Twitter Sentiment Analysis Project Documentation

Introduction:

Sentiment analysis refers to the process of identifying and classifying the sentiments expressed in a text source. It has become increasingly significant in understanding public opinion across various domains. Social media platforms, especially Twitter, generate vast amounts of data daily. By analyzing tweets, we can extract valuable insights into public sentiment on a wide array of topics.

In the context of social media, sentiment analysis is crucial as platforms like Twitter are often used to track trends, opinions, and events in real-time. Companies, researchers, and policymakers can leverage this information to make data-driven decisions, monitor customer feedback, or even predict market behavior. Through sentiment analysis of tweets, we can observe shifts in public opinion almost instantly, providing a powerful tool for gauging societal trends.

Aim:

The aim of this project is to develop an Automated Machine Learning Sentiment Analysis Model to compute customer perception from Twitter data. The challenge lies in dealing with non-useful characters and noise present in the data, which can hinder the model's performance. The goal is to efficiently clean and preprocess this data to improve the accuracy and reliability of the sentiment analysis. This model can be applied to various use cases, such as understanding customer satisfaction, identifying market trends, and tracking public opinion on current events or products based on tweet sentiment.

Objective:

The primary objectives of this project are:

- To preprocess and clean the Twitter data by removing noise such as URLs, punctuations, stopwords, and non-alphanumeric characters.
- To build a machine learning model that accurately classifies the sentiment of tweets as either positive or negative.
- To evaluate the model's performance using appropriate metrics and fine-tune it to avoid overfitting.

Data Collection:

The dataset used for this project contains 1.6 million tweets with labeled sentiments. The data includes the following features:

- Target: Sentiment label (0 for negative, 1 for positive)
- Text: The tweet text
- Additional metadata such as Id, Date, User, etc., which were not used for analysis.

Data Preprocessing:

Data preprocessing is crucial for the success of any machine learning model. The following steps were implemented:

1. **Lowercasing**: All text data was converted to lowercase to maintain uniformity.
2. **Stopwords Removal**: Common stopwords were removed to focus on meaningful words.
3. **Punctuation and URL Removal**: All punctuations and URLs were stripped from the text to reduce noise.
4. **Tokenization**: The text was split into individual tokens (words) for analysis.
5. **Lemmatization**: Words were lemmatized to their base forms to ensure consistency.
6. **Handling Imbalanced Data**: The target distribution was analyzed to ensure balanced classes for training.

Feature Engineering:

The cleaned text data was transformed into numerical features using the TF-IDF (Term Frequency-Inverse Document Frequency) vectorizer, which converts textual data into a matrix of token counts while taking into account the importance of words in the entire corpus. The vectorizer was configured to consider both unigrams and bigrams, with a maximum of 10,000 features.

Modeling:

Several machine learning models were trained and evaluated, including:

- **Logistic Regression**: A basic model with L2 regularization to avoid overfitting.
- **Support Vector Classifier (SVC)**: Known for its robustness in high-dimensional spaces.

- **Bernoulli Naive Bayes (BNB)**: A simple probabilistic model suitable for binary/boolean features.

Cross-Validation:

To ensure the robustness of the models, 5-fold cross-validation was employed. This technique split the training data into 5 subsets, training the model on 4 subsets while validating it on the remaining one, and repeating this process 5 times. The cross-validation scores provided a reliable measure of model performance by averaging results over different splits, reducing the chance of overfitting on a single train-test split.

Evaluation:

The models were evaluated on the test set using key performance metrics, including:

- Accuracy: The percentage of correct predictions out of all predictions.
- Precision & Recall: To measure how well the model handles positive and negative classes.
- F1-Score: The harmonic mean of precision and recall to balance both metrics.
- ROC-AUC: To assess the model's performance across different classification thresholds.

Additionally, a confusion matrix was plotted to visualize the true positives, false positives, true negatives, and false negatives. The ROC-AUC curve was also plotted to provide a more comprehensive view of the model's discriminative ability.

Results:

- The Logistic Regression model with L2 regularization provided a good balance between bias and variance.
- The SVC model also performed well, but required tuning to avoid overfitting.
- The Bernoulli Naive Bayes model was effective but slightly less accurate compared to the others.

Model	Training Accuracy	Testing Accuracy	Precision	Recall	F1-Score	Accuracy (Overall)
Logistic Regression	77.97%	77.51%	0.79 (0), 0.76 (1)	0.75 (0), 0.80 (1)	0.77 (0), 0.78 (1)	78.0%
Support Vector Classifier	77.90%	77.46%	0.79 (0), 0.76 (1)	0.75 (0), 0.80 (1)	0.77 (0), 0.78 (1)	77.5%
Bernoulli Naive Bayes	76.06%	75.88%	0.78 (0), 0.74 (1)	0.73 (0), 0.79 (1)	0.75 (0), 0.77 (1)	76.0%

Conclusion:

This project successfully developed and implemented multiple sentiment analysis models for classifying Twitter data into positive and negative sentiments. After comparing several models, Logistic Regression was found to be the most balanced, providing good performance while maintaining simplicity and generalizing well across different datasets.

The insights gained from this sentiment analysis could be useful in several domains, including customer feedback analysis, market research, and opinion mining.