NLP Mini Project

Sentiment Analysis of Customer Reviews

Problem Statement:

The restaurant aims to enhance customer satisfaction by addressing negative feedback promptly. To achieve this, a binary review classification model is needed to categorize customer reviews as either positive or negative.

Dataset Description:

The dataset comprises customer reviews for the restaurant, along with corresponding positive/negative labels. Additionally, there are fresh customer reviews that require labeling. This dataset will be used to train the sentiment analysis model.

Approach and Implementation

1. Data Loading and Preprocessing

Data Loading

The dataset used for this project was sourced from Kaggle, a popular platform for machine learning datasets. The load\_dataset function reads the data from a CSV file downloaded from Kaggle, ensuring that the dataset is easily accessible and ready for analysis.

Kaggle Dataset Source

The dataset, obtained from Kaggle, includes customer reviews for the restaurant, paired with sentiment labels. Utilizing datasets from platforms like Kaggle provides a diverse and real-world set of reviews, enhancing the model's ability to generalize to various customer sentiments.

Data Preprocessing

In the preprocess\_data function, the 'Review' column undergoes preprocessing steps to make the text data suitable for analysis. The function checks for NaN values, replacing them with an empty string, and then converts the text to lowercase while removing non-alphanumeric characters. These steps streamline the text data for downstream tasks.

2. Data Representation

Bag-of-Words (BoW) Representation

To convert the preprocessed text data into a numerical format, a bag-of-words (BoW) representation is created using scikit-learn's CountVectorizer. Scikit-learn is a powerful machine learning library that provides tools for data preprocessing, modeling, and evaluation.

scikit-learn Overview

Scikit-learn is an open-source machine learning library in Python that simplifies the implementation of various machine learning algorithms. It offers a wide range of functionality, including tools for data preprocessing, feature extraction, model training, and evaluation. In this project, scikit-learn's CountVectorizer is used to transform text data into a format suitable for machine learning models.

3. Model Building - Naive Bayes Classifier

Choice of Naive Bayes Classifier

A Naive Bayes classifier from scikit-learn, specifically the MultinomialNB implementation, is chosen for its simplicity and effectiveness in text classification tasks.

scikit-learn's Naive Bayes Implementation

Scikit-learn provides an easy-to-use implementation of the Naive Bayes algorithm, making it accessible for both beginners and experienced practitioners. The MultinomialNB classifier is suitable for discrete data like the bag-of-words representation used in this project.

Training the Classifier:

The build\_classifier function utilizes scikit-learn's implementation to initialize and train the Naive Bayes classifier on the preprocessed and BoW-represented training data (X\_train and y\_train).

4. Prediction on Unseen Reviews:

Generating Predictions:

The trained Naive Bayes classifier is applied to the test set (X\_test) using the predict\_reviews function. This step simulates the model's performance on unseen reviews, and the resulting predictions are used for evaluating the model's accuracy and generating a detailed classification report.

5. Model Evaluation:

Accuracy and Classification Report:

The model's performance is evaluated using accuracy, providing an overall measure of correct predictions. Additionally, the classification report offers a more detailed assessment, presenting precision, recall, and F1-score for each sentiment class. This comprehensive evaluation guides further analysis and improvements.

Results and Discussion:

Model Performance:

Accuracy:

The developed sentiment analysis model achieved an accuracy of 61.8% on the test set. This accuracy metric provides an overall indication of the model's correctness in predicting sentiment labels.

Classification Report Insights;

Positive Sentiments:

The model excelled in identifying positive sentiments, achieving a precision of 0.74 and a recall of 0.79. This indicates that the model correctly identified the majority of positive reviews and had a high precision in classifying reviews as positive.

Challenges in Certain Classes:

The model faced challenges in certain sentiment classes, such as '2' and '2.5.' These classes showed lower precision and recall scores, suggesting that the model struggled to accurately classify reviews with these specific sentiment labels. Further investigation into the characteristics of these classes and potential data imbalances could provide insights for improvement.

Recommendations:

Explore Data Anomalies:

To enhance model performance, it is recommended to explore and analyze misclassified instances, especially in classes with lower precision and recall. Identifying patterns or anomalies in the data that contribute to misclassifications can guide improvements in preprocessing or feature engineering.

Feature Engineering:

Consider additional features or alternative representations of text data that might capture more nuanced sentiment information. Experimenting with advanced natural language processing techniques, such as word embeddings or deep learning approaches, could further improve the model's ability to capture complex sentiment patterns.

Hyperparameter Tuning:

Fine-tuning the hyperparameters of the Naive Bayes classifier and exploring alternative machine learning models may provide opportunities for performance enhancement. Adjusting parameters such as smoothing parameters in the Naive Bayes algorithm could lead to better generalization.

Conclusion:

In conclusion, the implemented sentiment analysis model provides a valuable foundation for categorizing customer reviews for the restaurant. While achieving an overall accuracy of 61.8%, the model's strengths in identifying positive sentiments are evident. However, there are opportunities for improvement, especially in handling specific sentiment classes.The insights gained from the classification report guide future iterations of the model. By addressing challenges in misclassification and exploring advanced techniques, the restaurant can refine the model to better serve its customer satisfaction goals.

References:

Kaggle Dataset Source:

<https://www.kaggle.com/datasets/joebeachcapital/restaurant-reviews?resource=download>

scikit-learn Documentation:

<https://scikit-learn.org/0.21/documentation.html>

General References:

"Introduction to Machine Learning with Python: A Guide for Data Scientists" by Andreas C. Müller & Sarah Guido.

"Natural Language Processing in Action" by Lane, Howard, & Hapke.