Activity based

Project Report on

Artificial Intelligence

Project Phase - I

Submitted to Vishwakarma University, Pune

Under the Initiative of

Contemporary Curriculum, Pedagogy, and Practice (C2P2)

By

Imaad Imran Hajwane

SRN No: 202101132

Roll No: 23

Div: A

Third Year Engineering

Faculty In charge: - Prof. Tarapore Sir

Date Of Project Phase 1: - 16th February 2024

Department of Computer Engineering

Faculty of Science and Technology

Academic Year

2023-2024 Term-II

AI: Phase I

Project Name: Product Review Sentiment Analysis for Restaurant Review with Food Images

Introduction:

In the modern digital landscape, where consumer opinions shared online significantly influence business decisions, the restaurant industry faces a unique challenge in analysing sentiment due to the prevalence of both textual reviews and accompanying food images. This project aims to bridge this gap by developing an advanced sentiment analysis system that integrates both textual and visual data.

Traditional sentiment analysis techniques primarily focus on text, yet in the realm of restaurant reviews, images play a pivotal role in shaping perceptions. By combining textual reviews with food images, this study seeks to provide a more comprehensive understanding of customer sentiment towards restaurants.

Problem Statement

1. Design the Sentiment Analysis system tailored for product reviews, with a focus on classifying reviews as positive, negative, or neutral. Consider incorporating aspect-based sentiment analysis to identify sentiments related to specific product features. Define the data processing pipeline, feature extraction methods, and sentiment classification model. Plan the integration of aspect-based analysis to provide a more nuanced understanding of user sentiments towards specific aspects of the product.

Objective

To develop a hybrid sentiment analysis system for restaurant reviews that integrates textual data, aiming to provide a comprehensive understanding of customer sentiment and enhance decision-making processes for restaurant owners and stakeholders.

Processing strategy:

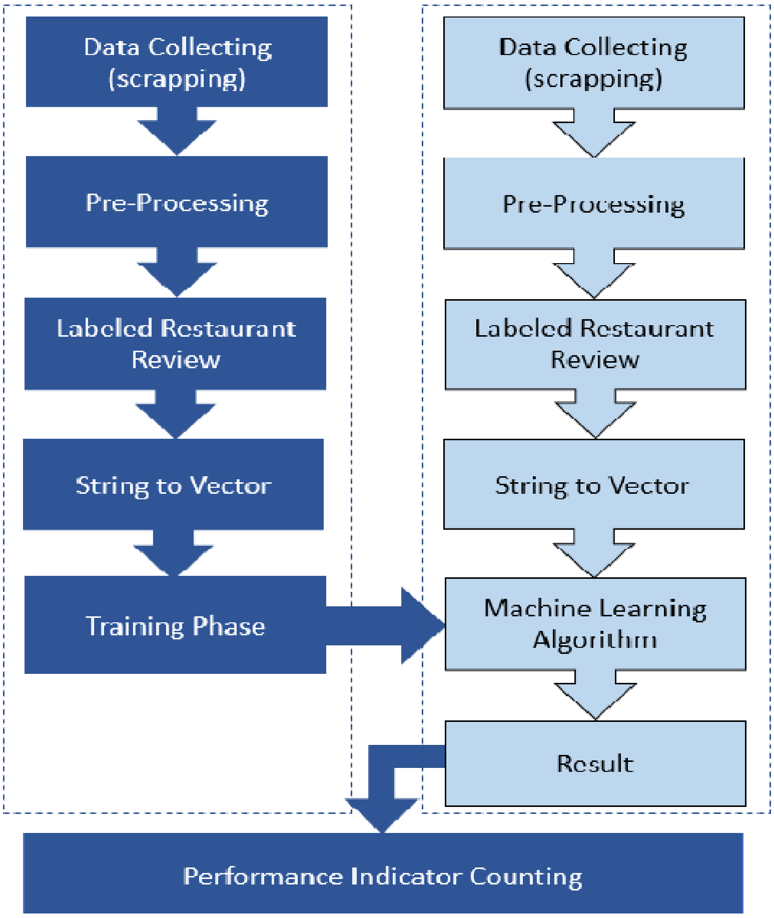
Data Collecting (scrapping)
Pre-Processing
Labeled Restaurant Review
String to Vector
Training Phase
Data Collecting (scrapping)
Pre-Processing
Labeled Restaurant Review
String to Vector
Machine Learning Algorithm
Result
Performance Indicator Counting

Figure 1: Flow of the Processing Ideology.

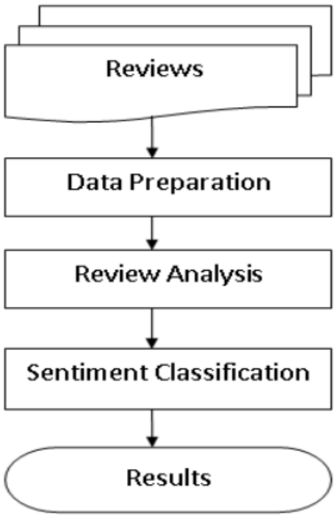

Figure 2: Flow chart for understanding the analysis methods

Class	Reviews
Positive	The taste of the food was very good, had a very good experience
Negative	The taste of the food was not good, it was not a good experience.

Table 1: Classification Predicting Information.

Domain Knowledge:

1. Natural Language Processing (NLP):
    - NLP is a branch of artificial intelligence (AI) focused on enabling computers to understand, interpret, and generate human language in a way that is both meaningful and useful.
    - It involves techniques for analysing, processing, and generating natural language data, allowing computers to interact with human language in various forms, such as text, speech, and images.
    - NLP applications include sentiment analysis, machine translation, speech recognition, text summarization, and question answering systems.

2. <u>NLTK (Natural Language Toolkit):</u>
    - NLTK is a leading platform for building Python programs to work with human language data.
    - It provides easy-to-use interfaces and libraries for tasks such as tokenization, stemming, part-of-speech tagging, parsing, and semantic reasoning.
    - NLTK offers a comprehensive suite of text processing libraries and tools, making it a popular choice for NLP research, education, and development.
    - It is widely used in academia and industry for tasks ranging from simple text preprocessing to complex NLP applications.

Diagrammatic representation:

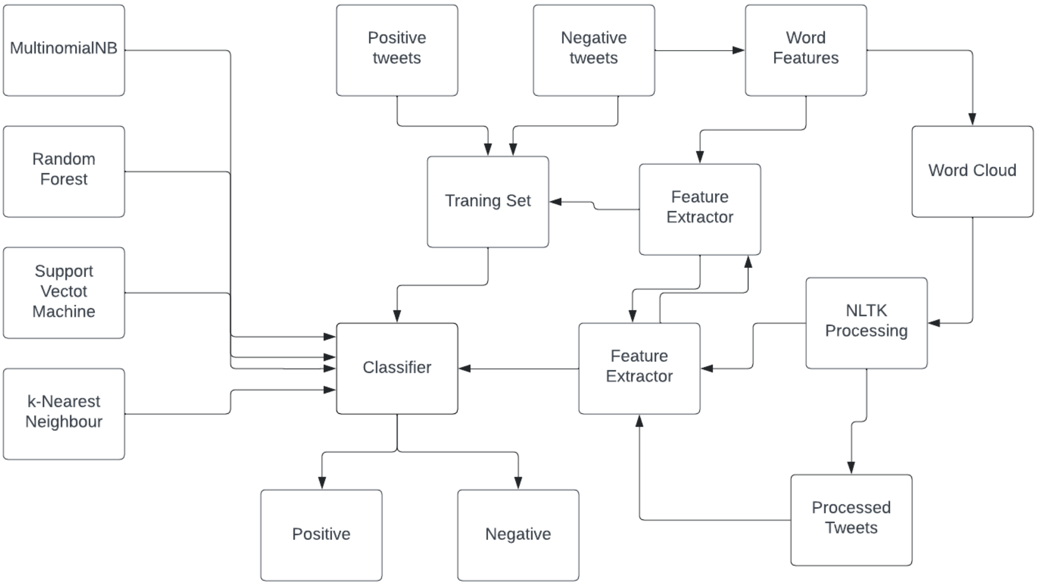

Figure 03: Block Diagram for the Sentiment Analysis

Algorithm and supporting methods:

Introduction to Classification Algorithms:

Classification algorithms are essential tools in machine learning, assigning predefined categories to input data based on their features. In this overview, we'll explore four common algorithms: Multinomial Naive Bayes, Random Forest, Support Vector Machine (SVM), and k-Nearest Neighbours (KNN). Understanding these algorithms is crucial for various applications, including text classification and predictive modelling. Let's dive into each to understand their principles and applications.

The four main algorithms are:

1. Multinomial Naive Bayes
2. Random Forest
3. Support Vector Machine (SVM)
4. k-Nearest Neighbors (KNN)

Descriptions:

1. Multinomial Naive Bayes

- Multinomial Naive Bayes is a probabilistic classification algorithm based on Bayes' theorem.
- It's commonly used for text classification tasks, such as spam detection or sentiment analysis.
- It assumes that features are independent and distributed according to a multinomial distribution.

Training:

1. Given a set of training examples with features xi and corresponding class labels yi.
2. Count the occurrences of each feature xi for each class y.
3. Calculate the probability $P(xi \mid y)$, representing the likelihood of feature xi occurring given class y, using Laplace smoothing to handle unseen features.

Prediction:

1. Given a new example with features $xnew$.
2. Calculate the posterior probability $P(y \mid xnew)$ for each class y using Bayes' theorem.
3. The class with the highest posterior probability is assigned to the new example.

2. Random Forest

- Random Forest is an ensemble learning algorithm that constructs multiple decision trees during training.
- Each tree is trained on a random subset of the training data and a random subset of features.
- It makes predictions by averaging or voting across the predictions of individual trees, which often leads to robust and accurate results.

Training:

1. Given a set of training data with features and corresponding class labels.
2. Randomly select a subset of the training data (bootstrap sample) and a subset of features for each decision tree.
3. Grow multiple decision trees independently using the selected subsets.
4. At each node of the tree, randomly select a subset of features to split on, optimizing for the best split based on a criterion such as Gini impurity or information gain.
5. Continue growing the trees until a stopping criterion is met, such as reaching a maximum depth or having minimum samples per leaf.

Prediction:

1. Given a new example with features.
2. Let each decision tree in the forest make a prediction based on the input features.
3. Aggregate the predictions of all trees through averaging (for regression) or voting (for classification).
4. The final prediction is the average or majority vote of all individual tree predictions.

3. Support Vector Machine (SVM)

- Support Vector Machine is a supervised learning algorithm used for classification and regression tasks.
- It works by finding the hyperplane that best separates different classes in the feature space.
- SVM aims to maximize the margin between classes, making it effective even in high-dimensional spaces and with non-linear data through the use of kernel functions.

Training:

1. Given a set of labeled training data, where each data point belongs to one of two classes.
2. SVM aims to find the hyperplane that best separates the data points of different classes while maximizing the margin between the classes.
3. The hyperplane is defined by a set of weights and biases that minimize the classification error and maximize the margin.
4. In cases where the data is not linearly separable, SVM uses kernel functions to map the input data into a higher-dimensional space where it becomes linearly separable.

Prediction:

1. Given a new data point, SVM predicts its class by determining which side of the hyperplane it falls on.
2. If the data point lies on the positive side of the hyperplane, it's classified as belonging to one class; if it lies on the negative side, it's classified as belonging to the other class.

4. k-Nearest Neighbors (KNN)

- k-Nearest Neighbors is a simple and intuitive classification algorithm.
- It classifies data points based on the majority class of their k nearest neighbors in the feature space.
- KNN is non-parametric and lazy, meaning it doesn't make explicit assumptions about the underlying data distribution and doesn't require training before making predictions.

Training:

1. Given a set of labeled training data, where each data point has a feature vector and corresponding class label.
2. KNN stores all training data points in memory.

Prediction:

1. Given a new data point to classify:
2. Calculate the distances between the new data point and all training data points using a distance metric like Euclidean distance.
3. Select the k-nearest neighbors to the new data point based on the calculated distances.
4. Assign the class label to the new data point based on the majority class among its k-nearest neighbors.
    - For classification, the most common class among the neighbors is assigned.
    - For regression, the average or median value of the neighbors' labels is assigned.