

**MSc in AI - Assessment Cover Sheet**

Complete and attach this cover sheet to your assessment before submitting

**Course Code and Title:**

IT9002– Natural Language Processing

**Assessment Title:**

**Learning Outcomes:**

LO1 – Understanding the critical knowledge of fundamental concepts, algorithms, and models in Natural Language processing (NLP) for performing various linguistic NLP tasks.

LO2 - Demonstrate professional levels of insight, interpretation by utilizing the various NLP packages like Natural Language Tool Kit (NLTK) to apply, solve, implement,

evaluate, and improve the real time significant applications of NLP

Individual Project

**Student IDs:**

**12011066**

**Student Names:**

**Sharifa Sayed Ali**

**Tutor:**

Sini Raj Pulari

**Individual Project Submission Date**

**Tuesday,19th December** 2023 by 11:55 p.m.

**Late Rule:**

**The maximum grade granted for late submission is 60 % for up to 3 calendar days. A**

**grade of 0 will be allocated for submission after 3 days**

***By submitting this assessment for marking, either electronically or as hard copy, I confirm the following:***

* This assignment is **our own work**
* Any information used has been properly referenced.
* I understand that a copy of my work may be used for moderation.
* I have kept a copy of this assignment

Do not write below this line. For Polytechnic use only*.*

**Assessor:**

**Date of Marking:**

**Grade/Mark:**

Table of Contents

[Task 1 – Problem Statement Formulation and definition 3](#_Toc152957793)

[ Motivation 3](#_Toc152957794)

[ Problem Statement/ project definition: 3](#_Toc152957795)

[ Expected Result - What are you trying to achieve: 3](#_Toc152957796)

[Task 2 – Selection of an appropriate dataset (Data collection) 3](#_Toc152957797)

[ Dataset selection 3](#_Toc152957798)

[ Initial data visualization: 4](#_Toc152957799)

[ Sentiment Labels 6](#_Toc152957800)

[Task 3 –Text Preprocessing 6](#_Toc152957801)

[ Text processing for the “reviewsText” column 6](#_Toc152957802)

[Task 4- Text Representation 7](#_Toc152957803)

[ POS tagging 7](#_Toc152957804)

[ Word Embedding 7](#_Toc152957805)

[ Bag of Words: 8](#_Toc152957806)

[Task 5 –Text Classification / Prediction 8](#_Toc152957807)

[ Text Classifier Using Multinomial Naive Bayes Classifier 8](#_Toc152957808)

[ Logistic Regression: 10](#_Toc152957809)

[ Support Vector Machines (SVM): 11](#_Toc152957810)

[Task 6 – Evaluation, Inferences, Recommendation and Reflection 13](#_Toc152957811)

[Evaluation 13](#_Toc152957812)

[Inferences: 14](#_Toc152957813)

[Reflections: 14](#_Toc152957814)

[Possible improvements: 14](#_Toc152957815)

# Task 1 – Problem Statement Formulation and definition

## Motivation

The rapid growth of e-commerce platforms like Amazon has resulted in a massive volume of customer reviews generated daily. These reviews hold a wealth of valuable insights and opinions regarding products and services. Extracting meaningful information from these reviews can provide businesses with essential feedback, enabling them to gauge customer sentiments and make informed decisions to enhance their offerings.

## Problem Statement/ project definition:

The problem I aim to address is to analyze customer reviews on Amazon to gain insights into customer sentiments. My primary objective is to classify product reviews into two distinct categories: positive and negative. To accomplish this, I will employ a range of natural language processing and machine learning techniques to accurately analyze and classify the emotions conveyed in the reviews.

## Expected Result - What are you trying to achieve:

* Sentiment Analysis: Develop a model capable of classifying customer reviews into positive or negative sentiments. This will assist businesses in understanding the overall sentiment towards their products or services and pinpointing areas for improvement.
* Topic Extraction: Extract key topics or themes from customer reviews to identify common issues or features frequently mentioned by customers. This will allow businesses to focus on specific aspects of their products or services that require attention or enhancement.
* Trend Analysis: Analyze the temporal patterns of customer sentiments and identify trends over time. This will help businesses monitor changes in customer perceptions and adapt their strategies accordingly.

By achieving these results, businesses can gain valuable insights from customer reviews, make data-driven decisions, and improve their products or services based on customer feedback. This, in turn, can lead to improved customer satisfaction, increased sales, and a competitive edge in the e-commerce market.

# Task 2 – Selection of an appropriate dataset (Data collection)

## Dataset selection

I selected “amazon\_reviews.csv” dataset[[1]](#footnote-1) from Kaggle.com. This dataset containing Amazon Product Data includes product categories and various metadata. The product with the most comments in the electronics category has user ratings and comments. In this way, I expect to perform sentiment analysis of the reviews (positive or negative) with specific methods. The reasons for choosing this dataset are:

1. Accessibility: The dataset is conveniently accessible and can be easily obtained for analysis and modeling tasks.
2. Relevance: Customer reviews are critically important in determining the success of products on e-commerce platforms like Amazon. Sentiment analysis and prediction can provide valuable insights into customer preferences and product performance.

**Dataset features:**

|  |  |
| --- | --- |
| **Feature** | **Description** |
| Unnamed | index |
| reviewerName | User name. |
| overall | Product rating |
| reviewText | Evaluation Summary |
| reviewTime | Evaluation Time |
| day\_diff | Number of days since assessment |
| helpful\_yes | The number of times the evaluation was found useful. |
| helpful\_no | Number of people who didn't support the comment and didn't find it helpful |
| total\_vote | Number of votes given to the evaluation |
| score\_pos\_neg\_diff | score poz-neg |

## Initial data visualization:

- After importing the dataset and storing it in full\_df data frame, display the first five rows from the it using full\_df.head() command.

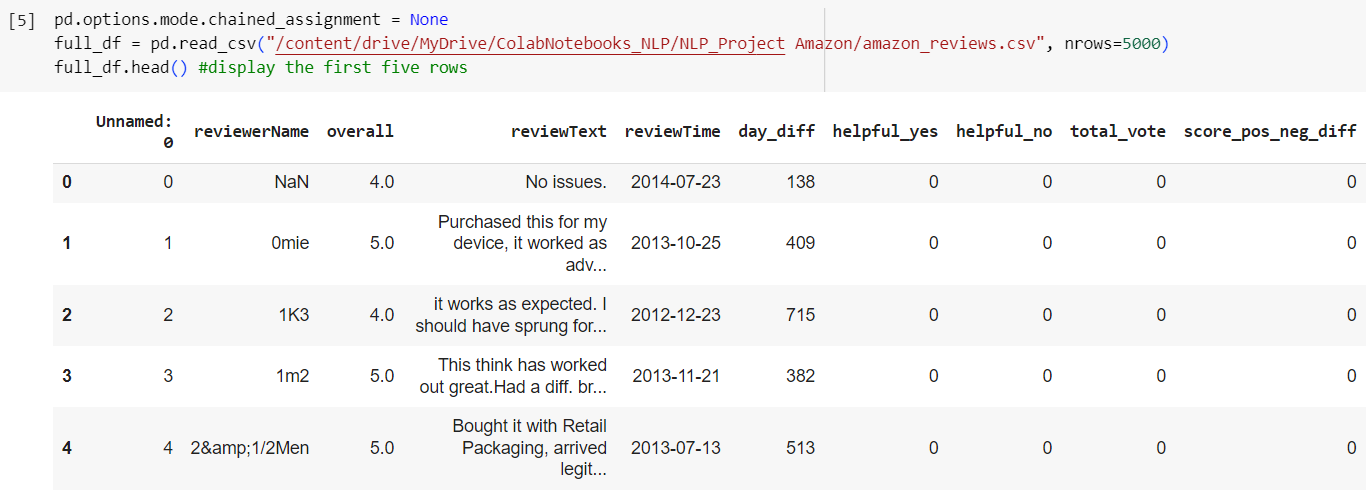


Figure : reading and displaying the first five rows of the dat

- Using full\_df.dtypes to display the datatype of each column

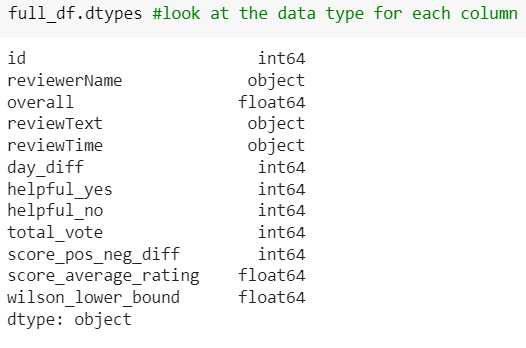


Figure : data type of each column

- Using full\_df.shape to display the dimensions of the dataset (number of rows, number of columns).

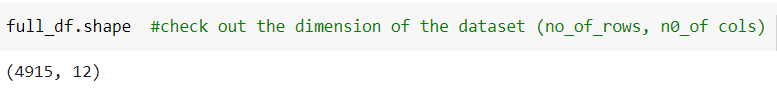


Figure : dataset shape

- Using full\_df.describe() to display the data summary

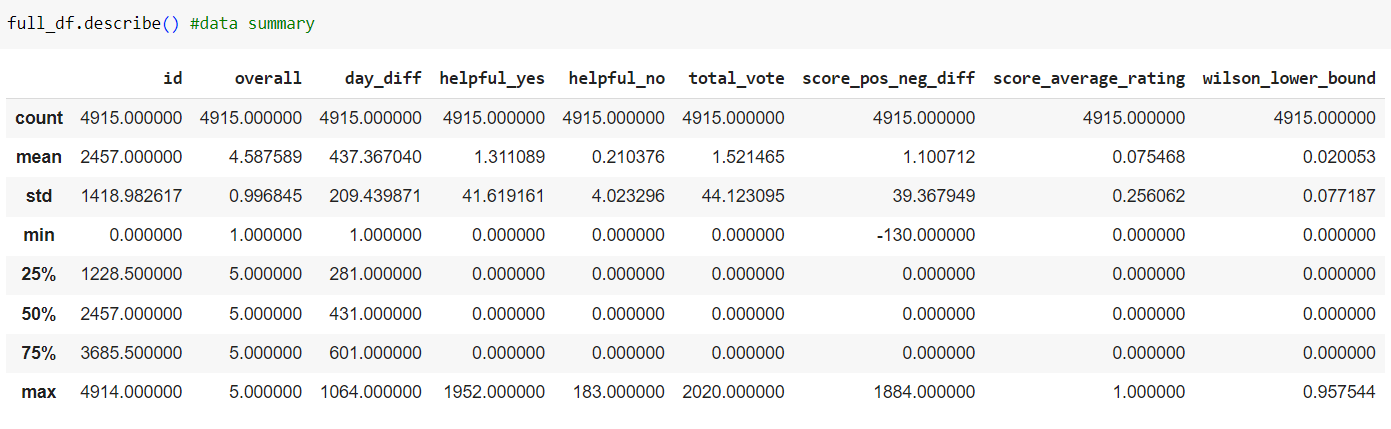


Figure : data summary

- visualize the distribution of ratings

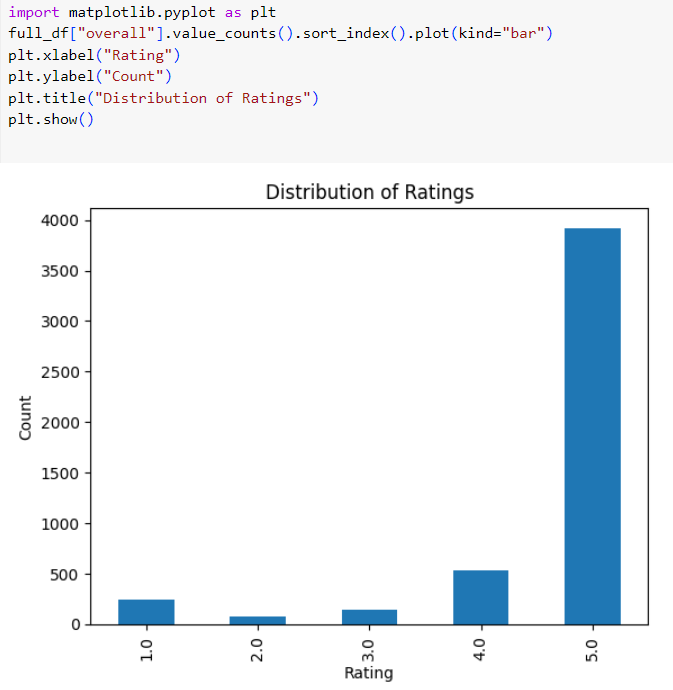


Figure : example of data visualization

The chart shows that the distribution of ratings is positively skewed which means there are more high ratings than low ratings. The chart also shows that the most rating is 5, followed by 4. This indicates that the overall sentiment is positive.

## ****Sentiment Labels****

* + Convert labels to positive or negative based on overall ratings.
  + Based on the overall, the number of Positive & Negative messages are counted and plotted

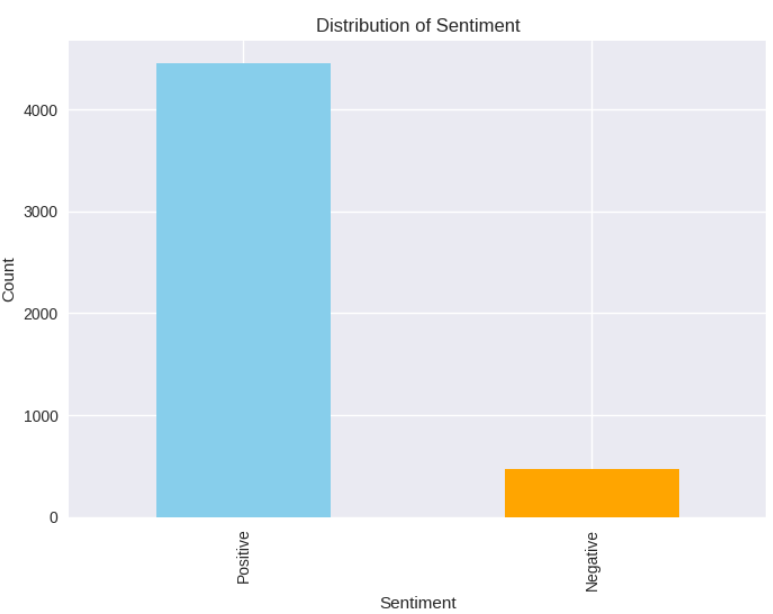


Figure : sentiment labels

# Task 3 –Text Preprocessing

## ****Text processing for the “reviewsText” column****

* + Lower casing
  + Removing punctuation
  + Removing numbers
  + Removing stop words
  + Stemming: the process of reducing inflected (or sometimes derived) words to their word stem, base, or root form
  + Lemmatization: the process of reducing words to their base or dictionary form. NLTK provides the WordNetLemmatizer for this purpose.

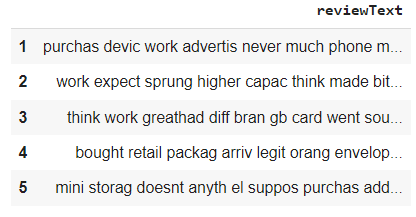


Figure : reviewText after preprocessing

* + Tokenization: the process of splitting text into individual words or tokens.

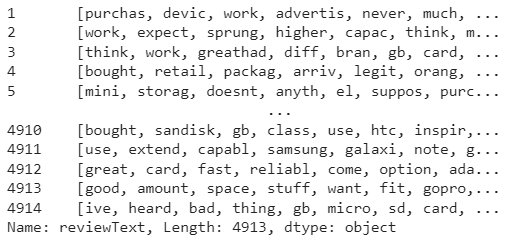


Figure : reviewText after tokenization

# Task 4- Text Representation

## ****POS tagging****

Is the process of assigning a grammatical label (tag) to each word in a sentence, indicating its part of speech (e.g., noun, verb, adjective).

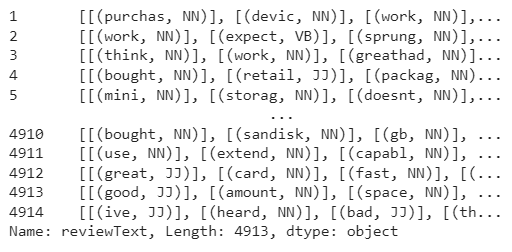


Figure : POS tagging

## ****Word Embedding****

**Word2Vec is a technique used in natural language processing (NLP) to represent words as dense numerical vectors in a continuous vector space. It is a form of distributed representation that captures the semantic and syntactic meaning of words based on their context within a given corpus.**

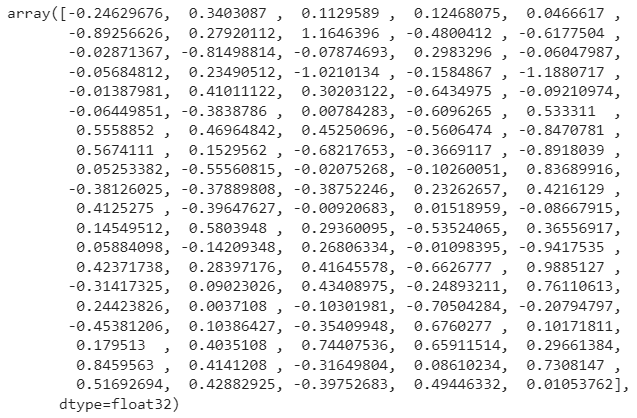


Figure : Word vector for "good"

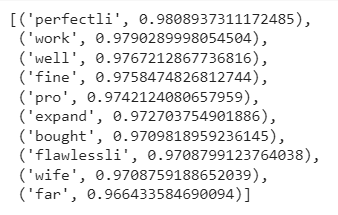


Figure : similar words to "great"

## ****Bag of Words:****

**Bag of Words technique, which represents text data as a matrix of word counts.**



Figure : part of Bag of Words representation

# Task 5 –Text Classification / Prediction

## ****Text Classifier Using Multinomial Naive Bayes Classifier****

“The multinomial Naive Bayes classifier is suitable for classification with discrete features (e.g., word counts for text classification). The multinomial distribution normally requires integer feature counts. However, in practice, fractional counts such as tf-idf may also work”[[2]](#footnote-2).

The classification report provides additional evaluation metrics for each class (negative and positive) based on the confusion matrix. It includes metrics such as precision, recall, and F1-score, along with support.

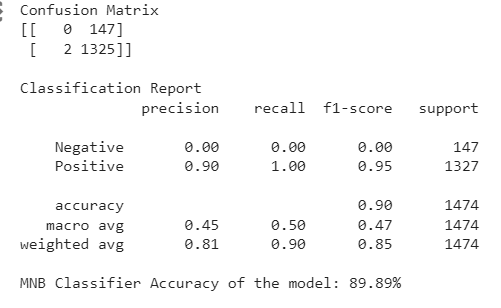


Figure : Multinomial Naive Bayes Classifier results

A chart with a blue and yellow squares

Description automatically generated

Figure : Multinomial Naive Bayes heatmap

A confusion matrix is a table that visualizes the performance of a classification model by showing the counts of true positive (TP), false positive (FP), true negative (TN), and false negative (FN) predictions.

Based on the results of the Multinomial Naive Bayes (MNB) classifier:

* The MNB classifier achieved an accuracy of 89.89%.
* The confusion matrix shows that the MNB classifier correctly predicted 0 instances as negative (true negatives) and 1325 instances as positive (true positives).
* However, it incorrectly predicted 147 instances as positive when they were actually negative (false positives), and it failed to identify any instances as negative (false negatives).

A graph showing a red line

Description automatically generated

Figure : Multinomial Naive Bayes ROC- AUC

**AUC-ROC:** “When we need to check or visualize the performance of the multi-class classification problem, we use the AUC (Area Under the Curve) ROC (Receiver Operating Characteristics) curve. It is one of the most important evaluation metrics for checking any classification model’s performance. It is also written as AUROC (Area Under the Receiver Operating Characteristics)”[[3]](#footnote-3)

* The AUC value indicates the overall performance of the MNB classifier in distinguishing between positive and negative instances based on the Receiver Operating Characteristic (ROC) curve.
* An AUC of 0.82 suggests that the MNB classifier has a reasonably good ability to differentiate between positive and negative instances.
* The closer the AUC value is to 1, the better the classifier is at correctly classifying positive instances as positive and negative instances as negative.

## ****Logistic Regression:****

**In Logistic Regression, the algorithm applies a logistic function (also known as the sigmoid function) to a linear combination of the input features. The sigmoid function maps the linear combination to a value between 0 and 1, representing the probability of the instance belonging to the positive class. The model then uses a threshold (typically 0.5) to classify instances into the positive class if the predicted probability is above the threshold, and into the negative class otherwise.**

A screenshot of a computer

Description automatically generated

Figure : Logistic Regression classifier result

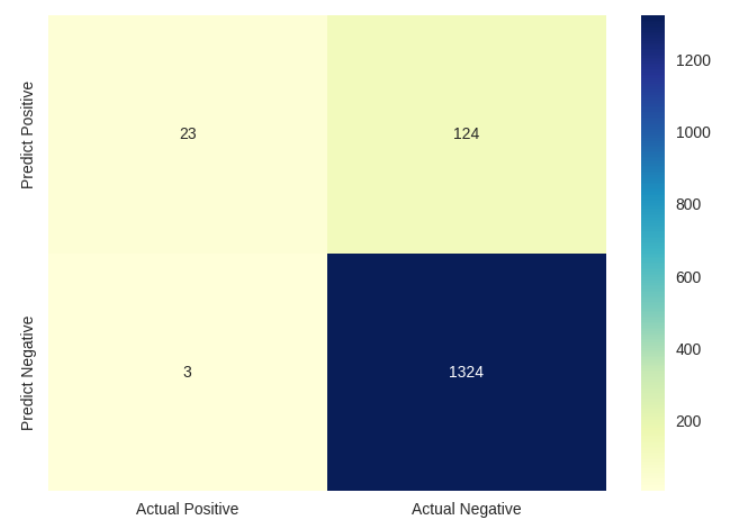


Figure : Logistic Regression heatmap

Based on the results for the Logistic Regression classifier, the performance of the model can be assessed as follows:

* The model correctly predicted 23 instances as negative (true negatives).
* The model incorrectly predicted 124 instances as positive when they were actually negative (false positives).
* The model incorrectly predicted 3 instances as negative when they were actually positive (false negatives).
* The model correctly predicted 1324 instances as positive (true positives).
* **Precision**: The precision for the negative class is 0.88, indicating that out of all instances predicted as negative, 88% were actually negative. The precision for the positive class is 0.91, indicating that out of all instances predicted as positive, 91% were actually positive.
* **Recall**: The recall (also known as sensitivity) for the negative class is 0.16, indicating that only 16% of the actual negative instances were correctly identified. The recall for the positive class is 1.00, indicating that all positive instances were correctly identified.
* **F1-score:** The F1-score is the harmonic mean of precision and recall. The F1-score for the negative class is 0.27, and for the positive class, it is 0.95.
* **Support:** It represents the number of instances in each class.
* **The accuracy of the Logistic Regression classifier** is reported as 91.38%, indicating the overall proportion of correctly predicted instances.

A graph showing a red line

Description automatically generated

Figure : ROC- AUC

an **AUC** of 0.95 for a Logistic Regression classifier suggests a very good model that can be reliably used for classification and prediction tasks.

## ****Support Vector Machines (SVM):****

**SVM is a popular machine learning algorithm for text classification. It aims to find the best hyperplane that separates positive and negative sentiment in the feature space. SVM can handle high-dimensional data and has been successfully applied to sentiment analysis.**

A screenshot of a computer screen

Description automatically generated

Figure : Support Vector Machines classifier results

A screenshot of a chart

Description automatically generated

Figure : Support Vector Machines heatmap

**From the confusion matrix and the heatmap, we can infer the following:**

* **The model correctly predicted 95 instances as negative (TN).**
* **The model incorrectly predicted 52 instances as positive when they were actually negative (FP).**
* **The model incorrectly predicted 49 instances as negative when they were actually positive (FN).**
* **The model correctly predicted 1278 instances as positive (TP).**

**Precision**: It measures the proportion of correctly predicted positive instances out of all instances predicted as positive. For the negative class, the precision is 0.66, and for the positive class, it is 0.96.

**Recall**: It measures the proportion of correctly predicted positive instances out of all actual positive instances. For the negative class, the recall is 0.65, and for the positive class, it is 0.96.

**F1-score**: It is the harmonic mean of precision and recall, providing a single metric that balances both measures. For the negative class, the F1-score is 0.65, and for the positive class, it is 0.96.

**Support**: It represents the number of instances in each class.

**The classification report:** includes the overall accuracy of the model, which is 93.15%.

A graph showing a red line

Description automatically generated

Figure : Support Vector Machines ROC- AUC

Area Under the Curve (AUC) is 0.93 which means that the model has a high ability to distinguish between positive and negative instances: AUC values closer to 1 indicate that the model has a strong ability to correctly classify positive instances as positive and negative instances as negative.

# Task 6 – Evaluation, Inferences, Recommendation and Reflection

## Evaluation

To evaluate and compare the performance of the three classifiers (SVM, Logistic Regression, and Multinomial Naive Bayes), we can use multiple metrics, including accuracy, precision, recall, F1-score, and the ROC curve. Let's examine and compare the results for each model using these metrics:

**SVM Classifier:**

* Accuracy: 93.15%
* Precision: positive class (0.92), negative class (0.95)
* Recall: positive class (0.99), negative class (0.16)
* F1-score: positive class (0.95), negative class (0.27)
* AUC: 0.93

**Logistic Regression Classifier:**

* Accuracy: 91.38%
* Precision: positive class (0.91), negative class (0.88)
* Recall: positive class (1.00), negative class (0.16)
* F1-score: positive class (0.95), negative class (0.27)
* AUC: 0.95

**MNB Classifier:**

* Accuracy: 89.89%
* Precision: positive class (0.90), negative class (0.00)
* Recall: positive class (1.00), negative class (0.00)
* F1-score: positive class (0.95), negative class (0.00)
* AUC: 0.82

## Inferences:

* SVM excelled in accuracy, precision, and AUC.
* Logistic Regression performed well for positive sentiment but struggled with negativity.
* MNB achieved perfect recall for positive sentiment but completely missed negative instances.

## Reflections:

* SVM consistently performed well, making it reliable for sentiment analysis.
* Logistic Regression needs improvement in identifying negative sentiment.
* MNB's failure to recognize negativity makes it unsuitable for this task.

## Possible improvements:

* Address class imbalance: Balance the dataset to improve negative sentiment identification.
* Feature engineering: Improve feature extraction to capture sentiment nuances better.
* Model selection and tuning: Explore other models and optimize hyperparameters.
* Cross-validation: Ensure robust performance and identify overfitting.

GIThub link: <https://github.com/SharifaAli/NLP-Project.git>

1. Dataset source: https://www.kaggle.com/datasets/tarkkaanko/amazon/data [↑](#footnote-ref-1)
2. https://scikit-learn.org/stable/modules/generated/sklearn.naive\_bayes.MultinomialNB.html#:~:text=The%20multinomial%20Naive%20Bayes%20classifier,more%20in%20the%20User%20Guide. [↑](#footnote-ref-2)
3. https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5 [↑](#footnote-ref-3)