**Internship Project**
**On**

# SENTIMENT ANALYSIS ON AMAZON PRODUCT REVIEWS

**A project submitted to**
**Technocolabs, Indore**


**For fulfillment of**

## INTERNSHIP CERTIFICATE
## IN
## DATA ANALYTICS

**Under the Guidance:**
**Yaseen Shah, CEO, Technocolabs**
**Dr. S. Gomathi**

**Submitted By:**
**Abhishek De**
**Chaitra Bellur**
**Kotha Lokesh**
**Prashant Srivastava**
**Rahul Roy**
**Sanurhanaan Shaikh**

# CERTIFICATE OF EXAMINATION

This is to certify that we have examined the summer internship project report on "**Sentiment Analysis on Amazon Product Reviews**" and hereby accord our approval of it as an initial attempt carried out by **Abhishek De, Chaitra Bellur, Kotha Lokesh, Prashant Srivastava, Rahul Roy,** and **Sanurhanaan Shaikh.**

In the manner required for its acceptance in partial fulfillment of Internship certificate course for which it is submitted.

**Approved by:**

**Yaseen Shah**

**CEO, Technocolabs**

# ACKNOWLEDGEMENT

We express our deep sense of gratitude to our respected and learned guides for their valuable help and guidance, we are thankful to them for the encouragement they have given us in completing the project.

We are also grateful to respected, **Yaseen Shah, CEO, Technocolabs, Indore** for his constant support and encouragement which helped us throughout the internship.

We would also like to express our sincere thanks to our mentor, **Dr. S. Gomathi,** for her guidance, cooperation and supervision throughout this internship.

**Team-A, Technocolabs**

# CONTENT

# ABSTRACT

Sentiment analysis is the process of detecting positive or negative sentiment in text. It's often used by businesses to detect sentiment in social data, gauge brand reputation, and understand customers.

Since customers express their thoughts and feelings more openly than ever before, sentiment analysis is becoming an essential tool to monitor and understand that sentiment. Automatically analyzing customer feedback, such as opinions in survey responses and social media conversations, allows brands to learn what makes customers happy or frustrated, so that they can tailor products and services to meet their customers' needs.

For example, using sentiment analysis to automatically analyze 4,000+ reviews about your product could help you discover if customers are happy about your pricing plans and customer service.

# INTRODUCTION

Sentiment analysis is the process of computationally identifying and categorizing opinions expressed in a piece of text, especially in order to determine whether the writer's attitude towards a particular topic, product, etc. is positive, negative, or neutral.

**Types of Sentiment Analysis:**

Sentiment analysis models focus on polarity (positive, negative, neutral) but also on feelings and emotions (angry, happy, sad, etc), urgency (urgent, not urgent) and even intentions (interested v. not interested).

**Importance of Sentiment Analysis:**

Sentiment analysis is extremely important because it helps businesses quickly understand the overall opinions of their customers. By automatically sorting the sentiment behind reviews, social media conversations, and more, you can make faster and more accurate decisions.

# DATA PREPROCESSING

Initially, data preprocessing involves importing the required libraries NumPy, Pandas, Matplotlib, Seaborn, etc. and used Jupyter Notebook as IDE.

Data preprocessing is a data mining technique that is used to transform raw data into a useful and efficient format.

Steps involved:

1. **Data Cleaning:**

   Data cleansing or data cleaning is the process of detecting and correcting corrupt or inaccurate records from a record set, table, or database and refers to identifying incomplete, incorrect, inaccurate, or irrelevant parts of the data and then replacing, modifying, or deleting the dirty or coarse data.

   Dropping the Missing values:

   Since we were unable to figure the cause of the missing values, so we decided to remove them.

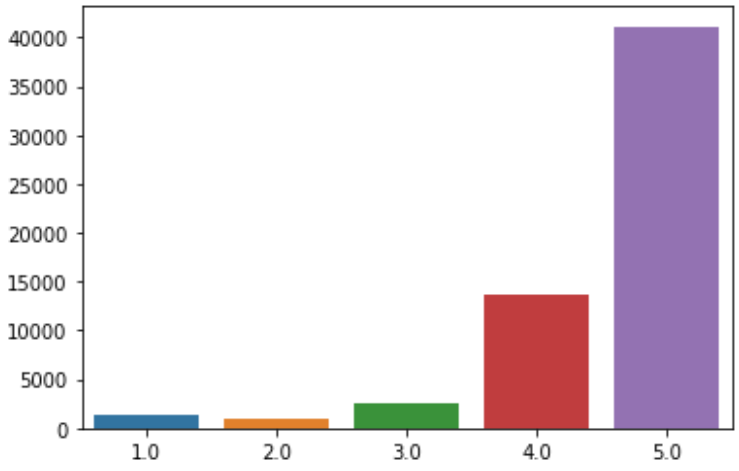
## 2. Data Transformation:

This step is taken in order to transform the data into appropriate forms suitable for the mining process.

In data transformation, we have transformed some of the texts written in the Reviews column into a vector form for the Machine Learning processing.

## 3. Data Visualization:

Data visualization is an interdisciplinary field that deals with the graphic representation of data.




We have analyzed the Ratings column to find out the number of people rating a product on Amazon.

# MODEL BUILDING AND HYPERTUNING

Support Vector Machine (SVM) is a relatively simple **Supervised Machine Learning Algorithm** used for classification and/or regression. In SVM, we plot each data item in the dataset in an N-dimensional space, where N is the number of features/attributes in the data. SVM can only perform binary classification (eg: positive and negative).

Basically, SVM finds a hyper-plane that creates a boundary between the types of data. The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.
To perform SVM on multi-class problems, we can create a binary classifier for each class of the data. The two results of each classifier will be:

- The data point belongs to that class OR
- The data point does not belong to that class.

**Linear SVM:** Linear SVM is used for linearly separable data, which means if a dataset can be classified into two classes by using a single straight line, then such data is termed as linearly separable data, and classifier is used called as Linear SVM classifier.

## Gaussian Radial Basis Function (RBF)

It is one of the most preferred and used kernel functions in svm. It is usually chosen for non-linear data. It helps to make proper separation when there is no prior knowledge of data.

RBF Kernel is popular because of its similarity to K-Nearest Neighborhood Algorithm. It has the advantages of K-NN and overcomes the space complexity problem as RBF Kernel Support Vector Machines just needs to store the support vectors during training and not the entire dataset.

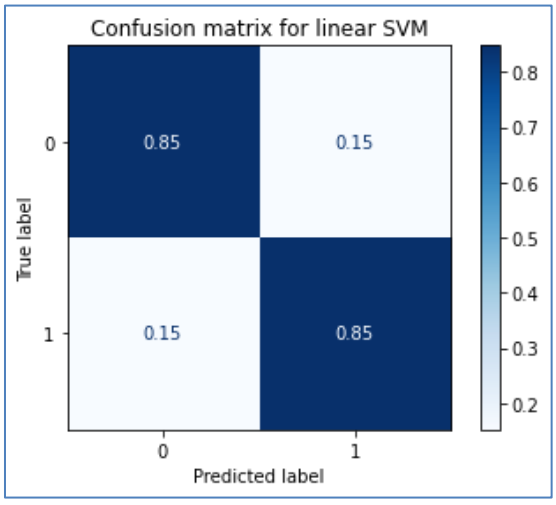
## Model with Linear SVM

```
from sklearn.svm import SVC
from sklearn.pipeline import Pipeline

# Create the SVM
svm_linear = Pipeline([("clf_linearSVC", SVC(random_state=42, kernel='linear'))])

# Fit the data to the SVM classifier
svm_linear.fit(X_train, y_train)

# Generate predictions
Y_Pred_lin = svm_linear.predict(X_test)

#Accuracy
print('Train accuracy :', (svm_linear.score(X_train, y_train))*100)
print('Test accuracy :', (svm_linear.score(X_test, y_test))*100)

#confusion matrix
matrix = plot_confusion_matrix(svm_linear, X_test, y_test,
                               cmap=plt.cm.Blues,
                               normalize='true')
plt.title('Confusion matrix for linear SVM')
plt.show(matrix)
plt.show()
```

Output:

Train accuracy: 95.21116138763198

Test accuracy: 84.77386934673366



## Model with Radial Basis Function SVM

```
svm_rbf = Pipeline([("clf_rbfSVC", SVC(random_state=42, kernel='rbf'))])
svm_rbf.fit(X_train, y_train)
Y_Pred_rbf = svm_rbf.predict(X_test)
```

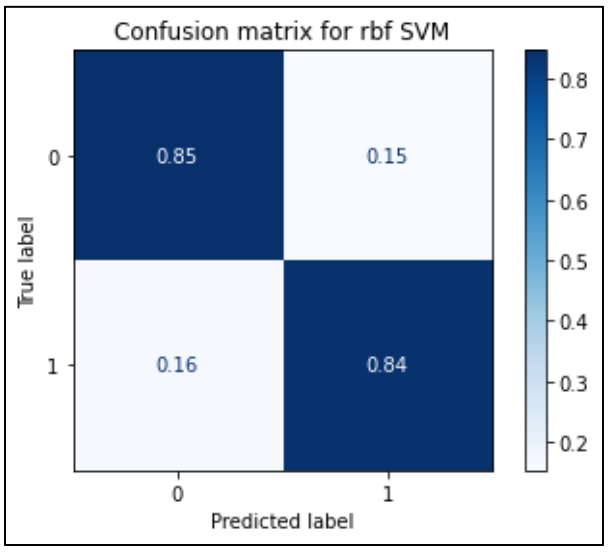
```
print('Train accuracy :', (svm_rbf.score(X_train, y_train))*100)
print('Test accuracy :', (svm_rbf.score(X_test, y_test))*100)
```

```
#confusion matrix
matrix = plot_confusion_matrix(svm_rbf, X_test, y_test,
                               cmap=plt.cm.Blues,
                               normalize='true')
plt.title('Confusion matrix for rbf SVM')
plt.show(matrix)
plt.show()
```

Output:

Train accuracy : 93.31322272498743

Test accuracy : 84.321608040201



## Fine-Tuning

```
print('Train accuracy :', (grid.best_estimator_.score(X_train, y_train))*100)
print('Test accuracy :', (grid.best_estimator_.score(X_test, y_test))*100)
```

```
grid_predictions = grid.best_estimator_.predict(X_test)
```

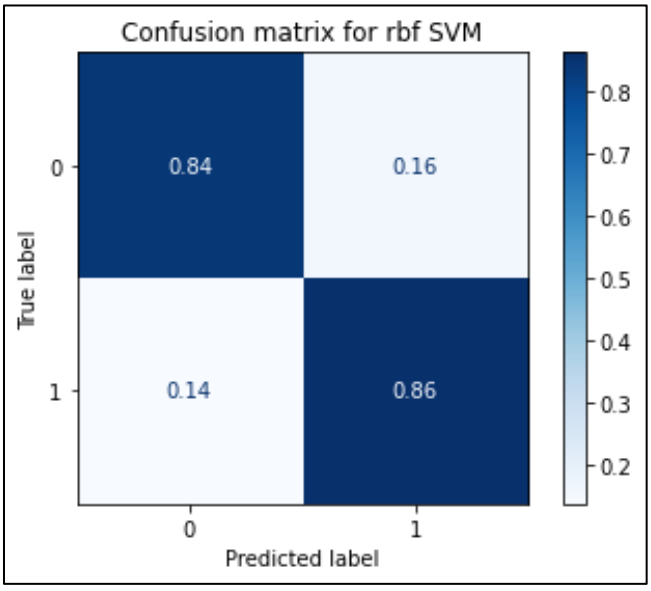
```
# print best parameter after tuning
print(grid.best_params_)

# print how our model looks after hyper-parameter tuning
print(grid.best_estimator_)
```

```
#confusion matrix
matrix = plot_confusion_matrix(grid, X_test, y_test,
                               cmap=plt.cm.Blues,
                               normalize='true')
plt.title('Confusion matrix for rbf SVM')
plt.show(matrix)
plt.show()
```

Output:

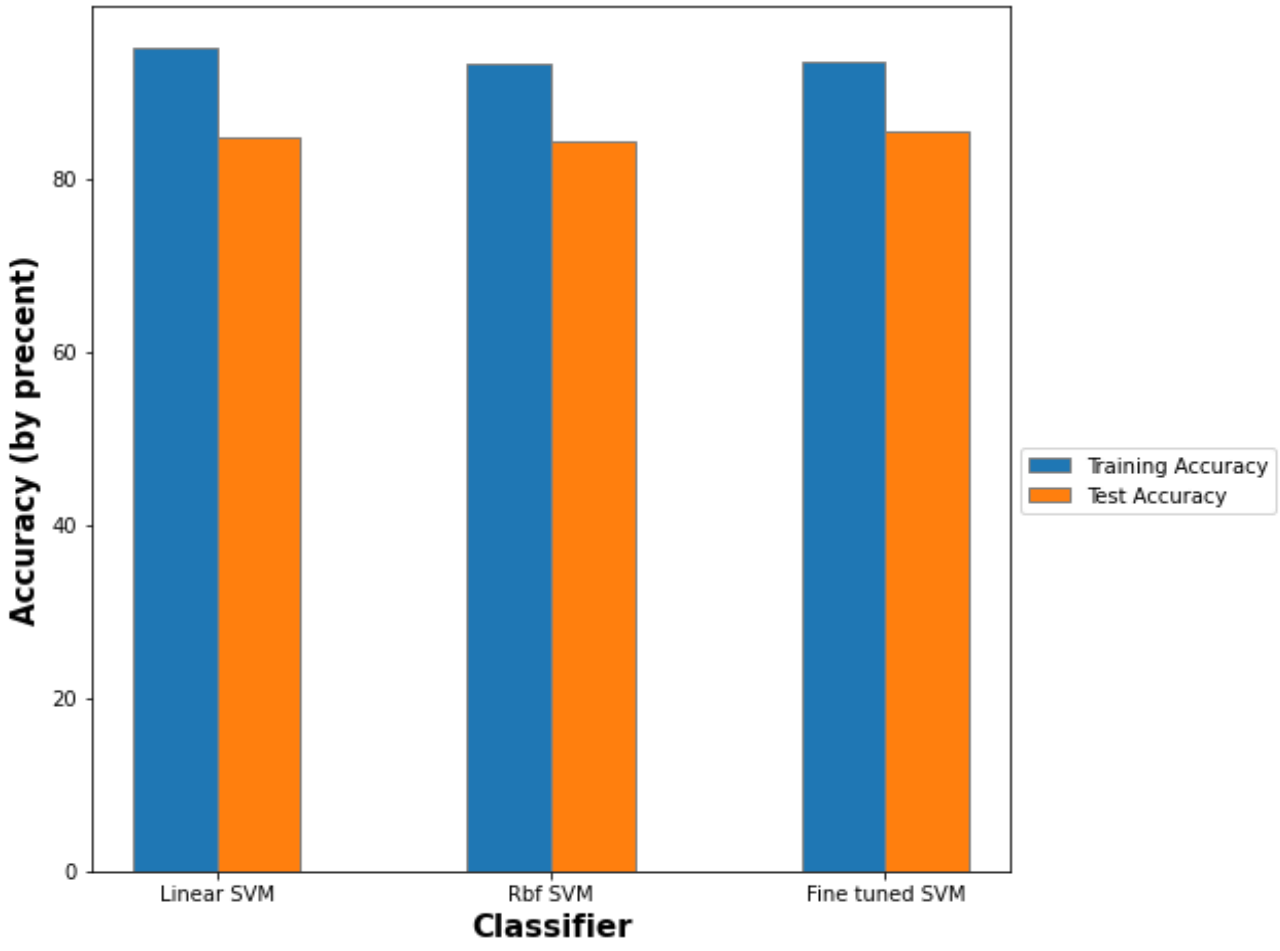
Train accuracy : 93.48919054801408

Test accuracy : 85.37688442211055



## Comparison between the models

| | Name | Training Accuracy | Test Accuracy |
|---|---|---|---|
| 0 | Linear SVM | 95.211161 | 84.773869 |
| 1 | Rbf SVM | 93.313223 | 84.321608 |
| 2 | Fine tuned SVM | 93.489191 | 85.376884 |



# MODEL DEPLOYMENT

Deployment of an ML-model simply means the integration of the model into an existing production environment which can take in an input and return an output that can be used in making practical business decisions.

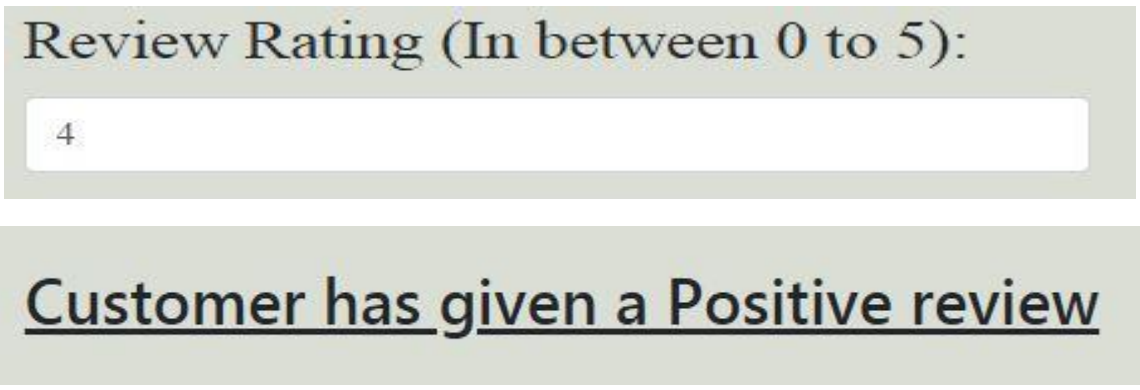
In the deployment part, we have included 2 sections:

1. Web Page Deployment
2. Flask Server

We created a website using HTML Programming and used Flask Server to connect it to the server.

Here, customers can log-in to provide the product feedback and we can predict the customer's sentiment towards the purchased product.

Website link: https://amaz-reviews.herokuapp.com



# REFERENCES


- https://www.w3schools.com/html/
- https://www.tutorialspoint.com/flask/flask_templates.htm
- https://github.com/ishikaarora/Aspect-Sentiment-Analysis-on-Amazon-Reviews
- https://github.com/Maha41/Sentiment-analysis-on-Amazon-Reviews-using-Python
- https://www.youtube.com/watch?v=VXt9SQx5eM0
- https://www.youtube.com/watch?v=2Jk96Tl0-dI