Sentiment Analysis of Amazon Product Reviews Using Machine Learning

Shaked Yudkovich & Yair Cohen

Introduction

Sentiment analysis plays a crucial role in understanding customer opinions and feedback. In this project, we focus on classifying Amazon product reviews into three sentiment categories: **positive, negative, and neutral**. We explore different machine learning models, preprocess the dataset for optimal performance, and compare model accuracy to determine the best approach for sentiment classification.

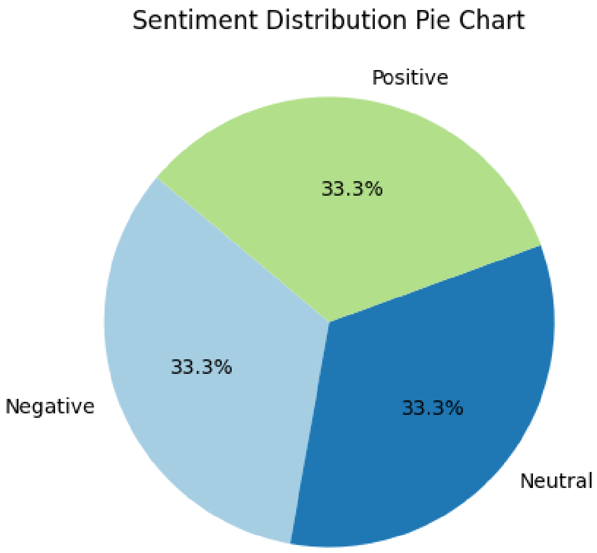
The primary research questions we aim to answer are:

- How can textual reviews be accurately classified into sentiment categories?
- Which machine learning algorithm provides the best performance for sentiment classification on the Amazon Product Reviews dataset?
- How does the classification accuracy vary between short and long reviews?

Dataset and Preprocessing

The dataset consists of Amazon product reviews with corresponding star ratings. We map these ratings to sentiment labels: (2000 samples each)

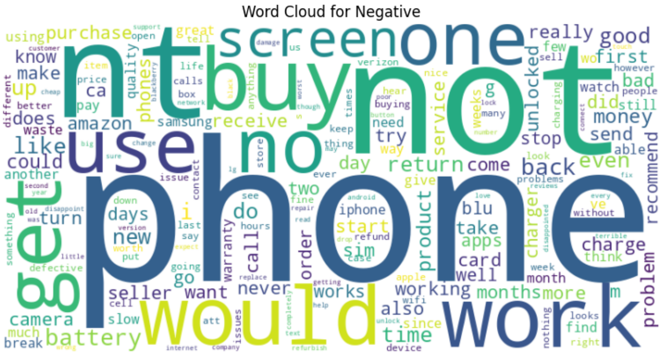
- **1-2 stars → Negative (-1)**
- **3 stars → Neutral (0)**
- **4-5 stars → Positive (1)**

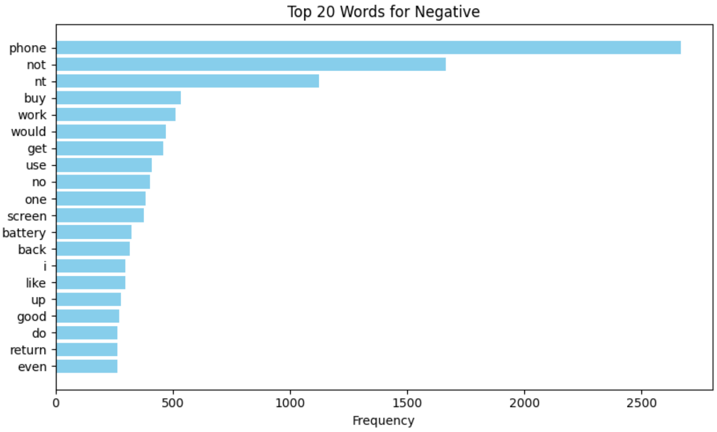


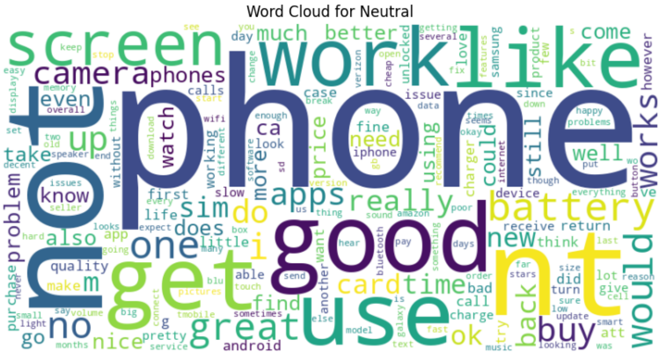
Data Cleaning and Preprocessing

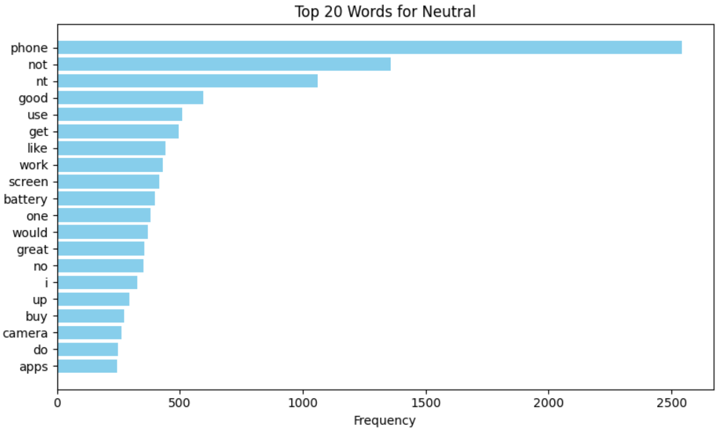
Before applying machine learning models, the dataset underwent several preprocessing steps:

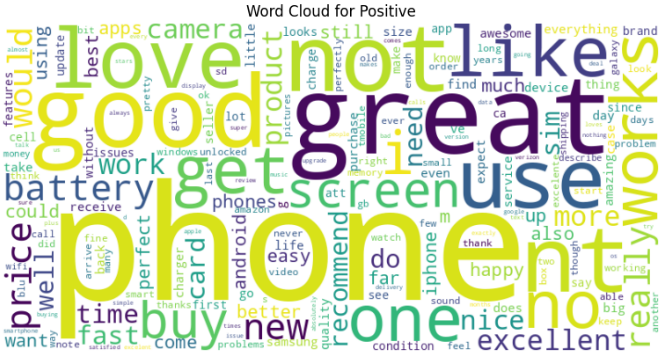
1. **Stopword Removal:** Common words that do not contribute to sentiment were removed using NLTK.
2. **Lowercasing:** Ensuring uniformity in text format.
3. **Lemmatization:** Converting words to their base form using SpaCy.
4. **Removing Irrelevant Words:** URLs, special characters, and non-alphabetical symbols were eliminated.
5. **TF-IDF Vectorization:** Transforming textual data into numerical representations for model training.

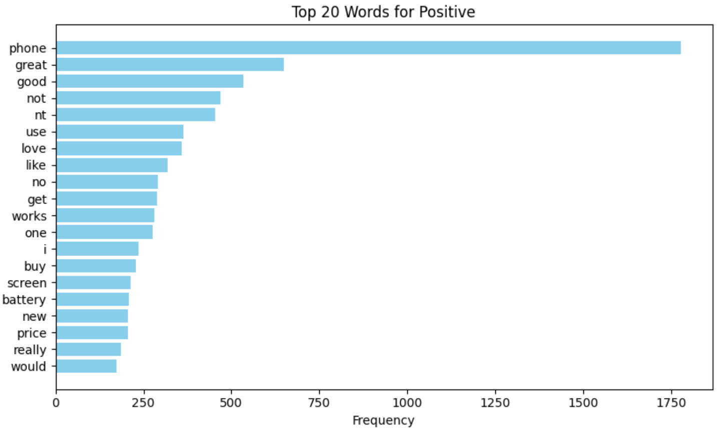












Machine Learning Models and Evaluation

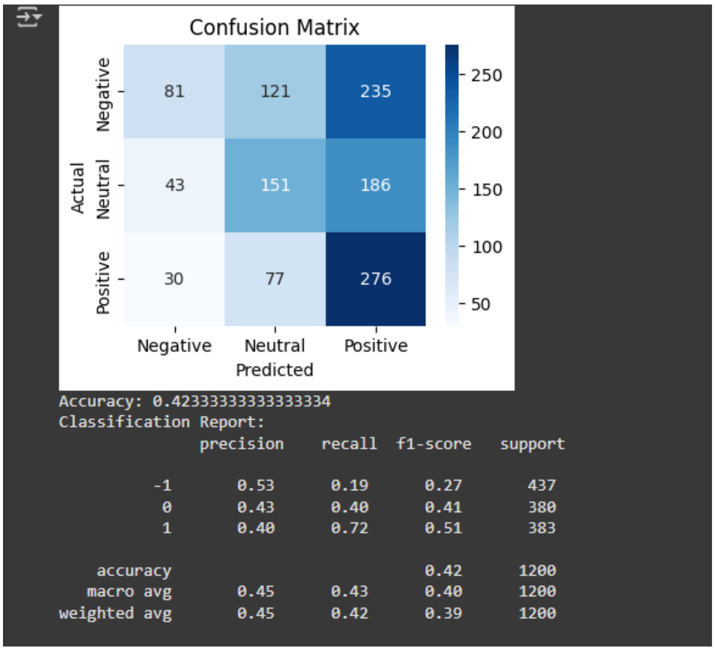
We experimented with multiple supervised learning algorithms to classify sentiment. Each model was tuned using **GridSearchCV** to optimize hyperparameters.

Models Used:

1. **K-Nearest Neighbors (KNN)**
   - GridSearchCV was applied to find the best `k` value.
   - The accuracy of KNN was relatively lower due to its sensitivity to high-dimensional text data.
2. **Decision Tree Classifier**
   - Hyperparameters such as `max_depth` and `min_samples_split` were optimized.
   - Provided interpretability but prone to overfitting.
3. **Support Vector Machine (SVM)**
   - Utilized a **linear kernel** with optimized `C` parameter.
   - Achieved a strong balance between precision and recall, making it a solid choice.
4. **Logistic Regression**
   - Tested different values of **regularization parameter `C` and `max_iter`**.
   - Performed well in classifying sentiments with competitive accuracy.

Comparison of Model Performance

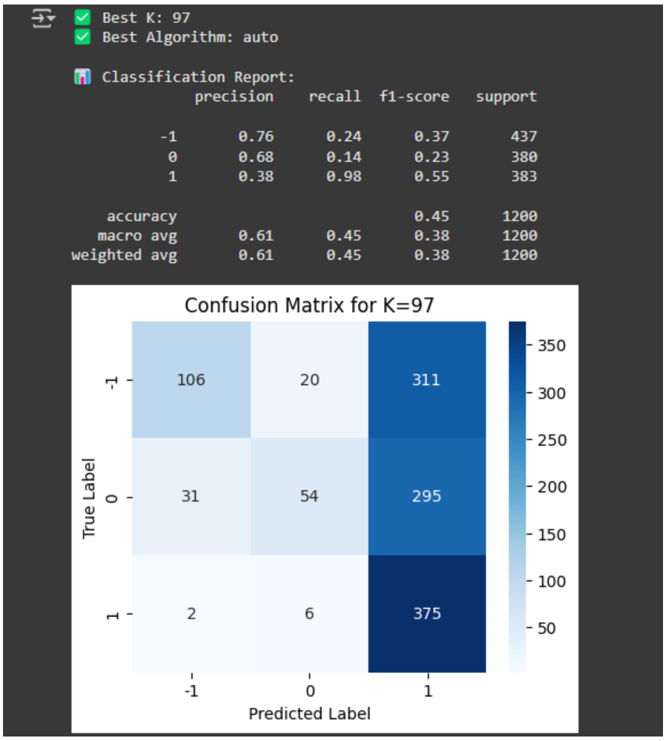
KNN:



```
Accuracy: 0.42333333333333334
Classification Report:
              precision    recall  f1-score   support

          -1       0.53      0.19      0.27       437
           0       0.43      0.40      0.41       380
           1       0.40      0.72      0.51       383

    accuracy                           0.42      1200
   macro avg       0.45      0.43      0.40      1200
weighted avg       0.45      0.42      0.39      1200
```



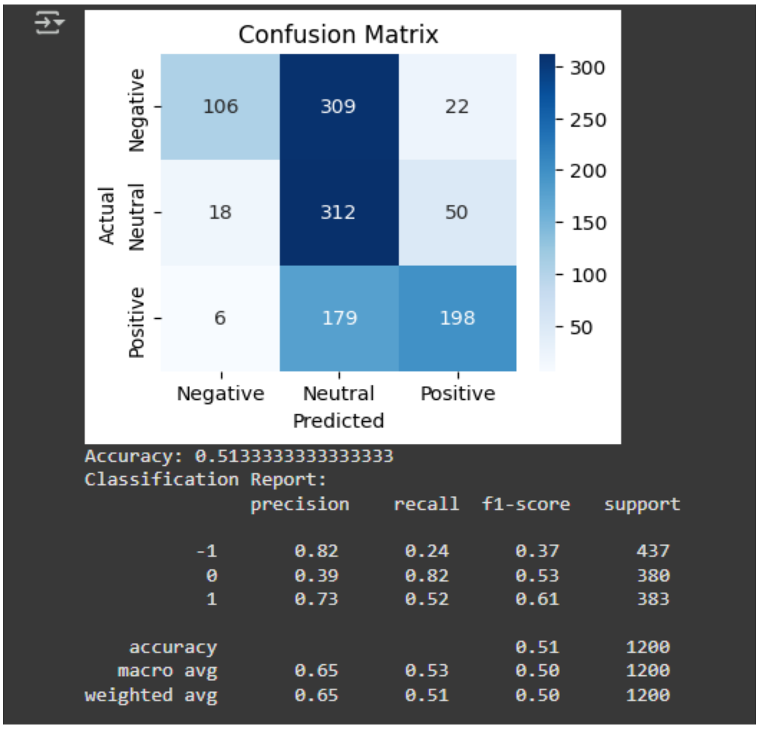
```
✅ Best K: 97
✅ Best Algorithm: auto

📊 Classification Report:
              precision    recall  f1-score   support

          -1       0.76      0.24      0.37       437
           0       0.68      0.14      0.23       380
           1       0.38      0.98      0.55       383

    accuracy                           0.45      1200
   macro avg       0.61      0.45      0.38      1200
weighted avg       0.61      0.45      0.38      1200
```

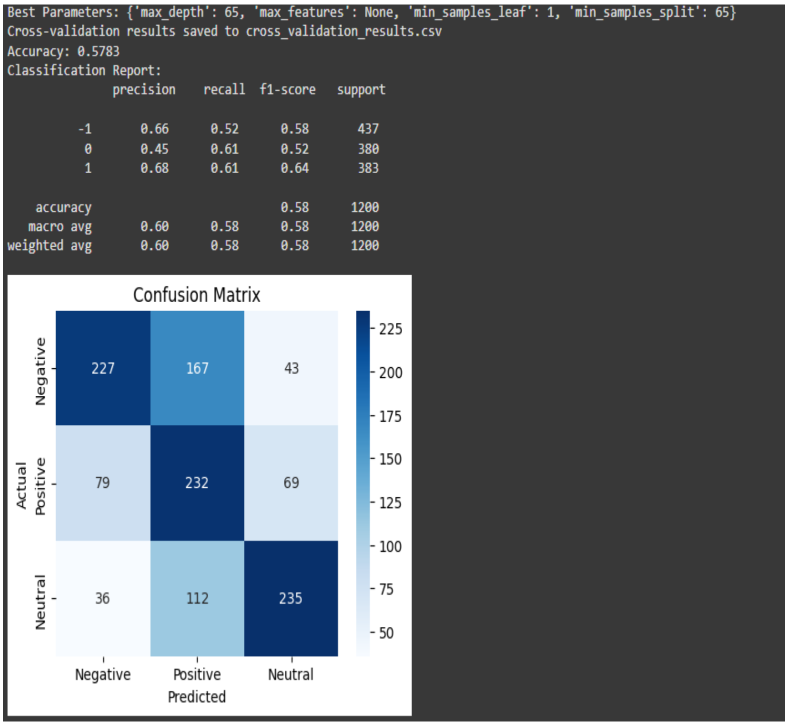
Decision Tree Classifier:



```
Accuracy: 0.5133333333333333
Classification Report:
              precision    recall  f1-score   support

          -1       0.82      0.24      0.37       437
           0       0.39      0.82      0.53       380
           1       0.73      0.52      0.61       383

    accuracy                           0.51      1200
   macro avg       0.65      0.53      0.50      1200
weighted avg       0.65      0.51      0.50      1200
```

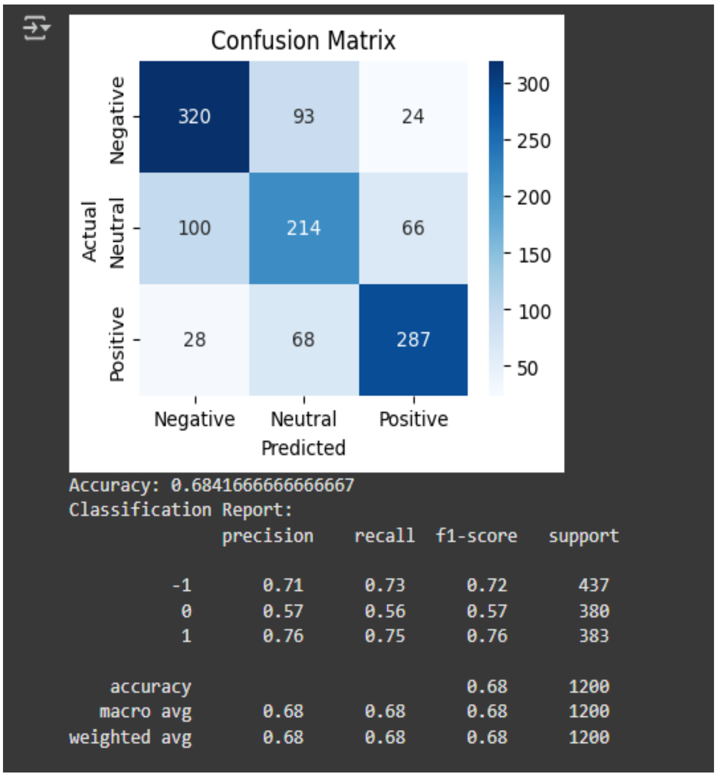


```
Best Parameters: {'max_depth': 65, 'max_features': None, 'min_samples_leaf': 1, 'min_samples_split': 65}
Cross-validation results saved to cross_validation_results.csv
Accuracy: 0.5783
Classification Report:
              precision    recall  f1-score   support

          -1       0.66      0.52      0.58       437
           0       0.45      0.61      0.52       380
           1       0.68      0.61      0.64       383

    accuracy                           0.58      1200
   macro avg       0.60      0.58      0.58      1200
weighted avg       0.60      0.58      0.58      1200
```

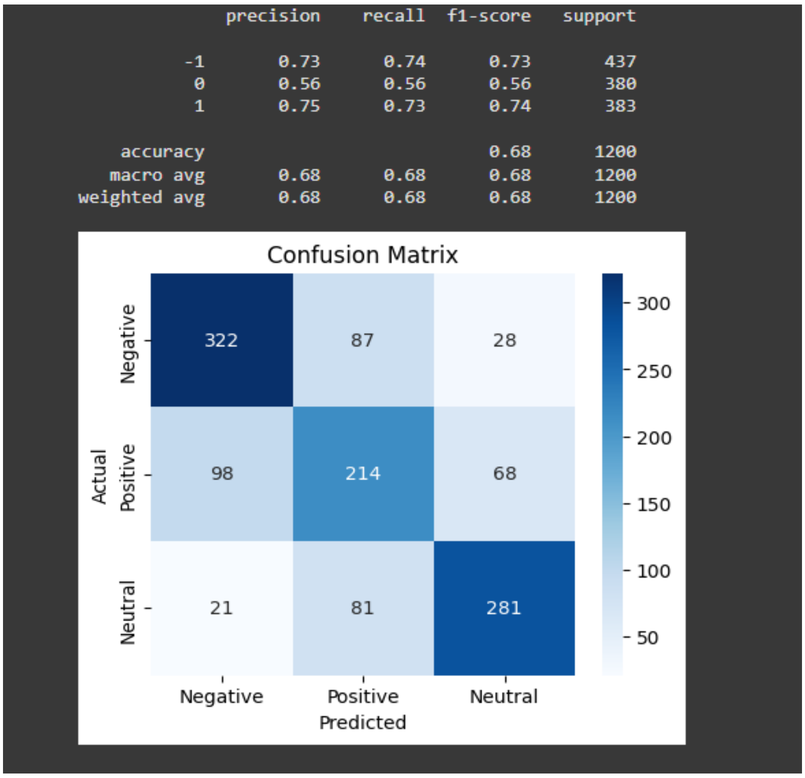
SVM:



```
Accuracy: 0.6841666666666667
Classification Report:
              precision    recall  f1-score   support

          -1       0.71      0.73      0.72       437
           0       0.57      0.56      0.57       380
           1       0.76      0.75      0.76       383

    accuracy                           0.68      1200
   macro avg       0.68      0.68      0.68      1200
weighted avg       0.68      0.68      0.68      1200
```

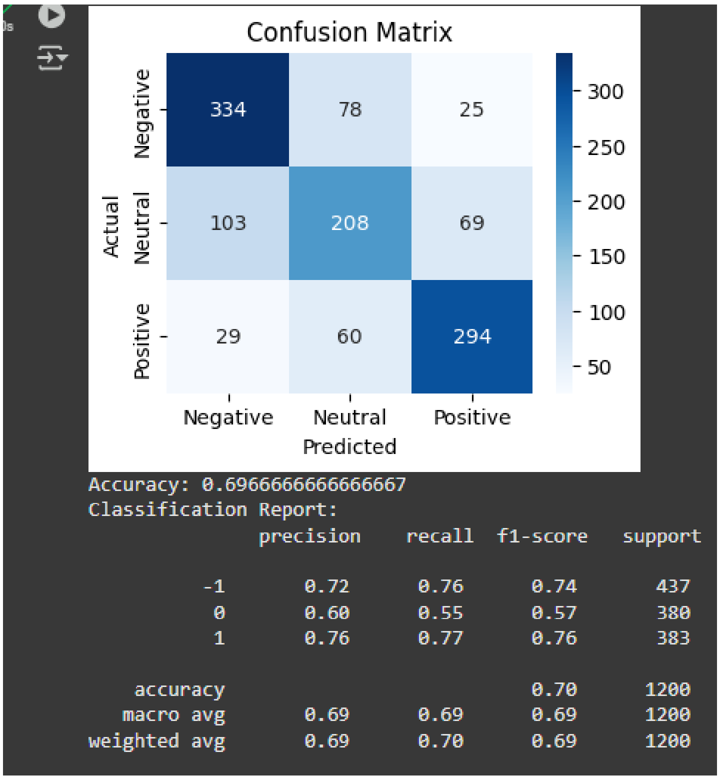


```
              precision    recall  f1-score   support

          -1       0.73      0.74      0.73       437
           0       0.56      0.56      0.56       380
           1       0.75      0.73      0.74       383

    accuracy                           0.68      1200
   macro avg       0.68      0.68      0.68      1200
weighted avg       0.68      0.68      0.68      1200
```

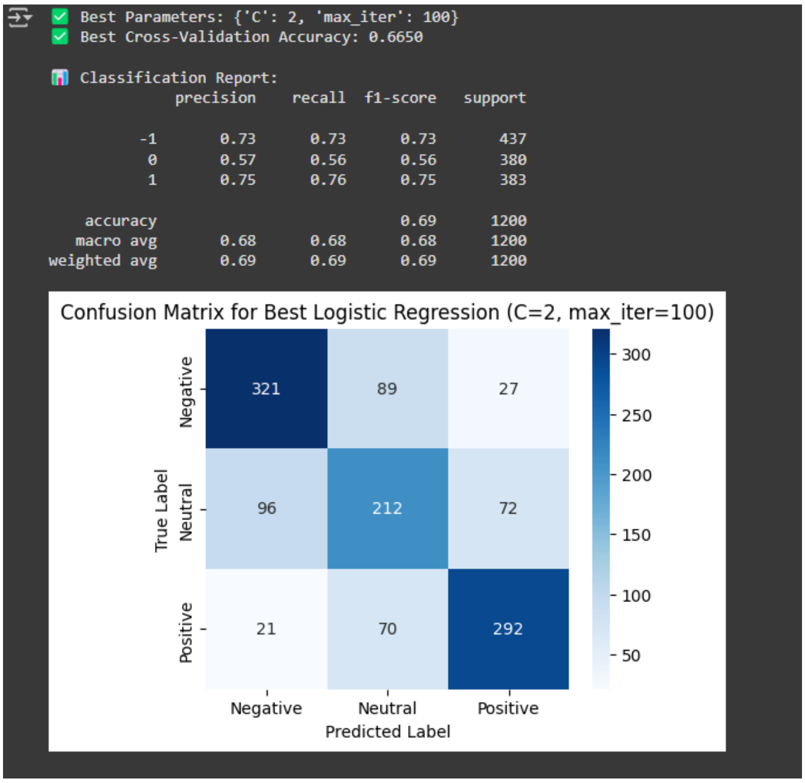
Logistic Regression:



```
Accuracy: 0.6966666666666667
Classification Report:
              precision    recall  f1-score   support

          -1       0.72      0.76      0.74       437
           0       0.60      0.55      0.57       380
           1       0.76      0.77      0.76       383

    accuracy                           0.70      1200
   macro avg       0.69      0.69      0.69      1200
weighted avg       0.69      0.70      0.69      1200
```



```
✅ Best Parameters: {'C': 2, 'max_iter': 100}
✅ Best Cross-Validation Accuracy: 0.6650

📊 Classification Report:
              precision    recall  f1-score   support

          -1       0.73      0.73      0.73       437
           0       0.57      0.56      0.56       380
           1       0.75      0.76      0.75       383

    accuracy                           0.69      1200
   macro avg       0.68      0.68      0.68      1200
weighted avg       0.69      0.69      0.69      1200
```

Observations:

- **Logistic Regression and SVM achieved the highest accuracy (~68-69%)**, making them the most effective models for sentiment classification.
- **KNN struggled due to high dimensionality** of TF-IDF vectorized text data.
- **Decision Tree provided interpretability but did not generalize as well as SVM.**

Observations:

- **Logistic Regression and SVM achieved the highest accuracy (~68-69%)**, making them the most effective models for sentiment classification.
- **KNN struggled due to high dimensionality** of TF-IDF vectorized text data.
- **Decision Tree provided interpretability but did not generalize as well as SVM.**

Challenges and Solutions

Challenges Faced:

1. **Handling Imbalanced Classes**: Some sentiment classes had fewer samples.
   - *Solution*: Adjusted training set balance and used cross-validation.
2. **Choosing the Best Model**: Different algorithms had varying strengths.
   - *Solution*: GridSearchCV was used to fine-tune hyperparameters.
3. **Processing Long and Short Reviews**: Short reviews lacked context, making classification harder.
   - *Solution*: TF-IDF helped capture meaningful word importance.

How Sentiment Classification Varies Between Short and Long Texts

- **Short reviews** often contained ambiguous words, leading to misclassification.
- **Longer reviews** provided more context, improving classification accuracy.

Conclusions

- **Logistic Regression and SVM were the most effective models**, offering the highest classification accuracy.
- **Data preprocessing was essential** for improving model performance.
- **Sentiment classification accuracy improved with longer reviews** due to additional context.