Sentiment Analysis Using E-Commerce Review with a Hybrid Machine Learning-Based Model

Submitted by
125018018
Dineish.V.S
Btech Computer Science and Business systems
Sastra Deemed University
Thanjavur

Submitted to
Swetha Varadarajan

Table of Contents

1.Introduction:

Customer reviews are essential for businesses to gauge public sentiment toward their products and services. This project aims to develop a sentiment analysis system using VADER (Valence Aware Dictionary for Sentiment Reasoning) to classify reviews as Positive, Negative, or Neutral. By combining VADER's sentiment scores with custom preprocessing and machine learning classifiers, we seek to enhance accuracy, particularly in identifying neutral sentiments, which often present classification challenges. Our solution will enable businesses to analyze customer feedback effectively and make informed strategic decisions.

2. Abstract

In this project, we aimed to build a sentiment analysis system using VADER (Valence Aware Dictionary for Sentiment Reasoning) to classify product reviews as Positive, Negative, or Neutral. Sentiment analysis plays a crucial role in understanding customer feedback by automatically identifying the sentiment conveyed in textual data. Our approach involves applying VADER sentiment scores to reviews, focusing on the 'compound' score to classify sentiment. We also implemented a custom sentiment classification algorithm based on threshold adjustments to ensure proper handling of Neutral reviews. A custom pipeline using VADER scores, data preprocessing, and machine learning classifiers was tested for better accuracy. Results demonstrated improved sentiment prediction, with specific challenges around Neutral sentiment detection.

3.Project Objectives: The objective of this project is to develop a sentiment analysis system that can automatically classify customer reviews into three sentiment categories: Positive, Negative, or Neutral. The system is intended to help businesses analyze customer opinions at scale, improving their understanding of customer satisfaction and dissatisfaction, which can inform strategic decision-making.

4.Problem Formulation: Customer reviews are an invaluable source of feedback for companies. However, manually analyzing thousands of reviews to understand customer sentiment is labor-intensive and error-prone. Sentiment analysis, which is the use of natural language processing (NLP) and machine learning (ML) to determine the sentiment behind textual data, addresses this challenge. Existing models like VADER provide strong performance, especially in dealing with social media texts and short, informal writing. However, accurately classifying Neutral reviews remains a challenge, as many models tend to skew towards Positive classification due to slight positivity in language structure. Therefore, this project focuses on overcoming this bias and providing more accurate sentiment classification.

Proposed Solution: We propose using the VADER sentiment analysis tool combined with custom pre-processing and classification techniques to achieve more accurate sentiment classification, especially for Neutral reviews. This will be augmented with machine learning models trained on the same dataset to improve accuracy over a wide variety of reviews.

5.Dataset Overview:

The dataset used for this project consists of amazon customer product reviews. These reviews were cleaned and processed to remove noise, stop words, and irrelevant characters, ensuring that the analysis focuses on the most important parts of the text. Each review includes a column for the actual text of the review, which is processed and passed through the sentiment analysis pipeline.

1) id object
2) asins object
3) brand object
4) categories object
5) colors object
6) dateAdded object
7) dateUpdated object
8) dimension object
9) ean float64
10) keys object
11) manufacturer object
12) manufacturerNumber object
13) name object
14) prices object
15) reviews.date object
16) reviews.doRecommend object
17) reviews.numHelpful float64
18) reviews.rating float64
19) reviews.sourceURLs object
20) reviews.text object
21) reviews.title object
22) reviews.userCity float64
23) reviews.userProvince float64
24) reviews.username object
25) sizes float64
26) upc float64
27) weight object

6.Preprocessing Steps

The text preprocessing pipeline follows these key steps:

1. **Text Cleaning:** Reviews are stripped of special characters, HTML tags, and unnecessary symbols.
2. **Tokenization:** Each review is broken down into words, which are then analyzed for sentiment.
3. **Stopwords Removal:** Common stopwords (e.g., "the," "is") that do not contribute to the meaning of the text are removed.

4. **Stemming:** Words are reduced to their base form (e.g., "running" becomes "run") to focus on root words during analysis.

7.VADER Sentiment Analysis

VADER, a rule-based sentiment analysis tool specifically designed to detect sentiments in short text snippets, was applied to each review. VADER generates four sentiment scores for each review:

- **Negative**: The proportion of the text that conveys negative sentiment.
- **Neutral**: The proportion of the text that conveys neutral sentiment.
- **Positive**: The proportion of the text that conveys positive sentiment.
- **Compound**: A normalized score that summarizes the overall sentiment (ranging from -1 to 1).

The `compound` score, which is a combination of all three categories, was used to classify the sentiment of the review. Based on threshold values, reviews were classified as Positive, Negative, or Neutral:

- Reviews with `compound` scores above 0.05 were considered Positive.
- Reviews with `compound` scores below -0.05 were considered Negative.
- Reviews with `compound` scores between -0.05 and 0.05 were considered Neutral.

8.Machine Learning Models

In addition to VADER, we experimented with machine learning models trained on a labeled dataset of reviews. The models used include:

```
1. GaussianNB
2. DecisionTreeClassifier
3. RandomForestClassifier
4. LogisticRegression
5. XGBClassifier
6. KNeighborsClassifier
```

For each model, we used the Term Frequency-Inverse Document Frequency (TF-IDF) vectorizer to convert the cleaned text data into numerical features that could be fed into the classifiers.

9.Model Performance Evaluation:

The dataset was split into training and testing sets. The model training pipeline involved:

- **Data Splitting**: 80% of the data was used for training, and 20% for testing.

- **Model Evaluation**: Accuracy, precision, recall, and F1-score were used as evaluation metrics. The models were evaluated for their ability to correctly classify reviews into Positive, Negative, or Neutral sentiment categories.
- **Hyperparameter Tuning**: Grid search was used to find the optimal hyperparameters for the machine learning models.

Evaluation metrics:

1. **Precision**: Measures the accuracy of positive predictions (true positives / (true positives + false positives)).
2. **Recall**: Measures the ability to find all relevant instances (true positives / (true positives + false negatives)).
3. **Support**: The number of actual occurrences of the class in the specified dataset.
4. **F1 Score**: The harmonic mean of precision and recall, balancing both metrics (2 * (precision * recall) / (precision + recall)).

10.Comparison of Machine Learning Models:

Model	Type	Advantages	Disadvantages	Use Cases
Gaussian Naive Bayes	Classification	Fast, effective with small datasets	Assumes feature independence	Text classification, spam detection
Logistic Regression	Classification	Interpretable, good with linear data	Assumes linearity, sensitive to outliers	Binary classification, marketing
Random Forest	Classification/Regression	Reduces overfitting, handles non-linearity	Less interpretable	Feature selection, complex datasets
K-Nearest Neighbors (KNN)	Classification/Regression	Simple, adaptable to new data	Expensive for large datasets	Recommendation systems
Decision Tree	Classification/Regression	Easy to interpret, han ↓ non-linearity	Prone to overfitting	Customer segmentation

11.Results and Insights:

Upon applying the VADER sentiment analysis tool to the product reviews dataset, we obtained the following key insights:

```
Model 1: GaussianNB
Training Accuracy: 0.9671
Testing Accuracy: 0.8844
-----------------
Model 2: DecisionTreeClassifier
Training Accuracy: 1.0000
Testing Accuracy: 0.8844
-----------------
Model 3: RandomForestClassifier
Training Accuracy: 1.0000
Testing Accuracy: 0.9187
-----------------
Model 4: LogisticRegression
Training Accuracy: 0.9178
Testing Accuracy: 0.9062
-----------------
Model 5: XGBClassifier
Training Accuracy: 0.8927
Testing Accuracy: 0.8781
-----------------
Model 6: KNeighborsClassifier
Training Accuracy: 0.9076
Testing Accuracy: 0.8938
-----------------
```

12.Error Analysis

Despite the improvements, there were some cases where even the best-performing models struggled:

- **Sarcasm Detection**: Reviews with sarcastic tones, such as "Oh great, another product that doesn't work," were often misclassified as Positive due to the wording.
- **Ambiguous Reviews**: Some Neutral reviews with mixed sentiments ("It's neither bad nor good, but could be better") still posed a challenge, particularly for simpler models like Logistic Regression.
- **Short Reviews**: Shorter reviews like "Good" or "Meh" were difficult to classify as the models lacked context, leading to inconsistencies

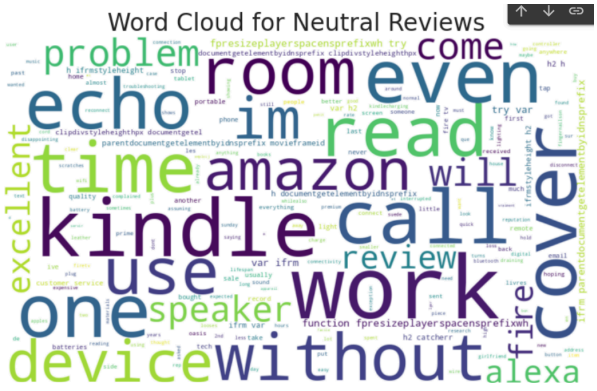
Code Link: ML_Project.ipynb

13.Code Implementation:

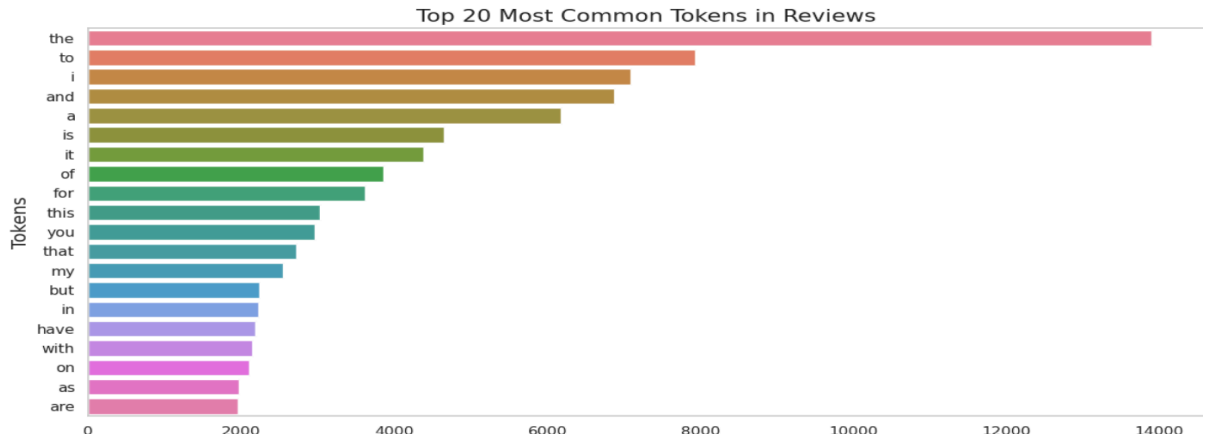
Tokenization:

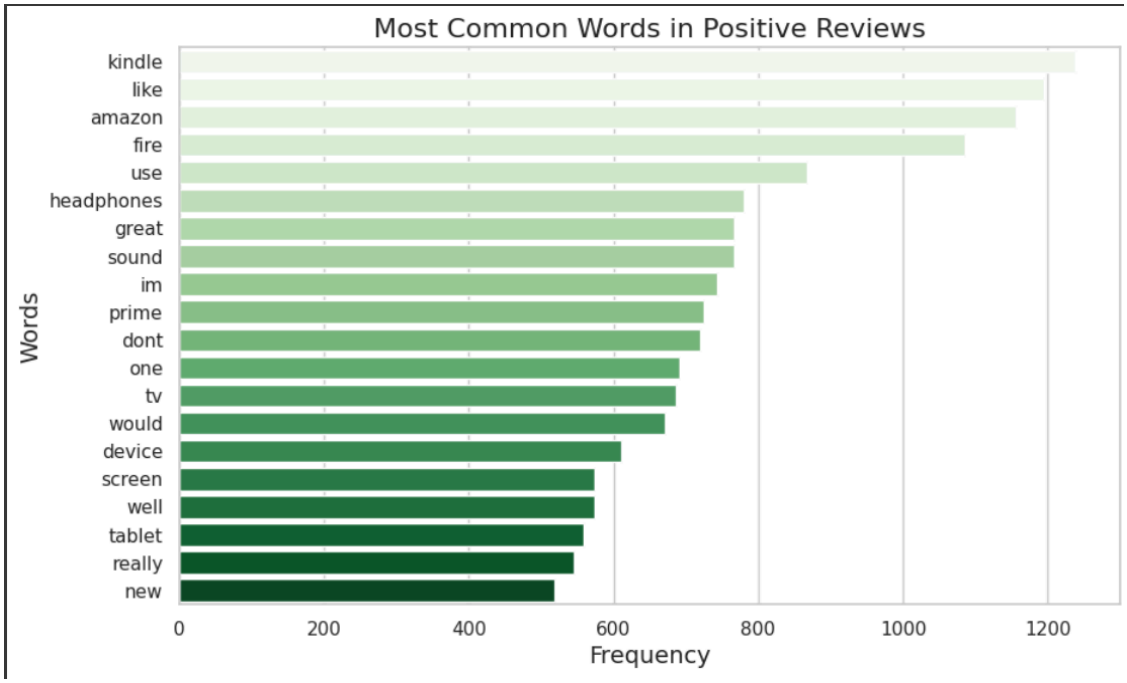
	reviews.text	Cleaned_Review	Tokenized_Review	Stemmed_Review	No_Stopwords_Review	compound	sentiment
0	I initially had trouble deciding between the p...	i initially had trouble deciding between the p...	[i, initially, had, trouble, deciding, between...	[i, initi, had, troubl, decid, between, the, p...	[initially, trouble, deciding, paperwhite, voy...	0.9879	Positive
1	Allow me to preface this with a little history...	allow me to preface this with a little history...	[allow, me, to, preface, this, with, a, little...	[allow, me, to, prefac, thi, with, a, littl, h...	[allow, preface, little, history, casual, read...	0.9881	Positive
2	I am enjoying it so far. Great for reading. Ha...	i am enjoying it so far great for reading had ...	[i, am, enjoying, it, so, far, great, for, rea...	[i, am, enjoy, it, so, far, great, for, read, ...	[enjoying, far, great, reading, original, fire...	0.4364	Positive
3	I bought one of the first Paperwhites and have...	i bought one of the first paperwhites and have...	[i, bought, one, of, the, first, paperwhites, ...	[i, bought, one, of, the, first, paperwhit, an...	[bought, one, first, paperwhites, pleased, con...	0.9746	Positive
4	I have to say upfront - I don't like coroporat...	i have to say upfront i dont like coroporate ...	[i, have, to, say, upfront, i, dont, like, cor...	[i, have, to, say, upfront, i, dont, like, cor...	[say, upfront, dont, like, coroporate, hermeti...	0.9980	Positive

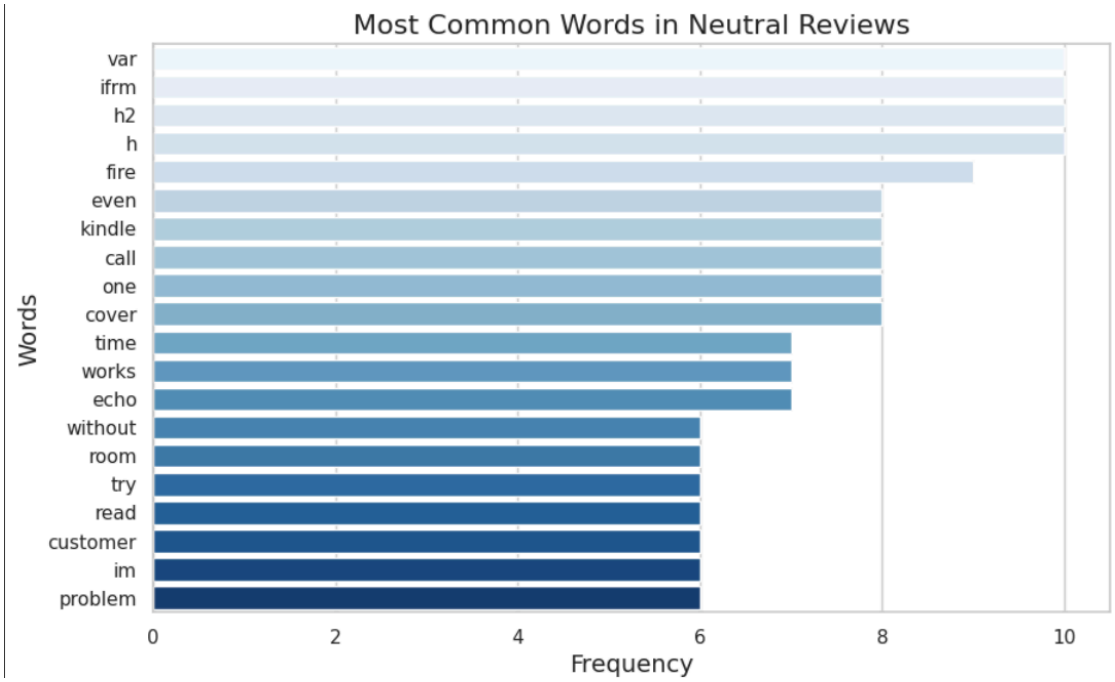
Word cloud:

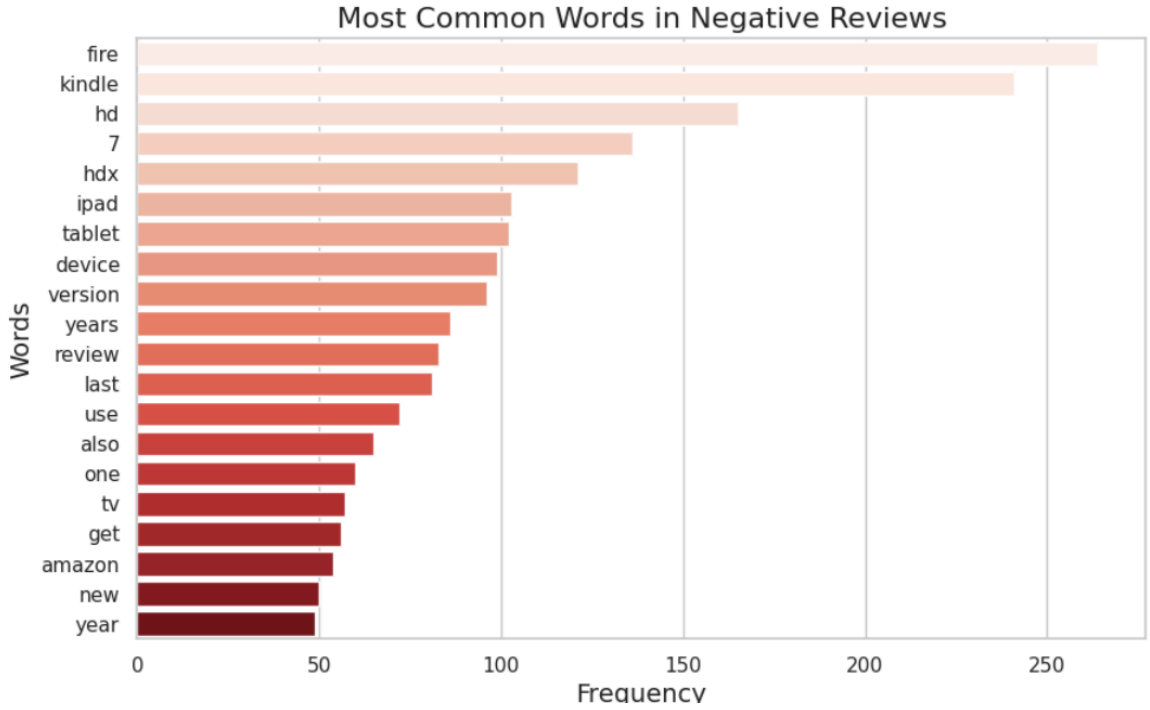


Plotting common words:

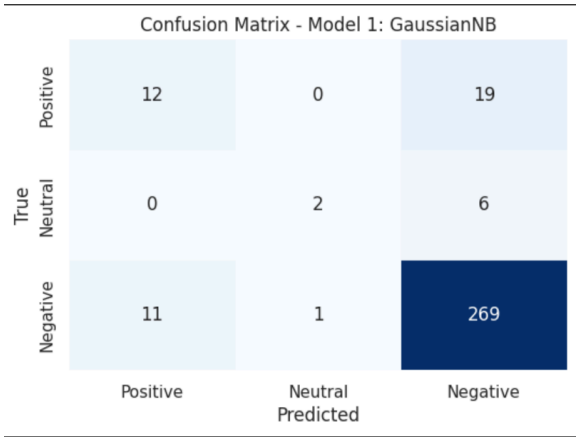


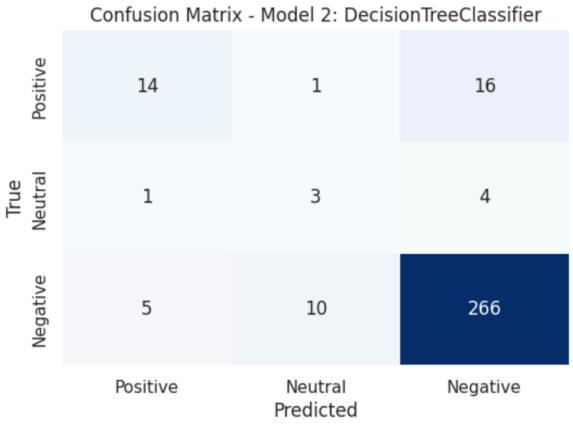


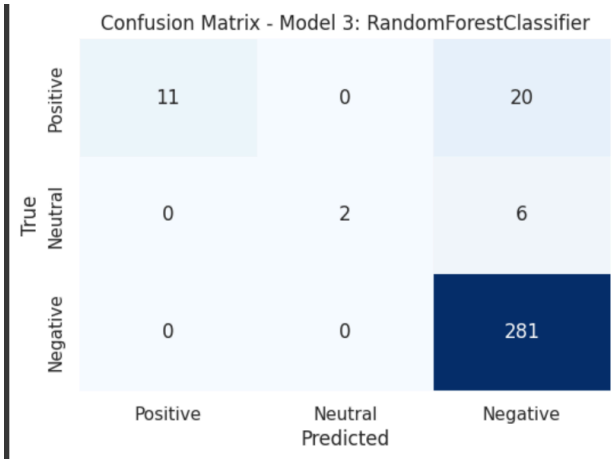

Most Common Words in Neutral Reviews
Words
var
ifrm
h2
h
fire
even
kindle
call
one
cover
time
works
echo
without
room
try
read
customer
im
problem
0
2
4
6
8
10
Frequency

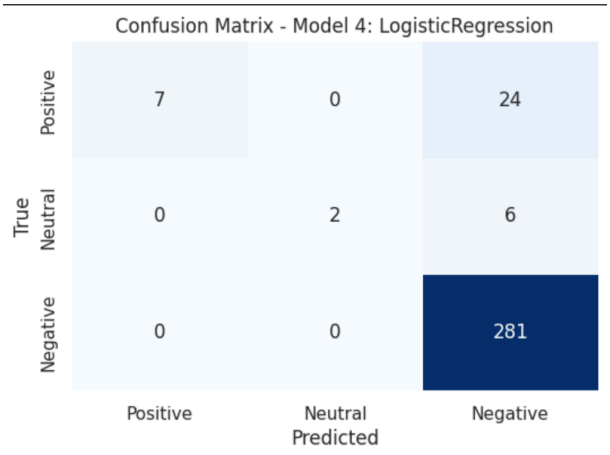

Most Common Words in Negative Reviews
Words
fire
kindle
hd
7
hdx
ipad
tablet
device
version
years
review
last
use
also
one
tv
get
amazon
new
year
0
50
100
150
200
250
Frequency

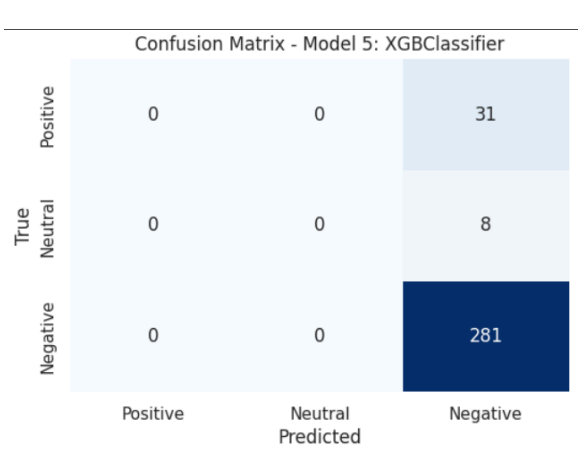
Confusion Matrix for Each Models:

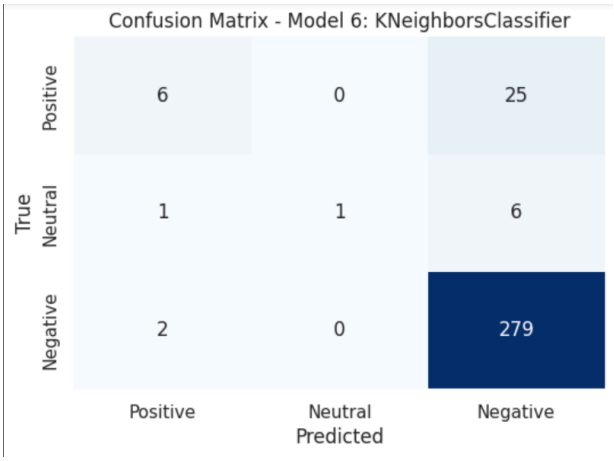












Learning Outcomes

1. **Understanding Sentiment Analysis:**
    - Gain insights into the principles and significance of sentiment analysis in extracting customer insights from reviews.
    - Learn about the role of natural language processing (NLP) and machine learning (ML) in classifying sentiments.
2. **Proficiency in VADER:**
    - Develop skills in utilizing VADER for sentiment analysis, focusing on its ability to analyze short and informal texts effectively.
    - Understand how to interpret VADER's sentiment scores, particularly the compound score for classification.
3. **Data Preprocessing Techniques:**
    - Acquire knowledge of essential text preprocessing steps, including cleaning, tokenization, stopwords removal, and stemming.
    - Recognize the importance of preprocessing in improving the accuracy of sentiment analysis.
4. **Machine Learning Application:**
    - Learn to implement and experiment with various machine learning classifiers (e.g., GaussianNB, Decision Trees, Random Forest, etc.) for sentiment classification.
    - Understand the process of feature extraction using techniques like Term Frequency-Inverse Document Frequency (TF-IDF).
5. **Model Evaluation and Metrics:**
    - Gain familiarity with model performance evaluation metrics, including accuracy, precision, recall, F1-score, and support.
    - Learn the significance of hyperparameter tuning and grid search in optimizing model performance.
6. **Handling Neutral Sentiments:**
    - Develop strategies for accurately classifying neutral sentiments, a common challenge in sentiment analysis.
    - Understand the implications of sentiment classification for business decision-making and customer satisfaction analysis.
7. **Application in Real-World Scenarios:**
    - Recognize the practical applications of sentiment analysis in e-commerce and other industries.
    - Understand how businesses can leverage sentiment analysis to enhance customer experience and drive strategic decisions.

Tools Used

1. **VADER (Valence Aware Dictionary for Sentiment Reasoning):**
    - A rule-based sentiment analysis tool for classifying sentiment in short texts.
2. **Python:**
    - The primary programming language used for implementing the sentiment analysis system.
3. **Libraries:**
    - **Pandas:** For data manipulation and analysis.
    - **NumPy:** For numerical operations and data handling.
    - **Scikit-learn:** For implementing machine learning models and evaluation metrics.
    - **NLTK (Natural Language Toolkit):** For text preprocessing and tokenization.
    - **XGBoost:** For the XGBClassifier implementation.
4. **Google colab:**
    - An interactive environment for developing and testing the code.
5. **TF-IDF Vectorizer:**
    - A feature extraction technique used to convert text data into numerical format for machine learning models.

Skills Used

1. **Natural Language Processing (NLP):**
    - Understanding and applying techniques for processing and analyzing textual data.
2. **Sentiment Analysis:**
    - Ability to classify sentiments from customer reviews using VADER and machine learning models.
3. **Data Preprocessing:**
    - Proficiency in cleaning and preparing text data for analysis, including tokenization and stopwords removal.
4. **Machine Learning:**
    - Knowledge of various classifiers (e.g., Decision Trees, Random Forests) and their implementation for sentiment classification.
5. **Model Evaluation:**
    - Skills in evaluating model performance using metrics like accuracy, precision, recall, and F1-score.
6. **Hyperparameter Tuning:**
    - Experience in optimizing machine learning models through techniques like grid search.
7. **Data Visualization:**
    - Ability to visualize results and insights from the analysis for better understanding and communication.

Sample Input and Output:

	Model	Accuracy	Precision	Recall	F1-score	Support
0	GaussianNB	0.884375	0.870665	0.884375	0.873766	320.000000
1	DecisionTreeClassifier	0.884375	0.889887	0.884375	0.883924	320.000000
2	KNeighborsClassifier	0.903125	0.893504	0.903125	0.885766	320.000000
3	RandomForestClassifier	0.918750	0.925631	0.918750	0.900040	320.000000
4	LogisticRegression	0.906250	0.915293	0.906250	0.879316	320.000000
5	XGBClassifier	0.925000	0.921430	0.925000	0.912840	320.000000

```
# Sample data: you can replace this with your actual DataFrame
data = {
    'Cleaned_Review': [
        "This product is amazing! I love it so much.",
        "This is the worst product I have ever used. Completely disappointing.",
        "The product is okay, not too bad, not too good."
    ]
}
```

```
Sample Review: This product is amazing! I love it so much.
VADER Scores: {'neg': 0.0, 'neu': 0.458, 'pos': 0.542, 'compound': 0.8516}
Predicted Sentiment: Positive

Sample Review: This is the worst product I have ever used. Completely disappointing.
VADER Scores: {'neg': 0.458, 'neu': 0.542, 'pos': 0.0, 'compound': -0.8221}
Predicted Sentiment: Negative

Sample Review: The product is okay, not too bad, not too good.
VADER Scores: {'neg': 0.17, 'neu': 0.494, 'pos': 0.336, 'compound': 0.3278}
Predicted Sentiment: Positive
```