README.md

Sarcasm Detection in Reddit Comments by CS3244 PG16

Overview

This project focuses on building machine learning-based sarcasm detection models to analyze user comments on the subreddits r/worldnews, r/news, and r/politics. By detecting sarcasm in textual data, the project aims to enhance the understanding of public sentiment, providing value to industries such as journalism, social media monitoring, and natural language processing (NLP) applications. Ideally, this model will eventually be able to be incorporated into chatbots to enhance their sentiment detection abilities and provide the most appropriate response to different situations.

Datasets

The training and test data sets were extracted from Kaggle and Princeton University Website. The datasets comprise labelled (sarcastic and non-sarcastic) comments scraped from Reddit, which may contain biases. Reddit comments are subjective opinions or reactions of users that may not be fact-checked. The dataset contains about 1 million sarcastic comments (1 million rows) and ten features.

Note: The training dataset is downloaded from Kaggle, while the test dataset is downloaded from the Princeton University website, which contains the correct test dataset.

Important Note:

GitHub imposes a maximum size limit for individual files in a repository: 100 MB per file. Repositories have a soft limit of 1 GB (though larger repositories can cause performance issues).

As an alternative:
On your local machine, download the datasets from this Google Drive Folder into the `Datasets` folder.

The `Datasets` directory should contain:

- `sentiment.csv` : Processed dataset for model training
- `sentiment_test.csv` : Processed dataset for final model testing/evaluation
- `sentiment_bigram_final.csv` : Processed datasets with top 50 bigrams
- `train-balanced-sarcasm.csv` : Raw train dataset before cleaning
- `test-balanced.csv` : Raw test dataset before cleaning
- `bert_embeddings_no_pooling_train.pkl` : BERT embeddings with no pooling used during model training

Exploratory Data Analysis (EDA)

The `EDA` folder includes Jupyter notebooks to explore sarcasm data across various domains:

- `Sarcasm EDA.ipynb` : General sarcasm exploration of the three subreddits (r/news, r/politics and r/worldnews)
- Topic-specific EDA:
    - `Sarcasm EDA (news).ipynb` : Focused on subreddit r/news only
    - `Sarcasm EDA (politics).ipynb` : Focused on subreddit r/politics only
    - `Sarcasm EDA (worldnews).ipynb` : Focused on subreddit r/wordlnews only

For our EDA, we mainly focused on `train-balanced-sarcasm.csv` to understand data characteristics, identify data quality issues in order to develop effective data cleaning procedures.

Feature Engineering and Data Cleaning

The `Feature Engineering : Data Cleaning` folder contains:

- `Train_Sarcasm Data Cleaning and Preprocessing.ipynb` : Preprocessing text and feature engineering on train dataset for sarcasm analysis
- `Train_Sarcasm Data Cleaning and Preprocessing.ipynb` : Preprocessing text and feature engineering on test dataset for sarcasm analysis
- `Bigrams_Data_Cleaning.ipynb` : Preprocessing training dataset for sarcasm analysis, but keeping the top 50 most common bigrams identified
- `Bert Embedding.ipynb` : Code used to generate the `bert_embeddings_no_pooling_train.pkl` under `Datasets` directory

Models

The `Models` folder includes implementations of the following machine learning algorithms:

- **Decision Tree**: `Decision Tree.ipynb`
- **K-Nearest Neighbors (KNN)**: `KNN.ipynb`
- **Logistic Regression**: `Logistic_Regression.ipynb`
- **Neural Network**: `Neural_Network.ipynb`
- **Random Forest**: `Random_forest.ipynb`
- **Support Vector Machine (SVM)**: `SVM.ipynb`

Important Files under Root Folder

- `PG-16 Presentation Slides.pdf` : Presentation slides for this project
- `Statement of Independent Work and References.pdf` : Declaration of Independent Work and Project References
- `TEAMMATES Proof of Submission` : Folder containing peer evaluations

Team Members

Nguyen Huy Dat, A0258929H

Jeong Youngkyu, A0252154M

Odele Pang Kun Ting, A0245935W

Ethan Yeo Alsagoff, A0240231B

Low Mei Lin, A0240908E

Yap Yi Pin, A0258069R