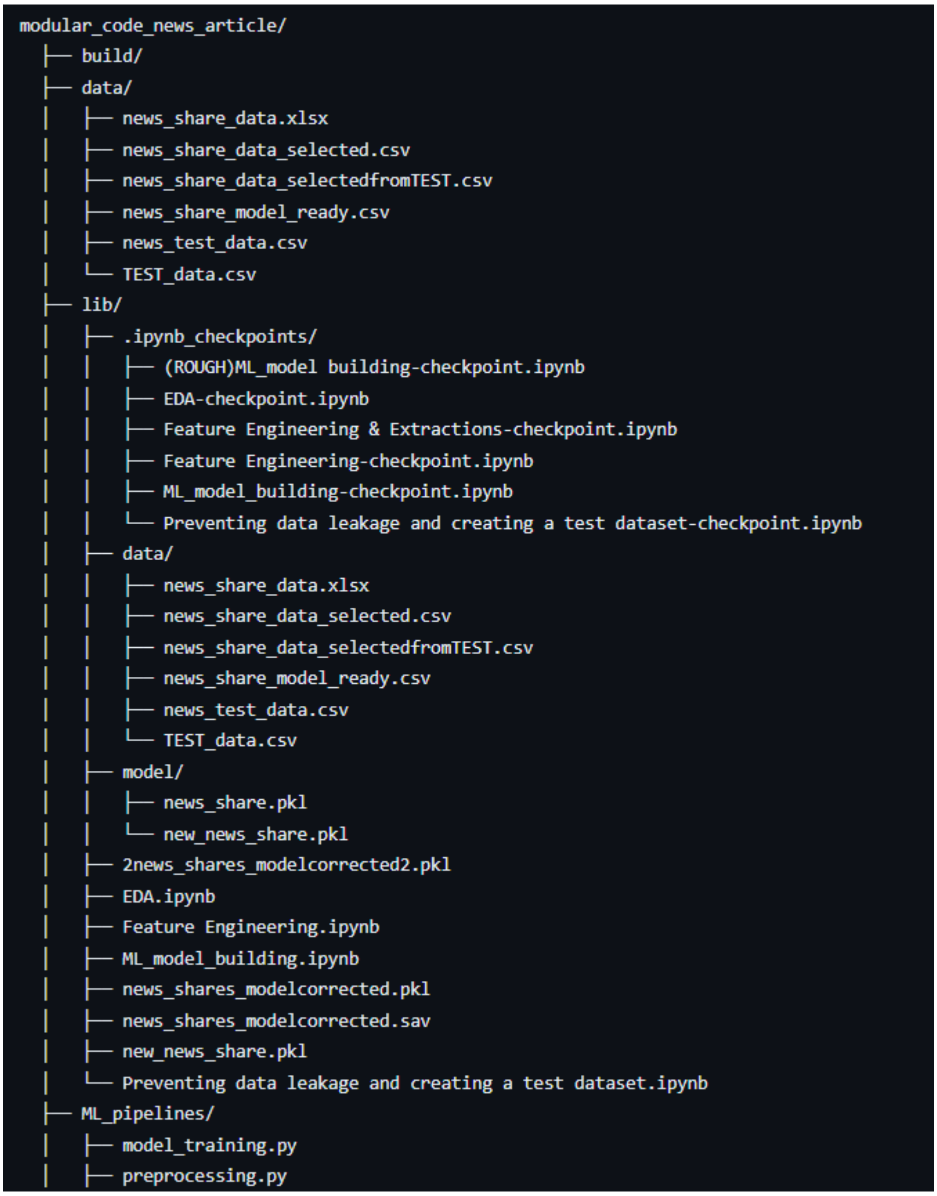
Tech Stack

- **Language**: Python
- **Libraries**: pandas, numpy, scikit-learn, nltk, matplotlib, seaborn

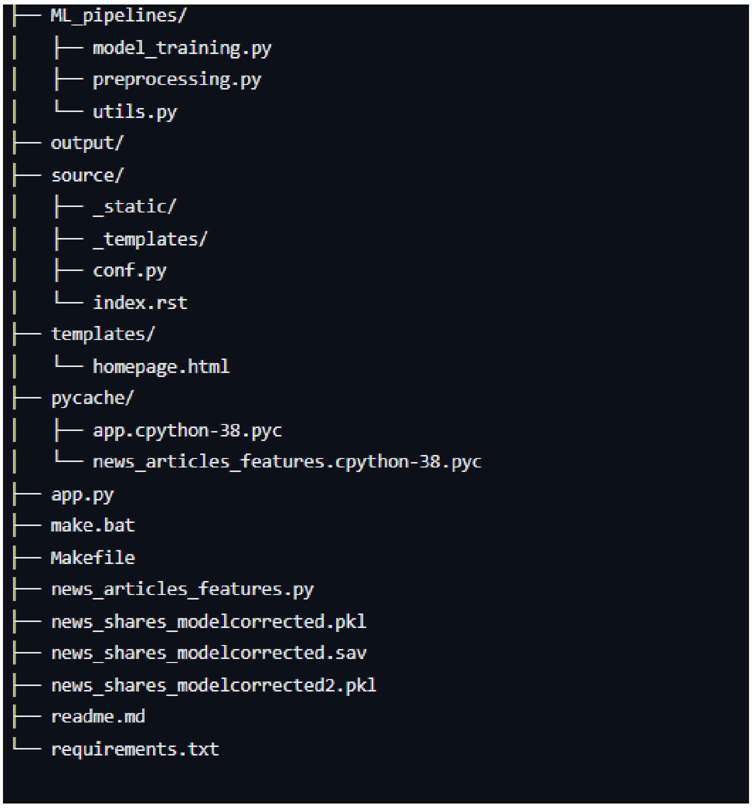
Approach

1. Data Collection
    - Gathered a dataset of news articles with attributes.
2. Data Preprocessing
    - Text cleaning
    - Feature extraction
    - Handling missing data
    - Categorical Data Cleaning
    - Using Regex Library
    - Univariate Data Analysis
    - Multivariate Data Analysis
    - Outlier Treatment
    - Feature Extraction
    - Text Data Processing
    - Parts of Speech Tagging
    - Count Vectorization and N-grams
3. Exploratory Data Analysis (EDA)
    - Understand data distributions
    - Identify correlations
4. Feature Engineering
    - Create new features from existing data
5. Machine Learning Models
    - Build and train machine learning models to predict shares.
    - Random Forest Regressor
6. Model Evaluation
    - Evaluate model performance using metrics such as Mean Absolute Error (MAE), R2 Squared and Root Mean Squared Error (RMSE).
7. Model Deployment
    - APIs
    - Web Application Development using FastAPI
    - Render Deployment

Modular Code Overview:



```
modular_code_news_article/
  ├── build/
  ├── data/
  │   ├── news_share_data.xlsx
  │   ├── news_share_data_selected.csv
  │   ├── news_share_data_selectedfromTEST.csv
  │   ├── news_share_model_ready.csv
  │   ├── news_test_data.csv
  │   └── TEST_data.csv
  ├── lib/
  │   ├── .ipynb_checkpoints/
  │   │   ├── (ROUGH)ML_model building-checkpoint.ipynb
  │   │   ├── EDA-checkpoint.ipynb
  │   │   ├── Feature Engineering & Extractions-checkpoint.ipynb
  │   │   ├── Feature Engineering-checkpoint.ipynb
  │   │   ├── ML_model_building-checkpoint.ipynb
  │   │   └── Preventing data leakage and creating a test dataset-checkpoint.ipynb
  │   ├── data/
  │   │   ├── news_share_data.xlsx
  │   │   ├── news_share_data_selected.csv
  │   │   ├── news_share_data_selectedfromTEST.csv
  │   │   ├── news_share_model_ready.csv
  │   │   ├── news_test_data.csv
  │   │   └── TEST_data.csv
  │   ├── model/
  │   │   ├── news_share.pkl
  │   │   └── new_news_share.pkl
  │   ├── 2news_shares_modelcorrected2.pkl
  │   ├── EDA.ipynb
  │   ├── Feature Engineering.ipynb
  │   ├── ML_model_building.ipynb
  │   ├── news_shares_modelcorrected.pkl
  │   ├── news_shares_modelcorrected.sav
  │   ├── new_news_share.pkl
  │   └── Preventing data leakage and creating a test dataset.ipynb
  ├── ML_pipelines/
  │   ├── model_training.py
  │   ├── preprocessing.py
```



```
├── ML_pipelines/
│   ├── model_training.py
│   ├── preprocessing.py
│   └── utils.py
├── output/
├── source/
│   ├── _static/
│   ├── _templates/
│   ├── conf.py
│   └── index.rst
├── templates/
│   └── homepage.html
├── pycache/
│   ├── app.cpython-38.pyc
│   └── news_articles_features.cpython-38.pyc
├── app.py
├── make.bat
├── Makefile
├── news_articles_features.py
├── news_shares_modelcorrected.pkl
├── news_shares_modelcorrected.sav
├── news_shares_modelcorrected2.pkl
├── readme.md
└── requirements.txt
```

Once you unzip the modular_code.zip file, you can find the following folders within it. They are:

1. The **build** directory doesn't have specific files listed. It was used for build artifacts or temporary files during development.
2. The **data** directory contains datasets used for analysis, including:
   - "news_share_data.xlsx"
   - "news_share_data_selected.csv"
   - "news_share_data_selectedfromTEST.csv"
   - "news_share_model_ready.csv"

   "news_test_data.csv"
   "TEST_data.csv"
3. The **lib** directory contains Python code and notebooks used for analysis, including:
   - Jupyter Notebook checkpoints in the ".ipynb_checkpoints" subdirectory.
   - "EDA.ipynb" for exploratory data analysis.
   - "Feature Engineering.ipynb" for feature engineering.
   - "ML_model_building.ipynb" for building machine learning models.

   "Preventing data leakage and creating a test dataset.ipynb" for data preprocessing. "news_share.pkl" and "new_news_share.pkl" for saved machine learning models.
4. The **ML_pipelines** directory contains Python files related to machine learning pipelines, including:
   - "model_training.py" for training models.
   - "preprocessing.py" for data

   preprocessing. "utils.py" for utility functions.
5. The **output** directory was used to store output files generated during analysis but now it's blank and shifted to root directory for some problems.

6. The **templates** directory contains HTML templates for the web application.
7. The **pycache** directory is automatically generated and contains Python bytecode files.
8. "**app.py"** is the main Python script for the FastAPI web application.
9. "**news_articles_features.py"** is a Python script related to news article features.
10. "**news_shares_modelcorrected.pkl**" and "**news_shares_modelcorrected.sav"** are saved machine learning models.
11. **"news_shares_modelcorrected2.pkl"** is another saved machine learning model.
12. **"readme.md"** is the documentation or readme file for the project.
13. **"requirements.txt"** lists the required Python packages for the project.
14. The **requirements**.txt file has all the required libraries with respective versions. Kindly install the file by using the command pip install -r requirements.txt
15. **All the instructions for running the code are present in readme.md file**