**MS in Applied Data Science - Portfolio**

**Name:** Sukhad Dnyanesh Joshi

* **Course: IST 664 - Natural Language Processing**

Sentiment Classification of Movie Reviews:-

**Project Overview:**

This project involves developing a sentiment classification model to analyze over 10,000 movie reviews. It leverages text normalization, vectorization techniques, and classification models to identify review sentiment (positive or negative). Extensive feature engineering was conducted using sentiment lexicons like LIWC and MPQA to extract emotionally meaningful features. Multiple machine learning algorithms were trained and evaluated to achieve the best sentiment prediction performance. The project highlights the impact of feature selection and explainable models on natural language understanding.

**Files in This Folder:**

|  |  |
| --- | --- |
| **File** | **Description** |
| corpus | Text corpus of movie reviews |
| classifyKaggle.py | Script for building and evaluating classifiers |
| crossval.py | Script for cross-validation experiments |
| filtered.csv (it includes multiple csv’s). | Processed feature datasets (bigram, POS, LIWC, etc.) |
| unfiltered.csv ((it includes multiple csv’s). | Raw feature datasets before filtering |
| sentiment\_read\_LIWC\_pos\_neg\_words.py | Script for LIWC sentiment processing |
| sentiment\_read\_subjectivity.py | Script for subjectivity feature extraction |
| README.md (IST 664\_Walkthrough) | Project overview and instructions |
| SentimentLexicons | Sentiment word lists (positive/negative) |

**GitHub Repository:**

<https://github.com/SukhadJoshi/MS-ADS-Portfolio_Sukhad-Dnyanesh-Joshi>

**Software Requirements:**

* Python 3.10 or later
* Jupyter Notebook
* Libraries: pandas, nltk, scikit-learn, matplotlib, numpy

**How to Review:**

1. Explore the corpus and SentimentLexicons folders for raw data and sentiment word lists.
2. Run sentiment\_read\_LIWC\_pos\_neg\_words.py and sentiment\_read\_subjectivity.py to generate feature-based inputs.
3. Use classifyKaggle.py to train models on different feature sets.
4. Apply crossval.py for cross-validation analysis across different datasets.
5. Examine filtered and unfiltered .csv files for feature engineering outputs.