Sentiment Analysis

Using Support Vector Machines

Vardan S Kamra

Abstract

Sentiment analysis, a subfield of natural language processing (NLP), plays a crucial role in understanding the opinions and emotions expressed in textual data. In this project, we explore the application of Support Vector Machines (SVMs) in sentiment analysis. SVMs are powerful supervised learning models known for their effectiveness in classification tasks.

Our project focuses on training SVMs using a dataset comprising online reviews, tweets, and other textual sources. The dataset is labelled with sentiment labels such as positive, negative, and neutral. By leveraging the features extracted from the text, SVMs learn to classify new instances into one of these sentiment categories.

We preprocess the textual data by removing noise, tokenizing the text, and applying techniques such as stemming or lemmatization to normalize the text. Feature engineering plays a vital role in SVMs, where we represent each text instance as a vector in a high-dimensional space. This vectorization process transforms the text into a format suitable for SVM classification.

We train the SVM model on a portion of the dataset and evaluate its performance using various metrics such as accuracy, precision, recall, and F1-score. Additionally, we employ techniques like cross-validation to ensure the robustness of our model and to prevent overfitting.

Through experimentation and analysis, we demonstrate the effectiveness of SVMs in sentiment analysis tasks. Our project contributes to the growing body of research in NLP and provides insights into the practical applications of machine learning algorithms, particularly SVMs, in understanding and analysing sentiment in textual data from diverse sources.

Contents

1. Project Overview

1.1 Objective of the Project

The primary objective of this project is to delve into the practical utilization of Support Vector Machines (SVMs) within the realm of sentiment analysis. By harnessing a diverse dataset encompassing online reviews, tweets, and an array of textual sources, the project endeavours to train SVM models effectively. Through this exploration, the project aims to contribute significantly to the field of Natural Language Processing (NLP), shedding light on the intricate nuances of machine learning algorithms' application in sentiment analysis tasks. By elucidating these practical applications, the project seeks to offer comprehensive insights into the intricate interplay between SVMs and sentiment analysis, thereby enriching the landscape of computational linguistics research.

1.2 Brief Description of the Project

The project entails a comprehensive process of training Support Vector Machine (SVM) models to effectively classify the sentiment of textual data into distinct positive, negative, or neutral categories. This intricate process begins with preprocessing the textual data, which involves essential steps such as tokenization, elimination of stopwords, and stemming to refine the dataset. Subsequently, the refined dataset is partitioned into training and testing sets to facilitate robust model evaluation. Leveraging the training data, SVM classifiers are meticulously trained to recognize patterns and relationships within the textual data. Once the models are adequately trained, they are deployed to predict the sentiment of new textual inputs, thereby offering valuable insights into sentiment analysis across diverse textual sources.

1.3 Technology Used

1.3.1 Hardware Used

- Core i5-11400H @ 2.70GHz
- 8 GB DDR4 RAM
- GTX 1650

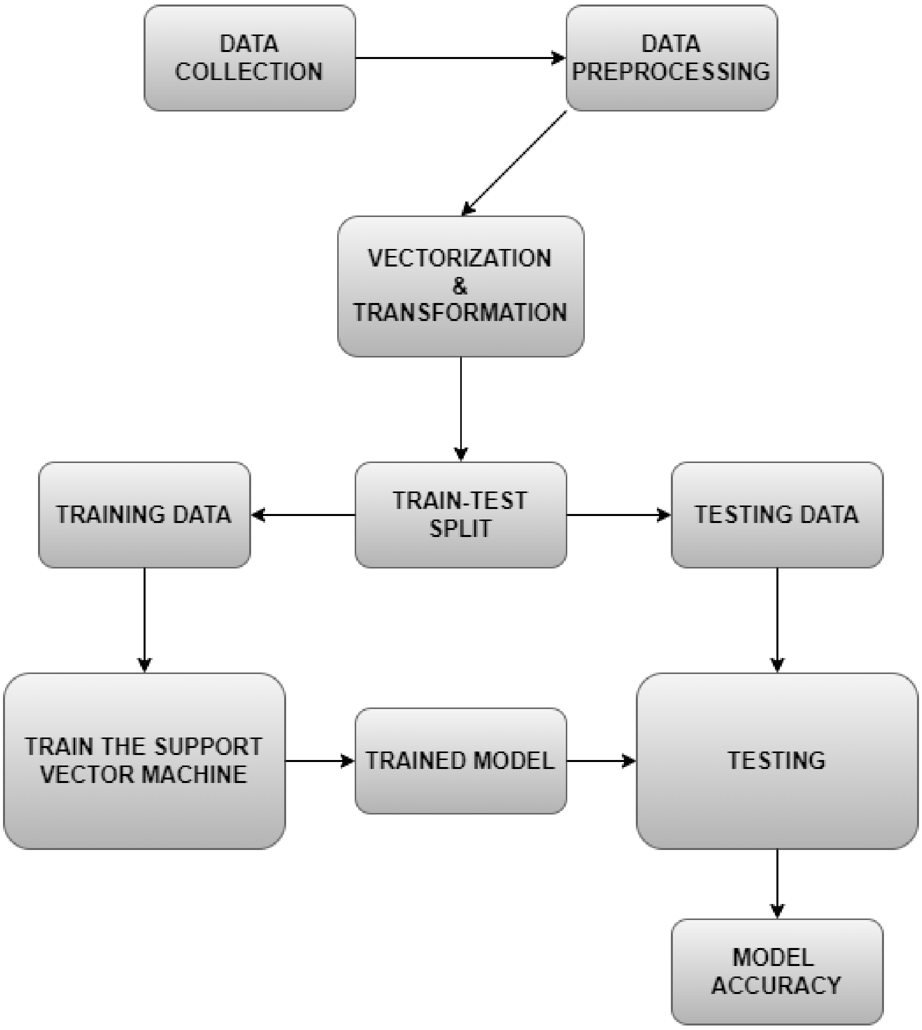
1.3.2 Software Used

- Python for scripting
- Libraries:
    - Pandas for data manipulation
    - NLTK for natural language processing tasks

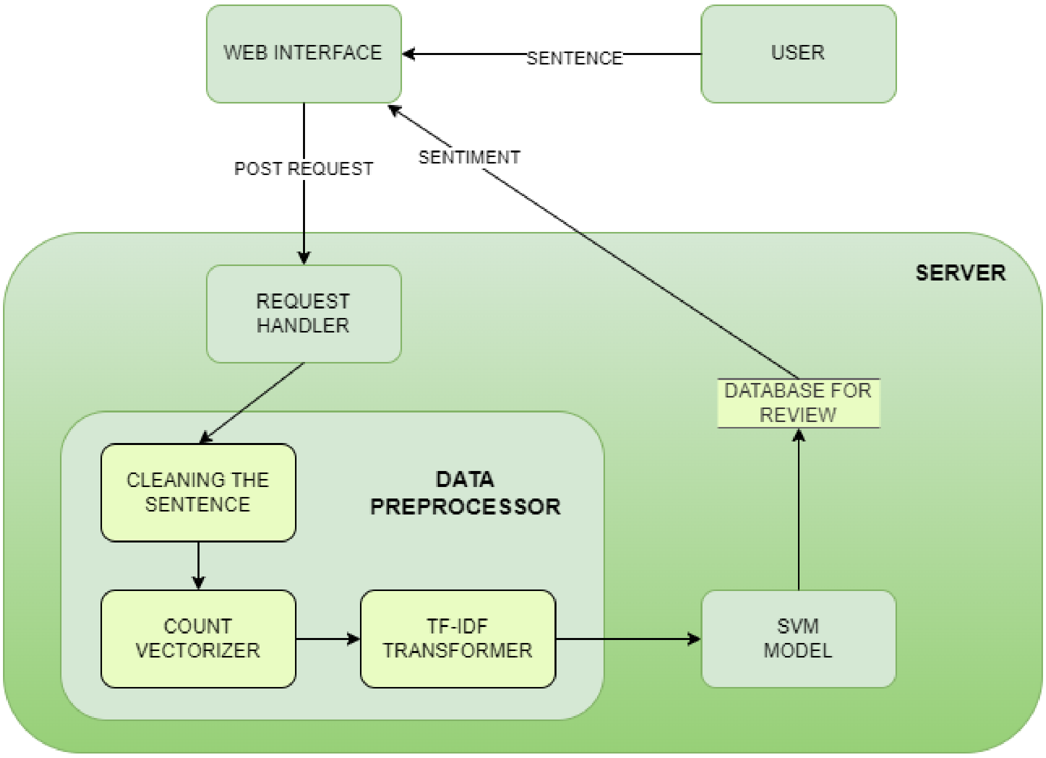
  - Scikit-learn for machine learning algorithms
  - Joblib for model serialization
  - SQLite: Database used to store predictions made by the sentiment analysis system.
- Node.js for server-side scripting
- Express.js for server framework
- HTML/CSS/JavaScript for web interface

2. Design Description

2.1 Flow Chart



2.2 Data Flow Diagram



3. Project Description

3.1 Database

The project utilizes SQLite for storing predictions made by the sentiment analysis system. SQLite is a lightweight, file-based database engine that is well-suited for small to medium-sized applications like this sentiment analysis project. SQLite offers simplicity, portability, and ease of integration, making it a suitable choice for managing prediction data.

3.2 File/Database Design

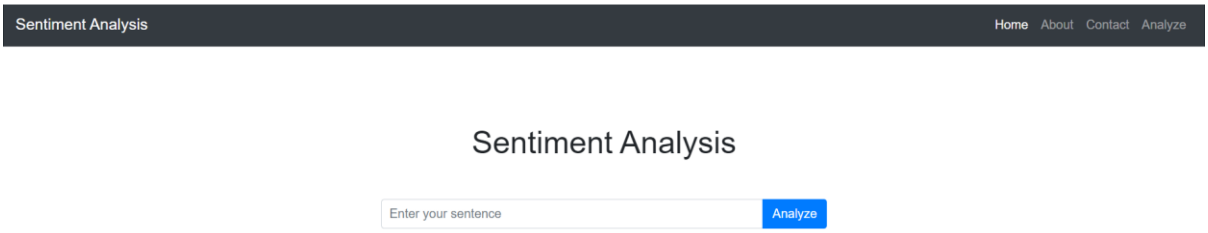
The database design consists of a single table named "predictions" with two columns:

- **input**: This column stores the input sentence or tweet that is analysed for sentiment.
- **prediction**: This column stores the predicted sentiment label (Positive, Neutral, or Negative) corresponding to the input sentence.

The table structure allows for storing each input along with its associated sentiment prediction. This design enables easy retrieval and analysis of predictions made by the sentiment analysis system. The database schema is simple yet effective for the requirements of the project, providing a straightforward way to store and access prediction data. We will leverage the collected data for model enhancement by conducting in-depth analysis of misclassified instances and identifying and addressing model weaknesses effectively.

4. Input/Output Form Design

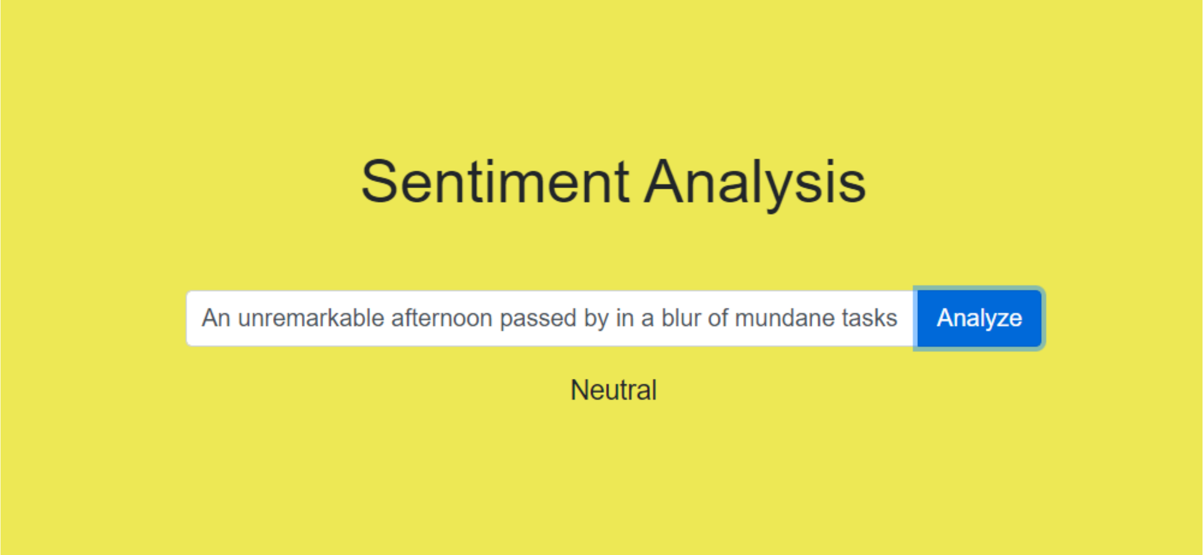
- *Input Form*



Sentiment Analysis

Enter your sentence

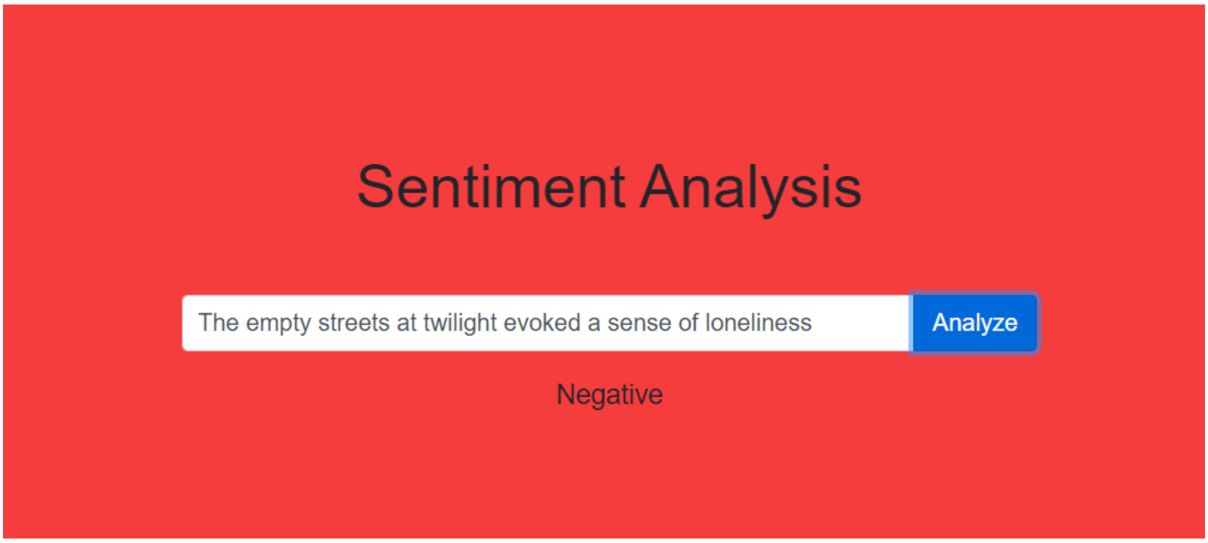
- *Output Forms*



Sentiment Analysis

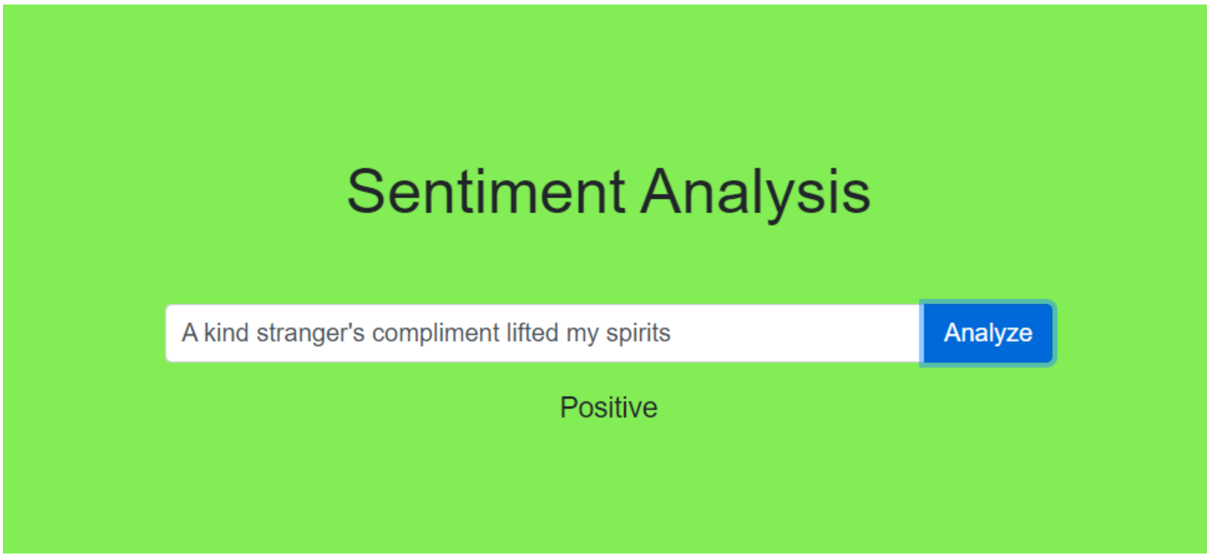
An unremarkable afternoon passed by in a blur of mundane tasks

Neutral


Sentiment Analysis

The empty streets at twilight evoked a sense of loneliness

Negative


Sentiment Analysis

A kind stranger's compliment lifted my spirits

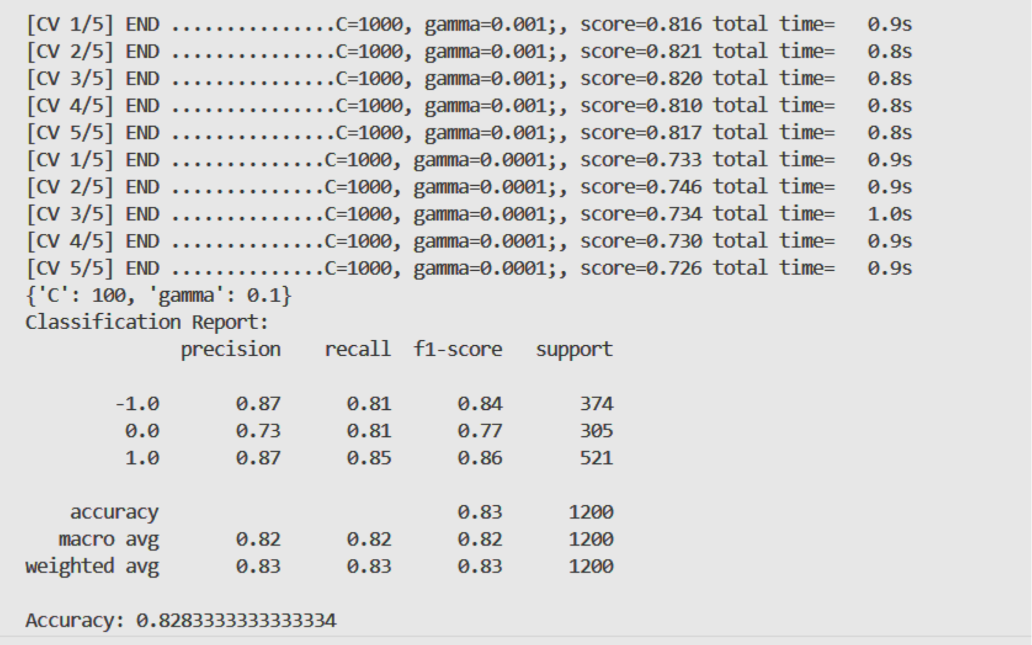
Positive

5. Testing & Tools

Grid Search CV:

To enhance the performance of our sentiment analysis model, we employed GridSearchCV, a technique for hyperparameter tuning, to optimize the parameters of the SVM classifier. Hyperparameters are crucial settings that influence the behavior and performance of the model.

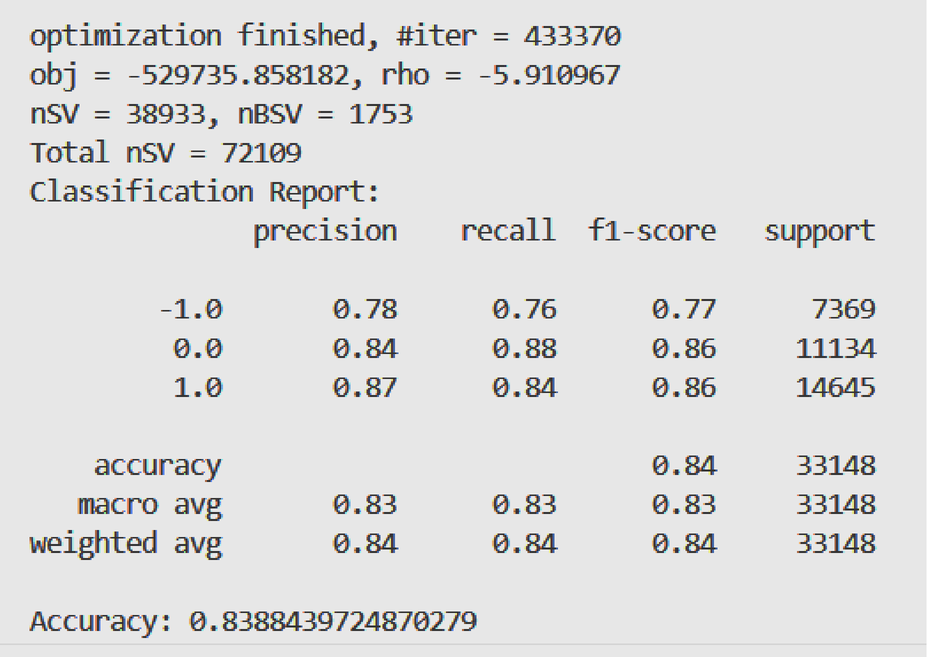
The best combination of hyperparameters (**best_params_**) determined by GridSearchCV were C=100, gamma=0.1



```
[CV 1/5] END ...............C=1000, gamma=0.001;, score=0.816 total time=   0.9s
[CV 2/5] END ...............C=1000, gamma=0.001;, score=0.821 total time=   0.8s
[CV 3/5] END ...............C=1000, gamma=0.001;, score=0.820 total time=   0.8s
[CV 4/5] END ...............C=1000, gamma=0.001;, score=0.810 total time=   0.8s
[CV 5/5] END ...............C=1000, gamma=0.001;, score=0.817 total time=   0.8s
[CV 1/5] END ..............C=1000, gamma=0.0001;, score=0.733 total time=   0.9s
[CV 2/5] END ..............C=1000, gamma=0.0001;, score=0.746 total time=   0.9s
[CV 3/5] END ..............C=1000, gamma=0.0001;, score=0.734 total time=   1.0s
[CV 4/5] END ..............C=1000, gamma=0.0001;, score=0.730 total time=   0.9s
[CV 5/5] END ..............C=1000, gamma=0.0001;, score=0.726 total time=   0.9s
{'C': 100, 'gamma': 0.1}
Classification Report:
              precision    recall  f1-score   support

        -1.0       0.87      0.81      0.84       374
         0.0       0.73      0.81      0.77       305
         1.0       0.87      0.85      0.86       521

    accuracy                           0.83      1200
   macro avg       0.82      0.82      0.82      1200
weighted avg       0.83      0.83      0.83      1200

Accuracy: 0.8283333333333334
```

Model Accuracy

The attained model accuracy of 84 percent signifies a commendable performance in sentiment analysis tasks. This level of accuracy indicates that our Support Vector Machine (SVM) classifier effectively learns and generalizes patterns from the provided textual data.



```
optimization finished, #iter = 433370
obj = -529735.858182, rho = -5.910967
nSV = 38933, nBSV = 1753
Total nSV = 72109
Classification Report:
              precision    recall  f1-score   support

        -1.0       0.78      0.76      0.77      7369
         0.0       0.84      0.88      0.86     11134
         1.0       0.87      0.84      0.86     14645

    accuracy                           0.84     33148
   macro avg       0.83      0.83      0.83     33148
weighted avg       0.84      0.84      0.84     33148

Accuracy: 0.8388439724870279
```

6. Future Scope

The project can be expanded in several ways, including:

- Incorporating more sophisticated NLP techniques for improved sentiment analysis.
- Experimenting with different machine learning algorithms and comparing their performance.
- Building a larger and more diverse dataset for training the model to enhance its accuracy.
- Implementing real-time sentiment analysis on streaming data sources such as social media feeds.

7. Conclusion

The project demonstrates the practical application of Support Vector Machines in sentiment analysis tasks. By preprocessing textual data, training an SVM classifier, and deploying it in a web-based environment, the project provides a functional system for sentiment analysis. However, there is room for further enhancement and refinement to improve accuracy and scalability.

8. Bibliography

- NLTK Documentation: https://www.nltk.org/
- Scikit-learn Documentation: https://scikit-learn.org/
- Theory: https://medium.com/scrapehero/sentiment-analysis-using-svm-338d418e3ff1
- Twitter Dataset: https://www.kaggle.com/datasets/saurabhshahane/twitter-sentiment-dataset
- Reviews Dataset: https://academictorrents.com/details/07e05fc1229555e124df72160a01b2540d04cebf