

**MULTIMEDIA UNIVERSITY OF KENYA**

**FACULTY OF COMPUTING & INFORMATION TECHNOLOGY**

**CUSTOMER FEEDBACK SENTIMENT ANALYSIS**

**BY**

**OKIBO SHADRACK ONYIEGO**

**REG NO: CIT-222-073/2018**

**SUPERVISOR: LECTURER KINUTHIA**

**JULY 2022**

Submitted in partial fulfillment of the requirements of Fourth Year Bachelor of Science in Computer Technology.

# 

# **DECLARATION**

I hereby declare that this Project is my own work and has, to the best of my knowledge, not been submitted to any other institution of higher learning.

Student: ..................................................... Reg No: ..................................................

Signature: ............................................... Date: .....................................................

This project has been submitted as a partial fulfillment of requirements for the Bachelor of Science in Computer Technology of Multimedia University of Kenya with my approval as the University supervisor.

Supervisor: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Signature: ..................................................... Date: ..................................................

# **ACKNOWLEDGEMENTS**

I would like to express my greatest appreciation to everyone who has helped and supported me throughout the project. I am thankful to my supervisor for his ongoing support during the project, from initial advice on my proposal, and encouragement, which led to the final report of this project.

A special acknowledgement goes to my classmates, who helped me in completing the project by exchanging interesting ideas and sharing their experience.

I wish to thank my parents as well for their undivided support and interest who inspired me and encouraged me to go my own way, without whom I would be unable to complete my project.

I would also want to thank my close friends like Joshua Nyandwaro and Haniel Okello who displayed appreciation to my work and motivated me to continue my work.

# **ABSTRACT**

At the end of the day, businesses can grow only when they truly understand the people using their products or services. This in itself is a huge task as the human experience comes with a wide range of complicated emotions and interactions.

Knowing your customer and understanding them, is very important for building and maintaining a positive brand perception. The contextual analysis of identifying information helps businesses understand their customers’ social sentiment by monitoring online conversations.

Sentiment analysis is mainly concerned with the identification and classification of opinions or emotions of each feedback. Uber Kenya can use sentiment analysis to understand the subjective reasons why customers are or are not responding to something, whether the user experience, or customer support.

The experience of the customers can either be positive, negative or neutral. And one of the most effective ways to measure marketing and branding campaigns is to analyze consumer sentiment around them.

The insights gained from sentiment analysis can help Uber Kenya bring accurate changes and transformation of their transport business. It can be in areas that are either creating the most negative sentiment features, such as transportation price, return policies or customer service. Overall, these strategic measures will help the Uber Kenya:

* Become more competitive
* Attract new customers
* Retain present customers
* Make customers more comfortable
* Improve marketing messages and campaigns

# **TABLE OF CONTENTS**

Contents

[**DECLARATION** ii](#_Toc110295333)

[**ACKNOWLEDGEMENTS** iii](#_Toc110295334)

[**ABSTRACT** iv](#_Toc110295335)

[**TABLE OF CONTENTS** v](#_Toc110295336)

[**LIST OF ABBREVIATIONS** vi](#_Toc110295337)

[**LIST OF FIGURES** vi](#_Toc110295338)

[**CHAPTER 1 – INTRODUCTION** 1](#_Toc110295340)

[**1.1** **Background of study** 1](#_Toc110295341)

[**1.2** **Problem statement** 1](#_Toc110295342)

[**1.2.1** **Proposal/Solution** 1](#_Toc110295343)

[**1.3** **Aim of the study** 1](#_Toc110295344)

[**1.3.1.** **Research objectives** 2](#_Toc110295345)

[**1.4** **Significance/Justification of the study** 2](#_Toc110295346)

[**1.5** **Scope (i.e., defines the system boundary)** 2](#_Toc110295347)

[**1.6** **Assumptions** 2](#_Toc110295348)

[**1.7** **Limitations (i.e., challenges and counter measures)** 2](#_Toc110295349)

[**CHAPTER 2 – LITERATURE REVIEW** 3](#_Toc110295350)

[**2.1 Introduction** 3](#_Toc110295351)

[**2.2 Related systems (2-3)** 3](#_Toc110295352)

[**2.3 Limitations/weaknesses of these systems** 6](#_Toc110295353)

[**2.4 How the proposed solution will handle these weaknesses.** 7](#_Toc110295354)

[**CHAPTER 3 – METHODOLOGY** 8](#_Toc110295355)

[**3.1 Introduction** 8](#_Toc110295356)

[**3.2 Methodology (stating the methodology, its description and justification of using this methodology.)** 8](#_Toc110295357)

[**3.3 Data collection methods and tools/Data sets** 13](#_Toc110295358)

[**CHAPTER 4 – SYSTEM ANALYSIS** 14](#_Toc110295359)

[4.1 Detailed analysis of current system using flow charts, DFDs, UML, Context diagrams, etc. 14](#_Toc110295360)

[4.2 System requirements 15](#_Toc110295361)

[**4.2.1 Functional requirements** 15](#_Toc110295362)

[**4.2.2 Nonfunctional requirements** 16](#_Toc110295363)

[**CHAPTER 5 SYSTEM DESIGN** 17](#_Toc110295364)

[5.1 Architectural design 17](#_Toc110295365)

[5.2 Database design 17](#_Toc110295366)

[5.3 User interface design 18](#_Toc110295367)

[**CHAPTER 6 IMPLEMENTATION AND TESTING** 21](#_Toc110295368)

[6.1 Development environment 21](#_Toc110295369)

[6.2 Test Plan (test data, test cases, test results) 27](#_Toc110295435)

[**CHAPTER 7: RESULTS & CONCLUSION** 29](#_Toc110295443)

[7.1 Achievements and lessons learnt 29](#_Toc110295444)

[7.2 Conclusions 29](#_Toc110295445)

[7.3 Recommendations 29](#_Toc110295452)

[**REFERENCES** 30](#_Toc110295453)

# **LIST OF ABBREVIATIONS**

BOW – Bag of words

NLP – Natural Language Processing

SA - Sentiment analysis

NPS - Net Promoter Score

BCS - Brand Consideration (BC) Score

CES - Customer Effort Score (CES) Survey Systems

CRS - Customer Review Score (CRS)

CSAT - Customer Satisfaction (CSAT) Survey Systems

# **LIST OF FIGURES**

[Figure 1: Net Promoter Score 3](#_Toc110296681)

[Figure 2: Customer Satisfaction Survey Systems 5](#_Toc110296682)

[Figure 3: Customer Effort Score 6](#_Toc110296683)

[Figure 4: Sentiment Analysis 8](#_Toc110296684)

[Figure 5: Stemming 9](#_Toc110296685)

[Figure 6: optimal separating hyperplane between two classes 11](#_Toc110296686)

[Figure 7: Logistic Regression 12](#_Toc110296687)

[Figure 8: Model Accuracy 12](#_Toc110296688)

[Figure 9: Methodology Flowchart 14](#_Toc110296689)

[Figure 10: System Flowchart 14](#_Toc110296690)

[Figure 11: Architectural Design 17](#_Toc110296691)

[Figure 12: Database Design 17](#_Toc110296692)

# 

# **CHAPTER 1 – INTRODUCTION**

## **Background of study**

A business breathes on the gratification of its customers. The experience of the customers can either be positive, negative or neutral. One of the most effective ways to measure marketing and branding campaigns is to analyze consumer sentiment around them.

As most of the people have networked themselves through social websites, they use them to express their sentiments through these websites. These sentiments have proved fruitful to an individual, business, government for making decisions. The impressions posted on different available sources can be used by a business to know the market mood about the services they are providing. Though, analyzing huge moods expressed with different features, style have raised challenge for users.

## **Problem statement**

Almost every industry, company or organization today is going through some form of digital transformation that results in greater and greater quantities of both structured and unstructured data. The biggest challenge that companies have is turning their unusable, unstructured data into useful insights that can help them make data driven decisions, create operational efficiencies, value improvement, and overall competitive advantage.

Nowadays, businesses are increasingly using the content in social media for decision making. If an individual wants to buy a consumer product, he or she is no longer limited to asking his or her friends and family for opinions because there are many user reviews and discussions in public forums on the web about the product or business.

Businesses can grow only when they truly understand the people using their products or services. This in itself is a momentous task as the human experience comes with a wide range of complicated emotions and interactions.

### **Proposal/Solution**

My proposal is to use sentiment analysis using machine learning as a tool for monitoring and understanding client sentiments as they share their opinions and emotions more openly. Uber Kenya can know what makes customers satisfied or frustrated by automatically evaluating customer feedback, such as comments and dialogues on social media especially on Twitter. This will allow them to customize their transportation services and prices to match their customers' demands.

## **Aim of the study**

The aim of this study is to build a binary classification sentiment analysis model using machine learning that can be able to classify whether customer reviews on Uber Kenya services are positive or negative.

This will enable Uber Kenya to have automated insights on customers feedback so as to have rich information that eliminates guesswork and to enhance execution of timely decisions.

### **Research objectives**

* Carry out text data collection from twitter
* Perform text pre-processing on the dataset created
* Build a binary classification model for sentiment classification
* Build a user interface for carrying out the sentiment analysis
* Analyze the positive tweets from the negative tweets

## **Significance/Justification of the study**

This will help Uber Kenya in understanding their customers better. They can serve their customers more efficiently when they know where they lag and where they excel.

## **Scope (i.e., defines the system boundary)**

The project aims to enable Uber Kenya be able to:

* Collect data from twitter and build their own dataset
* Perform text processing on the dataset
* Build their classification model
* Be able to leverage the power of this data and get the deepest insights to help them:
* Formulate business strategies
* Exceed customer expectations
* Generate leads
* Build marketing campaigns
* Get to list issues customers are majorly concerned with
* Generate an aggregated score of the overall feedback
* They will also be able to also take on reviews for their competitors and even get to have insights in their competitors reviews and see where their strengths are.

## **Assumptions**

* The review language is English
* Every customer feedback has an angle, so no neutral feedbacks considered
* The data to be collected (feedback, comments, reviews) from customers and product users was one posted by the them willingly.

## **1.7 Limitations (i.e., challenges and counter measures)**

* Due to time constraints, I could not be able to review Uber Kenya competitors reviews and have insights on their strengths in the market.
* Internet connectivity is needed to fetch data from twitter
* Due to time constraints, I wasn’t able to handle double negation of statements

# **CHAPTER 2 – LITERATURE REVIEW**

## **2.1 Introduction**

The era of getting valuable insights from surveys and social media has peaked due to the advancement of technology. Therefore, Uber Kenya can be in touch with the pulse of what their customers are feeling. They can use intelligent classifiers like sentiment analysis to leverage the power of data and get the deepest insights, formulate business strategies, exceed customer expectations, generate leads, build marketing campaigns, and open up new avenues for growth through natural language processing solutions.

Using sentimental analysis, Uber Kenya can even get to access sentiment data of their competitors which can give them the opportunity as well as the incentive to perk up their performance. Sentiment analysis can be very helpful in predicting the customer trends. Once they get acquainted with the current customer trends, strategies can easily be developed to capitalize on them. And eventually, gain a leading edge in the competition.

## **2.2 Related systems (2-3)**

***The NPS (Net Promoter Score) Approach***

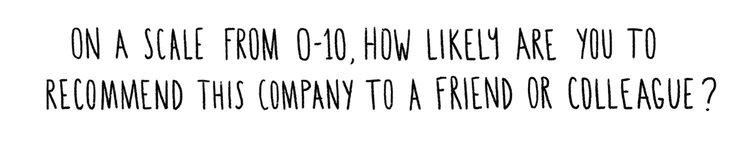


Figure 1: Net Promoter Score

Net Promoter Score (NPS) is a well-accepted measurement of customer satisfaction in most customer-facing industries. NPS surveys help you gain feedback for a business with the simple question: Will you recommend this brand, product, or service to your friend or family? The output is a single score on the number scale. Businesses use these sentiment scores to analyze the customer as promoters, detractors, and passives.

Here the goal is to find the overall customer experience and elevate your customer to promoter level. Theoretically, include the phases as: will buy more, stay longer and refer to another customer.

***Brand Consideration (BC) Score***

BC is used to monitor customer satisfaction. Brand Consideration measures the likelihood that a customer will purchase a particular brand based on their predisposition or feelings towards that brand. Both NPS and BC are primarily researched using customer surveys.

***Lexicon-Based Approach***

Lexicon based method uses sentiment dictionary with opinion words and match them with the data to determine polarity. They assign sentiment scores to the opinion words describing how Positive, Negative and Objective the words contained in the dictionary are.

Lexicon-based approaches mainly rely on a sentiment lexicon, i.e., a collection of known and precompiled sentiment terms, phrases and even idioms, developed for traditional genres of communication, such as the Opinion Finder lexicon;

***Dictionary-based Approach***

Dictionary based approach suggests you’d implement judging of sentiment based on presence of signaling sentiment words (and perhaps some shorter context, like negations in front of them) + some sort of counting mechanism to arrive at sentiment prediction. In research literature dictionary-based approach is usually referred to as the simplest (and hence of less accuracy) one.

It is based on the usage of terms (seeds) that are usually collected and annotated manually. This set grows by searching the synonyms and antonyms of a dictionary. An example of that dictionary is WordNet, which is used to develop a thesaurus called Sent WordNet.

***Corpus-Based Approach***

The corpus-based approach has objective of providing dictionaries related to a specific domain. These dictionaries are generated from a set of seed opinion terms that grows through the search of related words by means of the use of either statistical or semantic techniques.

* Methods based on statistics: Latent Semantic Analysis (LSA).
* Methods based on semantic such as the use of synonyms and antonyms or relationships from thesaurus like WordNet may also represent an interesting solution.

Corpus based suggests data-driven approach where you will have access not only to sentiment labels, but to a context which you can use to your advantage in an ML algorithm. Corpus also carries some domain specificity, that can inform your algorithm of sentiment label variety for a word depending on its context / domain.

***Rule Based Approach***

This extracts opinions on some aspects of an entity, so this approach is very close to natural language. The rules are based on contextual patterns, which capture the various properties of words and their relationships in the text. In product reviews, a grammatical relationship between aspects and opinions can be utilized using the extraction rule

This approach relies on manually crafted rules for data classification to determine sentiment. This approach uses dictionaries of words with positive or negative values to denote their polarity and sentiment strength to calculate a score. Additional functionality can also be added by including expressions. Rule based sentiment analysis algorithms can be customized based on context by developing even smarter rules.

*How it works:* It counts the number of positive and negative words in the given text. If the number of positives is more than the negatives, it will return a positive sentiment. If both are equal, it will return a neutral sentiment.

***Customer Review Score (CRS)***

For example, there are cases where thousands of customers are writing the reviews about a single brand, and there is no way the business owner can go through all of them to find about the overall good things and bad things about his/her business without investing a lot of time and resources. The NPS only uses one metric, but there is a huge information already stored in these reviews which are richer in content, and can tell you exactly why the customer is happy and why the customer is unhappy – Something the NPS can’t tell but the CRS can.

The problem with the reviews is the difficulty in analyzing them and extracting information, understanding customer emotion and the polarity of the review (positive/negative). This is something which only a human mind can do, and is not a job for machines.

With the advancement in data sciences and Natural Language Processing, this is now possible to some extent. Of course, the computer cannot find out the difference between a happy “I am Fine” and a sarcastic “I am Fine”, but it can be 75%-80% accurate in predicting the sentiment in a sentence or a paragraph as compared to Human Brain which does it at 100% accuracy. This is good enough for a predictor model, and skimming through thousands of documents/reviews in a matter of seconds.

***Customer Satisfaction (CSAT) Survey Systems***

Customer Satisfaction surveys usually contain a simple question with a binary response (e.g., yes/no, happy face/sad face). They ask things like “Did our product do what you wanted it to do?”. These scores are usually high (in the 98%+ range), so a sudden spike in negative scores tells you there’s an issue that needs immediate attention.

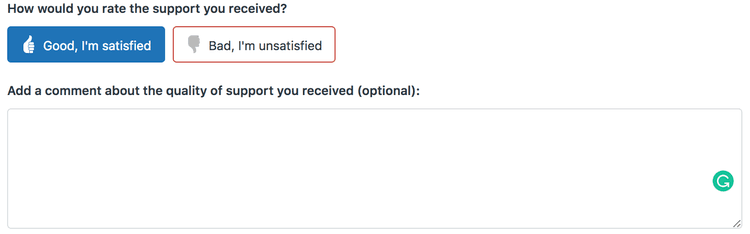


Figure 2: Customer Satisfaction Survey Systems

Customer satisfaction surveys are brief polls designed to measure customer satisfaction with a product or service in a single specific moment, usually by ranking their satisfaction or through a binary choice (happy face/sad face). CSAT scores are usually high, so a sudden negative dip is an indicator that something has gone wrong; they’re useful for pinpointing specific problem areas or issues within your website.

***Customer Effort Score (CES) Survey Systems***

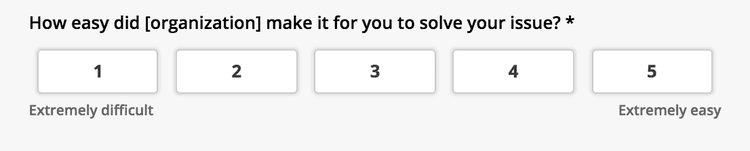


Figure 3: Customer Effort Score

Customer Effort Score (CES) is a measurement of how much effort a customer felt they had to put into an action or interaction with your business. Customers who find your product easy to use are more likely to continue using it and report overall satisfaction.

## **2.3 Limitations/weaknesses of these systems**

***The NPS (Net Promoter Score) Approach*** - Is based on asking the customer of their opinion and this may make a customer not talk out his/her whole opinion for the sake of being kind and empathetic. This now will yield inaccurate results in comparison to feedback written by a product user/customer on an online forum out of their own will.

NPS has two major drawbacks:

* NPS is sparse, given only a fraction of users respond to the survey
* NPS is slow. It takes at least a week for results to show up.

***Rule Based Approach*** *-* The simplicity of rules-based sentiment analysis makes it a good option for basic document-level sentiment scoring of predictable text documents, such as limited-scope survey responses. However, a purely rules-based sentiment analysis system has many drawbacks that negate most of these advantages. A rules-based system must contain a rule for every word combination in its sentiment library. Creating and maintaining these rules requires tedious manual labor. And in the end, strict rules can’t hope to keep up with the evolution of natural human language. Instant messaging has butchered the traditional rules of grammar, and no ruleset can account for every abbreviation, acronym, double-meaning and misspelling that may appear in any given text document.

In addition, a rules-based system that fails to consider negators and intensifiers is inherently naïve. Out of context, a document-level sentiment score can lead you to draw false conclusions.

***Lexicon Based Approach*** - This approach counts the number of positive and negative words in the given text but does not take into account how the words are combined in a sentence, it only looks at occurrences.

It is quick to implement but the model involves a long-term cost outlay as it requires regular maintenance so that you get consistent and improved results.

Drawback: Can’t deal with domain and context specific orientations

***Customer Review Score (CRS)*** - CRS not used widely. The answer is because of the difficulty to go through thousands of text-based reviews. You need to have a good amount of data to analyze, and it should be generalized (not collected from a single source).

## **2.4 How the proposed solution will handle these weaknesses.**

Sentiment Analysis using Machine Learning Approach

Instead of clearly defined rules, this sentiment analysis model will use machine learning to figure out the essence of the statement. This ensures that the exactitude of the analysis improves and information can be processed on many criteria without it being too complicated.

This approach involves the use of machine learning algorithms under supervision. An algorithm is trained with many sample passages until it can predict with accuracy the sentiment of the text. Then large pieces of text are fed into the classifier and it predicts the sentiment as negative or positive.

# **CHAPTER 3 – METHODOLOGY**

## **3.1 Introduction**

Sentiment Analysis is the NLP technique that is performed on text to determine whether the author’s intentions towards a particular topic, product, etc. are positive or negative.

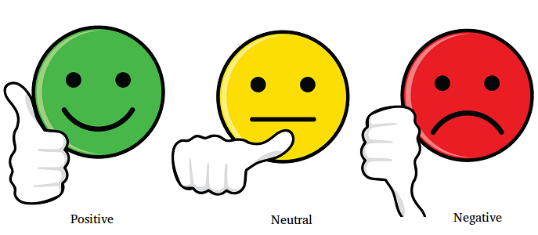


Figure 4: Sentiment Analysis

## **3.2 Methodology (stating the methodology, its description and justification of using this methodology.)**

***Step 1: Importing the dataset***

Import dataset ready for data preprocessing steps.

***Step 2: Exploring and Processing Text Data***

1. *Converting Text Data to Lowercase:*

The simplest way to do this is by using the default *lower()* function in Python. The *lower()* method converts all uppercase characters in a string into lowercase characters and returns them. For example, it is useless to have some words in different cases (e.g., ‘good’ and ‘GOOD’).

1. *Removing Punctuation and numbers:*

Punctuations and numbers don’t help much in processing the given text, if included, they will just increase the size of a bag of words and decrease the efficiency of an algorithm. The simplest way to do this is by using the regex and *replace()* function in Python.

1. *Removing Stop Words:*

Stop words are very common words that carry no meaning or less meaning compared to other keywords. If we remove the words that are less commonly used, we can focus on the important keywords instead.

The simplest way to do this by using the NLTK library, or building your own stop words file.

1. *Correcting Spelling:*

Most of the text data is in the form of either customer reviews, blogs, or tweets, where there is a high chance of people using short words and making typo errors. This will help us in reducing multiple copies of words, which represents the same meaning.

The simplest way to do this by using the TextBlob library.

1. *Stemming*

Stemming is a process of extracting a root word. Stemming is desirable as it may reduce redundancy as most of the time the word stem and their inflected/derived words mean the same.

Taking roots of the word

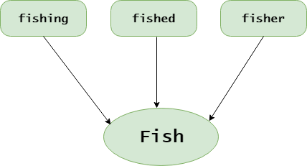


Figure 5: Stemming

Errors in Stemming: There are mainly two errors in stemming – Over stemming and Under stemming. Over stemming occurs when two words are stemmed to same root that are of different stems. Under-stemming occurs when two words are stemmed to same root that are not of different stems. The simplest way to do this by using NLTK or a TextBlob library.

1. *Lemmatizing Text:*

Lemmatization is a process of extracting a root word by considering the vocabulary. For example, “good,” “better,” or “best” is lemmatized into good.

Lemmatization can get better results.

* + The stemmed form of leafs is leaf.
  + The stemmed form of leaves is leav.
  + The lemmatized form of leafs is leaf.
  + The lemmatized form of leaves is leaf.

The simplest way to do this is by using NLTK or the TextBlob library.

1. *Build Wordcloud*

Wordcloud is the pictorial representation of the most frequently repeated words representing the size of the word.

***Step 3: Converting Text to Features Using TF-IDF***

The whole idea of having TF-IDF is to reflect on how important a word is to a document in a collection, and hence normalizing words appeared frequently in all the documents

**Term frequency (TF):** Term frequency is simply the ratio of the count of a word present in a sentence, to the length of the sentence. TF is basically capturing the importance of the word irrespective of the length of the document.

**Inverse Document Frequency (IDF):** IDF of each word is the log of the ratio of the total number of rows to the number of rows in a particular document in which that word is present. IDF = log(N/n), where N is the total number of rows and n is the number of rows in which the word was present. IDF will measure the rareness of a term. Words like “a,” and “the” show up in all the documents of the corpus, but rare words will not be there in all the documents. So, if a word is appearing in almost all documents, then that word is of no use to us since it is not helping to classify or in information retrieval. IDF will nullify this problem.

TF-IDF is the simple product of TF and IDF so that both of the drawbacks are addressed, which makes predictions and information retrieval relevant.

***Step 4:*** ***Splitting data into Training and Test set.***

Here we split the data into two; training data and testing data.

For this, we need class *train\_test\_split* from *sklearn.* Split can be made 80/20, via “*test\_size*”.

***Step 5:*** ***Fitting a Binary Classification Model***

*Machine learning models can be of two kinds:*

*a. Traditional Models* – This method requires the gathering of a dataset with examples for positive, negative, and neutral classes, then processing this data, and finally training the algorithm based on the examples. These methods are mainly used for determining the polarity of text. Traditional machine learning methods such as Naïve Bayes, Logistic Regression and Support Vector Machines (SVM) are widely used for large-scale sentiment analysis because they are capable of scalability.

*b. Deep Learning Models* – This provides more precise results than traditional models and includes neural network models such as CNN (Convoluted Neural Network), RNN (Recurrent Neural Network), and DNN (Deep Neural Network).

***Naive Bayes sentiment analysis***

It is called ‘Naïve’ because it uses the assumption that the occurrence of one feature is independent of other features. For instance, it identifies the orange fruit based on color, shape and taste with each feature independently being assessed to arrive at the conclusion. The ‘Bayes’ is because it is based on the principle of the Bayes theorem.

The Bayes theorem relies on the concept of conditional probability or the probability that event A occurs when event B occurs. The theorem in effect states that the probability of A if B is true = the probability of B if A is true, multiplied by the times the probability of A being true and the whole divided by the probability of B being true.

***Support vector machine (SVM)***

SVM is a supervised(feed-me) machine learning algorithm that can be used for both classification or regression challenges. Classification is predicting a label/group and Regression is predicting a continuous value. SVM performs classification by finding the hyper-plane that differentiate the classes we plotted in n-dimensional space.

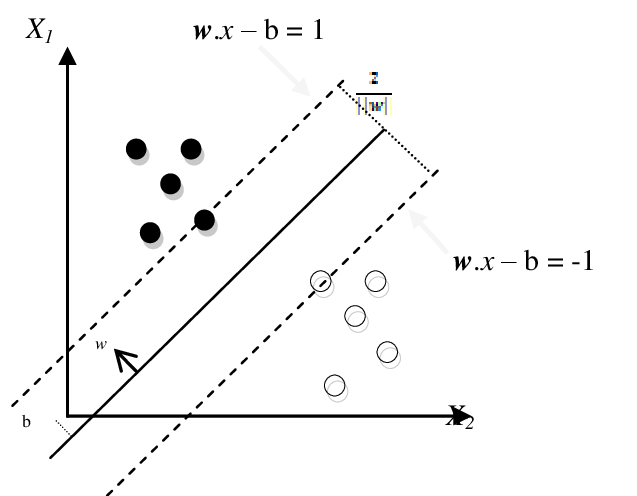


Figure 6: optimal separating hyperplane between two classes

SVM draws that hyperplane by transforming our data with the help of mathematical functions called “Kernels”. Types of Kernels are linear, sigmoid, RBF, non-linear, polynomial, etc.,

SVM is useful for a few reasons:

* SVMs do not penalize correctly-labeled examples, which is often useful for generalization.
* Gives sparse solutions when using the kernel trick (nice for scalability)

***Logistic Regression***

Logistic Regression is a classification that serves to solve the binary classification problem. The result is usually defined as 0 or 1 in the models with a double situation. Logistic Regression is a classification model that is very easy to implement and performs very well on linearly separable classes. Logistic regression is a good model because it trains quickly even on large datasets and provides very robust results.

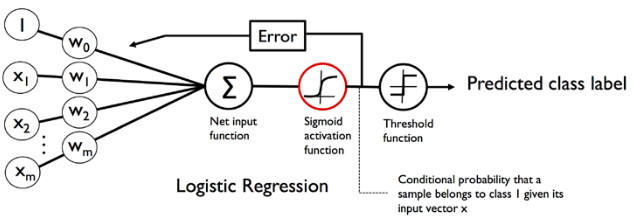


Figure 7: Logistic Regression

Logistic regression is useful for a few reasons:

* It outputs well-calibrated class probabilities
* It has an unconstrained, smooth loss function (compared to SVM)
* It plays well with Bayesian methods

***Step 6: Training of the model***

The aim of training the model is to figure out the best weights for our linear. In machine learning, we compute the optimal weights by optimizing the cost function.

***Step 7: Testing sentiment analysis model***

To test our model, we would run a subset of our data, known as the validation set, on the model to get predictions and compare it with the true label to calculate the model accuracy.

we can compute the accuracy of the model over the validation set. The accuracy is the number of times the model prediction matches with the true labels over the number of labels in the validation set. This metric gives an estimate of the times your logistic regression will correctly work on unseen data.

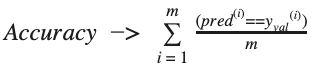


Figure 8: Model Accuracy

***Step 8: Model Evaluation***

After training the model we then apply the evaluation measures to check how the model is performing.

There are two main metrics for evaluating how well our model functions after we’ve trained it:

* *Accuracy -* Represents the percentage of correctly classified samples.
* *ROC AUC* - Area Under the Receiver Operating Characteristic Curve (ROC AUC) describes the relationship between the true positive rate (TRP) - that is, the ratio of samples that we correctly predicted belonging to the correct class - versus the false positive rate (FPR) - that is, the ratio of samples for which we incorrectly predicted their class membership. ROC AUC is preferable to accuracy, especially in multiclass prediction settings or when we have a class imbalance problem.

***Step 9:*** ***Predicting Final Results***

Finally, we run some tests:

*To know the accuracy, a confusion matrix is needed.*

Confusion Matrix is a 2X2 Matrix.

* TRUE POSITIVE: measures the proportion of actual positives that are correctly identified.
* TRUE NEGATIVE: measures the proportion of actual positives that are not correctly identified.
* FALSE POSITIVE: measures the proportion of actual negatives that are correctly identified.
* FALSE NEGATIVE: measures the proportion of actual negatives that are not correctly identified.

## **3.3 Data collection methods and tools/Data sets**

Text data collection from twitter using Snscrape

Snscrape is a python library that can be used to scrape tweets through Twitter’s API without any restrictions or request limits. Snscrape allows one to scrape basic information such as a user’s profile, tweet content and source.

Snscrape is not limited to twitter, but can also scrape data from other prominent social media networks like Facebook, Instagram and other sites.

Its advantages are that there are no limits to the number of tweets you can retrieve. So Snscrape allows you to get old data.

# **CHAPTER 4 – SYSTEM ANALYSIS**

4.1 Detailed analysis of current system using flow charts, DFDs, UML, Context diagrams, etc.

The Methodology Process

**The Methodology Process**

Text Cleaning or Preprocessing

Importing the dataset

Converting Text to Features Using TF-IDF

Exploring Text Data

Splitting into Training and Test set.

Fitting a Classification Model

Predicting Final Results

Business insights

Figure 9: Methodology Flowchart

How the system works flowchart

View Generated Classified Reviews

Click on Sentiment Analysis for the system to perform Sentiment analysis

Type in the Company’s Twitter Data to collect

Figure 10: System Flowchart

4.2 System requirements

**Software requirements**

* Jupyter notebook/Google collab
* Visual studio code
* Python 3.10.0
* NLP python modules
  + NLTK

Natural language toolkit and commonly called the mother of all NLP libraries. It is one of the mature primary resources when it comes to Python and NLP.

* + SpaCy

SpaCy is recently a trending library, as it comes with the added flavors of a deep learning framework. While SpaCy doesn’t cover all of the NLP functionalities, the things that it does do, it does really well.

* + TextBlob

This is one of the data scientist’s favorite libraries when it comes to implementing NLP tasks. It is based on both NLTK and Pattern. However, TextBlob certainly isn’t the fastest or most complete library.

* + CoreNLP

It is a Python wrapper for Stanford CoreNLP. The toolkit provides very robust, accurate, and optimized techniques for tagging, parsing, and analyzing text in various languages.

* + Scikit-learn

Scikit-learn is a free software machine learning library for the Python programming language. Scikit-learn is largely written in Python, with some core algorithms written in Cython to achieve performance.

* Web browser
* Operating System: Linux/windows

**Hardware Requirements**

Desktop/ Laptop with i3 ++ processor, 4 GB ++ RAM

### **4.2.1 Functional requirements**

**Web Application**

*Front-end*

* Admin Login
* Dashboard
* Sentiment Page

*Back-end*

* Database holding all data

### **4.2.2 Nonfunctional requirements**

*Reliability requirements*

The system must perform accurately towards the administrator request.

*Usability requirements*

The system should be user friendly and easy to use by the admin in doing the sentiment analysis for the data

*Implementation requirements*

The user interface is designed by Flask framework for a smooth user-friendly user interface.

*Security requirements*

Admin credentials should be encrypted so as to ensure confidentiality, integrity to avid entry of

# **CHAPTER 5 SYSTEM DESIGN**

5.1 Architectural design

The following Figure shows the entire proposed system architecture.

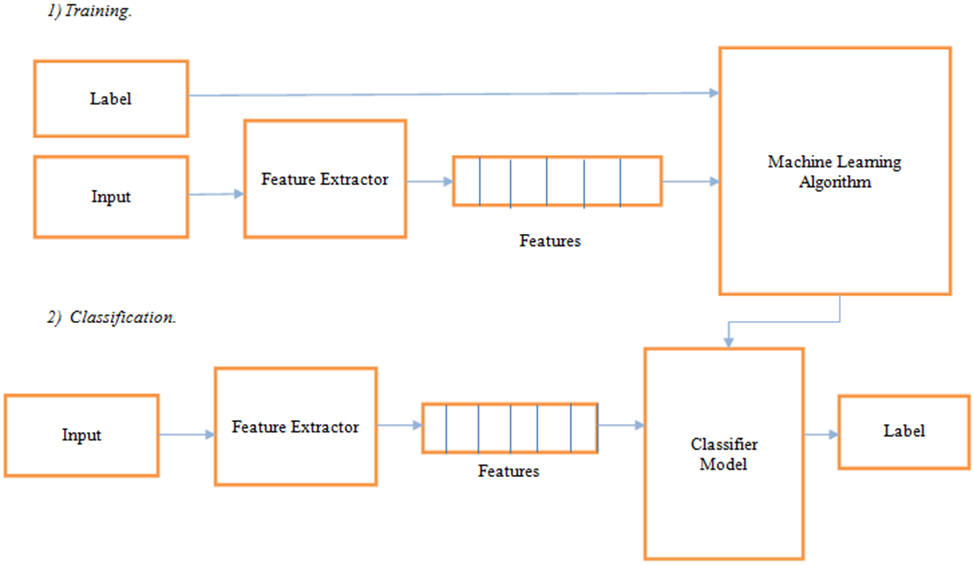


Figure 11: Architectural Design

5.2 Database design

My MySQL Database

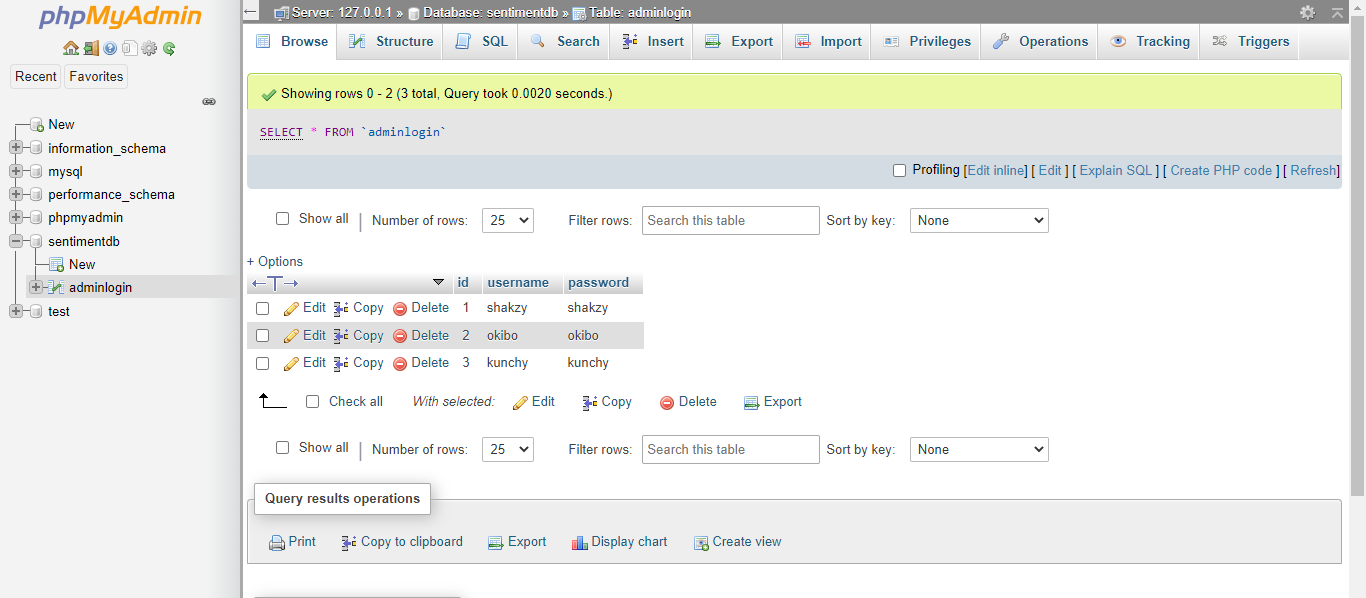
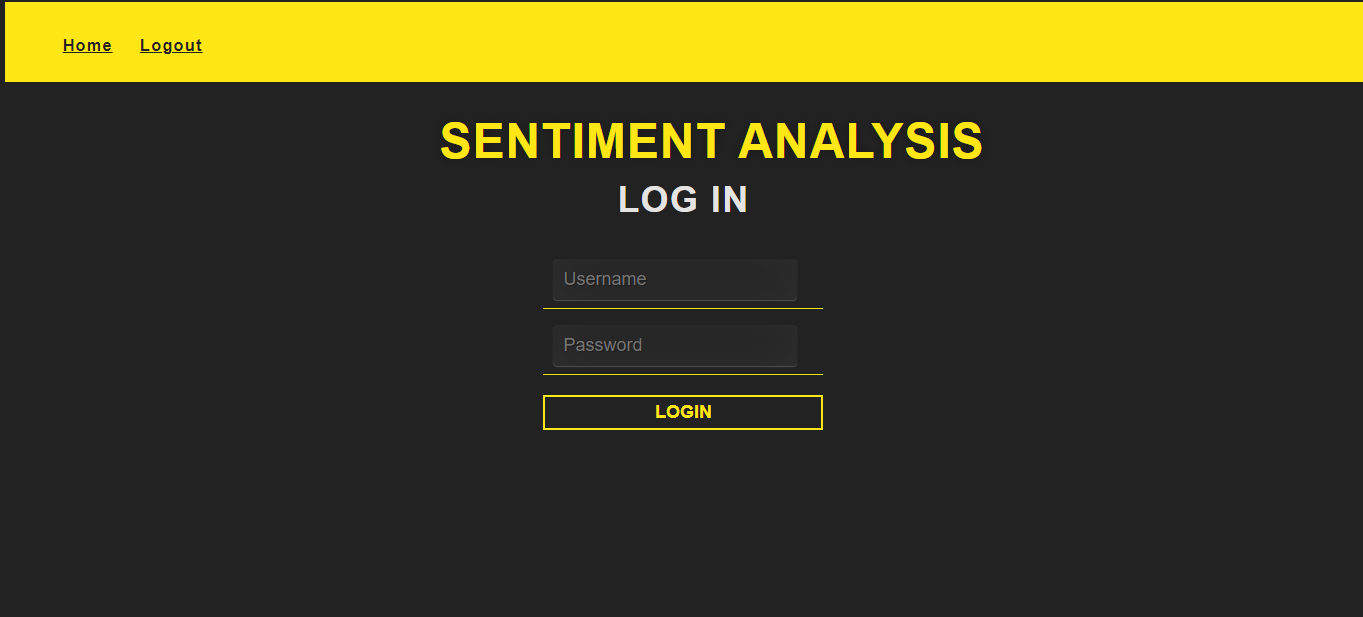


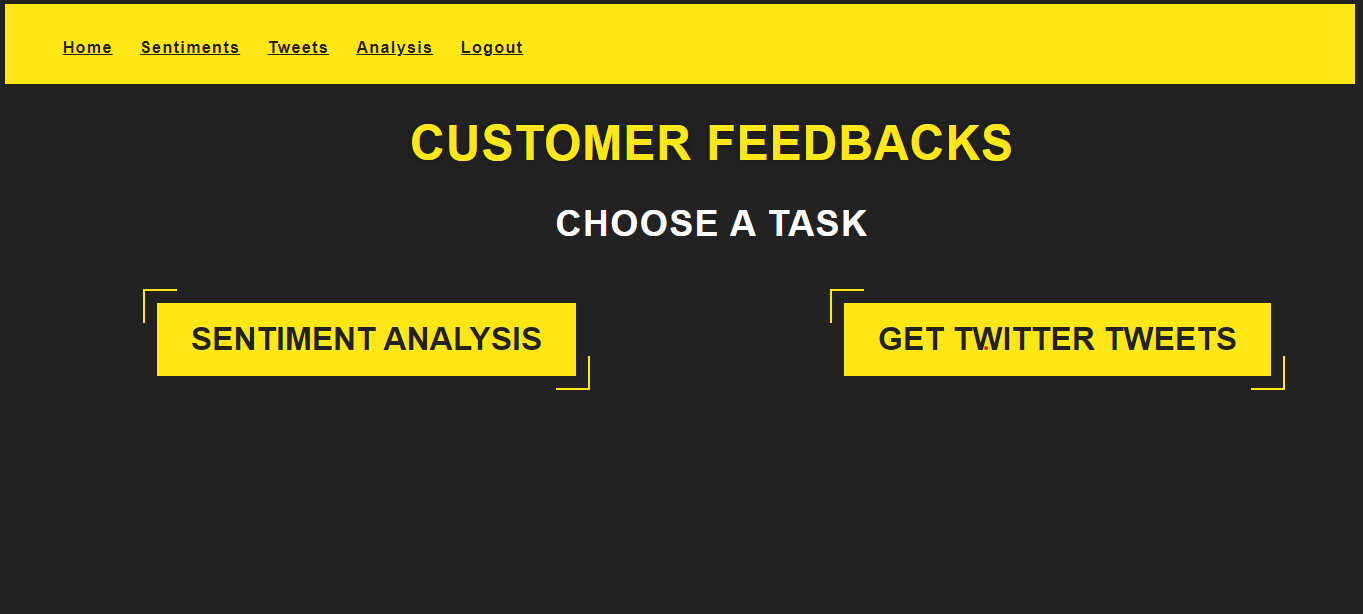
Figure 12: Database Design

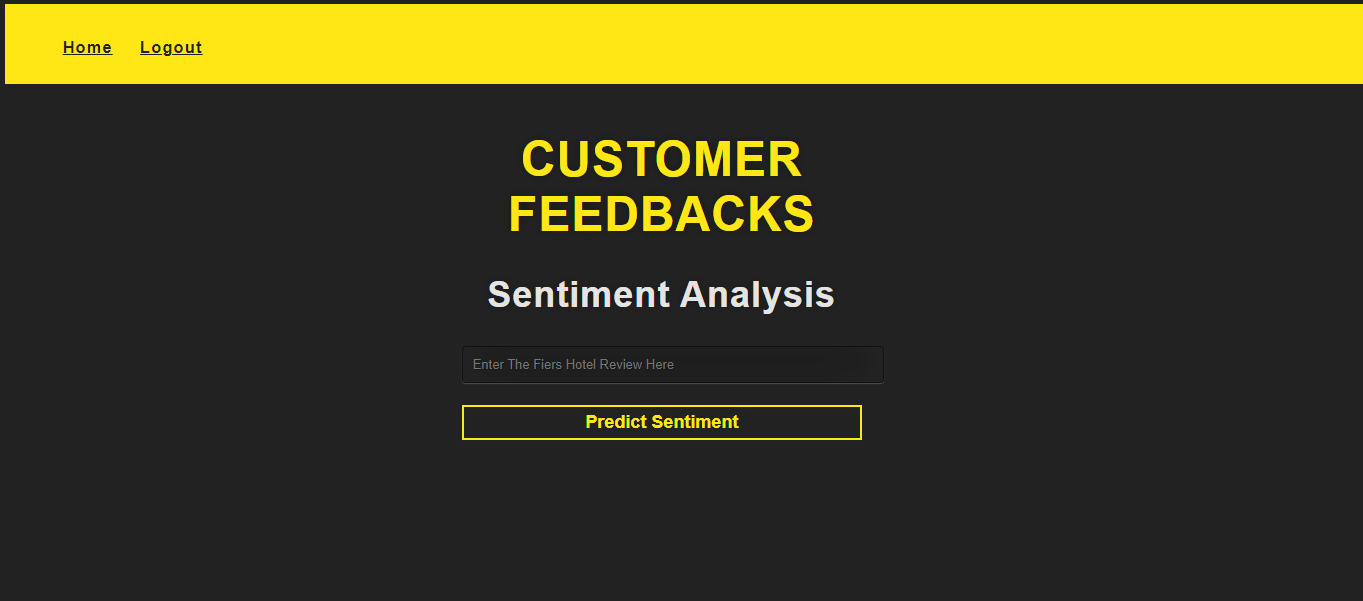
5.3 User interface design

The Login Page

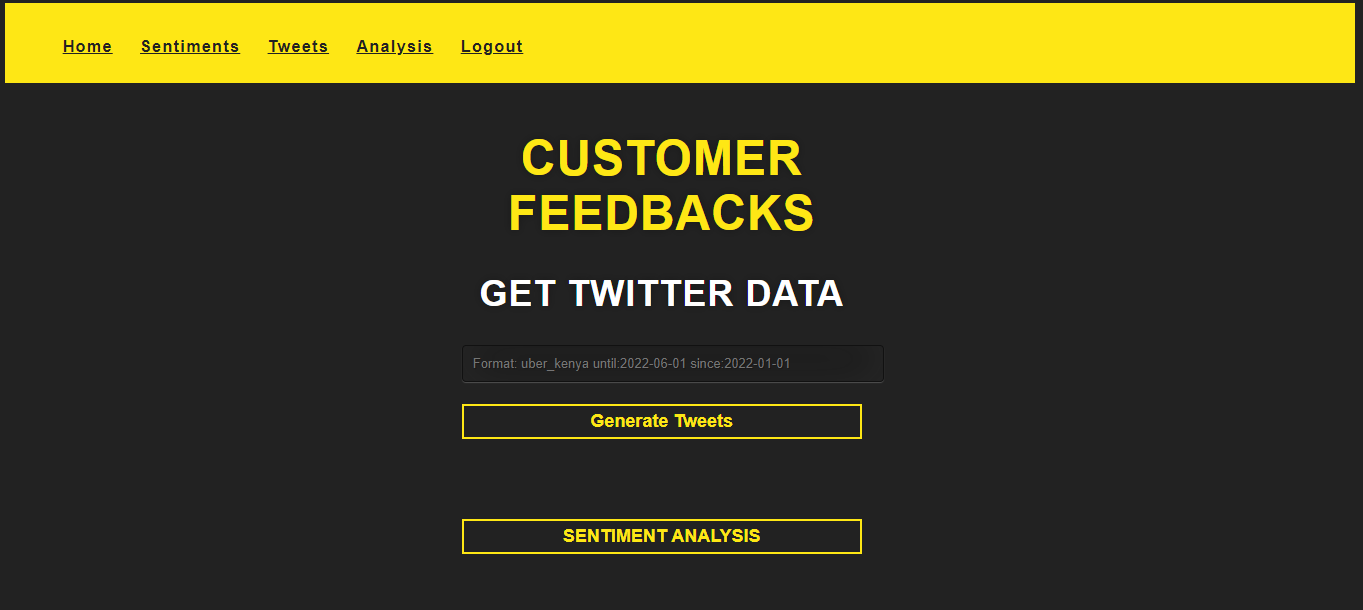


The Home Page

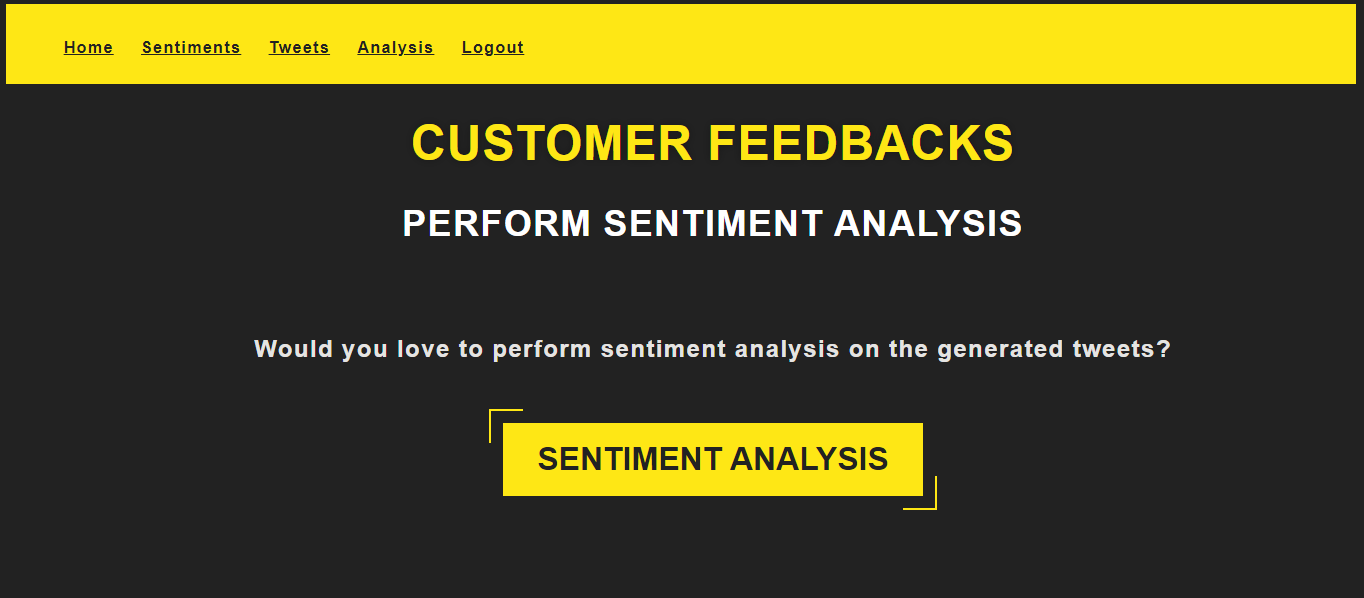


The Sentiment Analysis Page

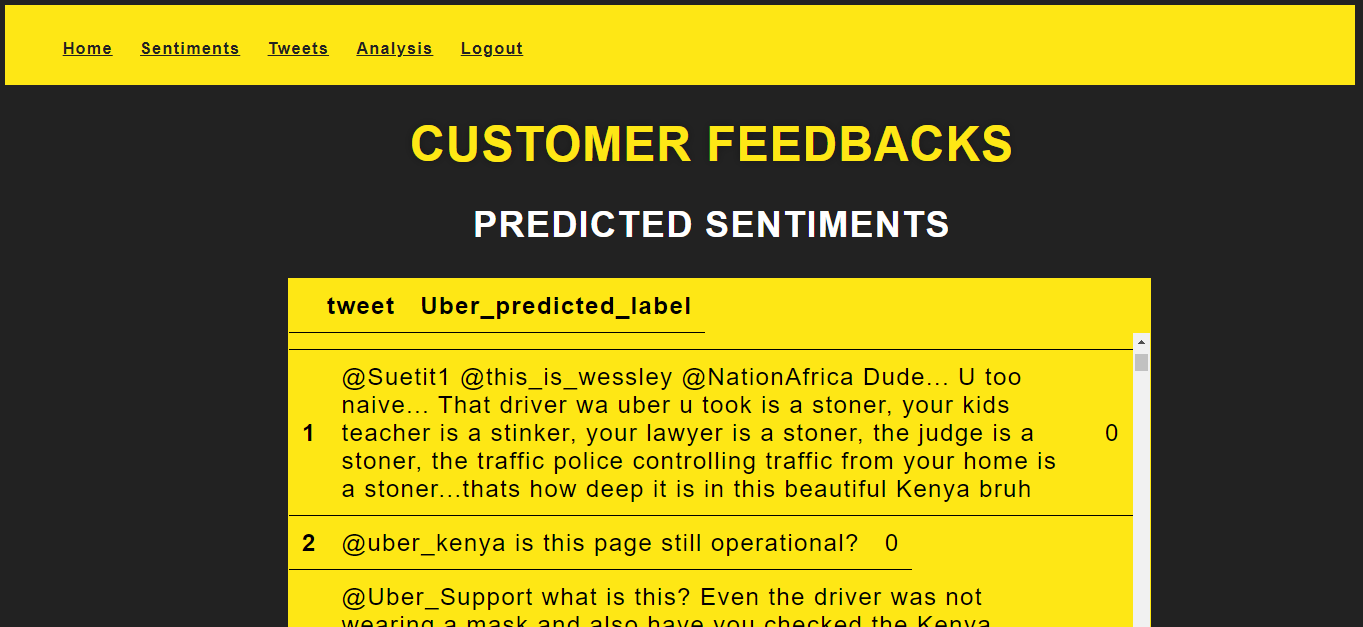
Getting data from twitter



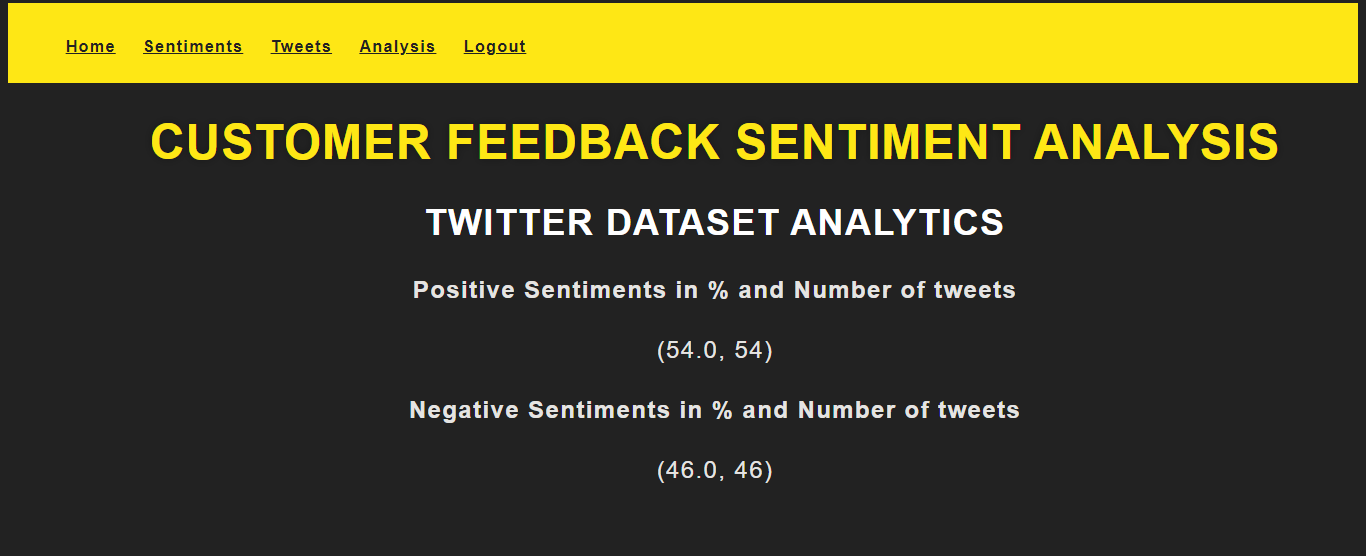
Performing sentiment analysis



Labeled dataset formed using sentiment analysis



Analysis of the labeled dataset

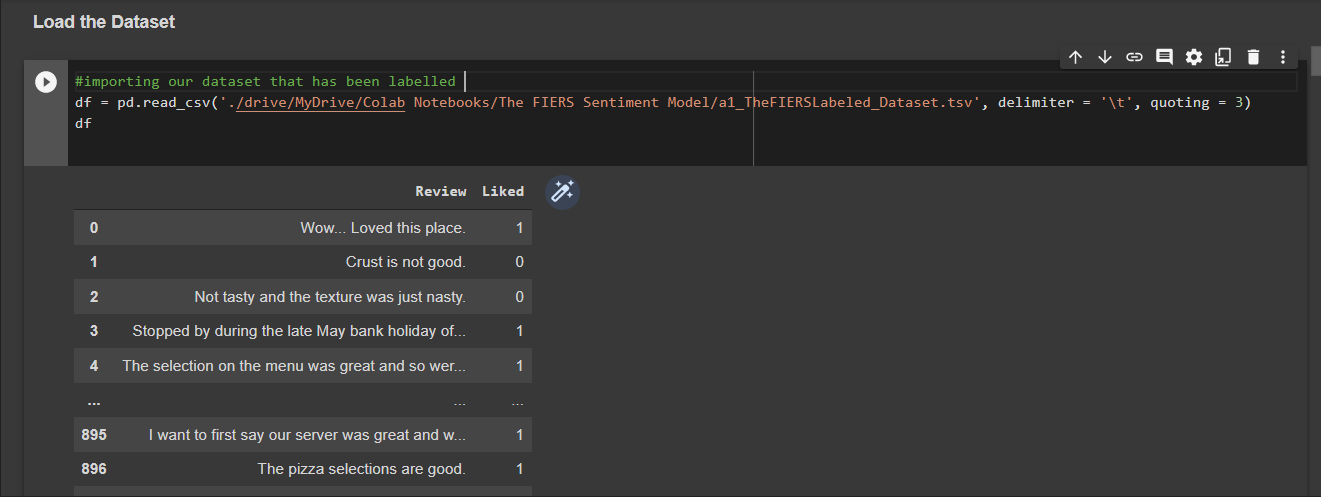


# **CHAPTER 6 IMPLEMENTATION AND TESTING**

6.1 Development environment

Step 1: Importing the dataset

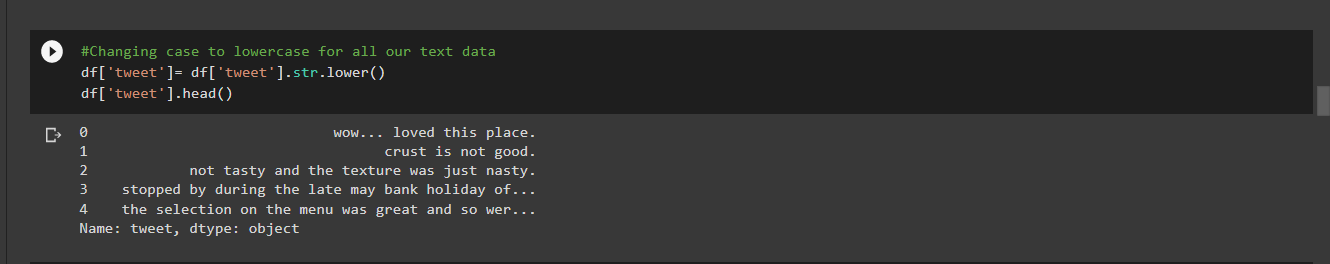
I started by importing the labeled dataset to take it through the preprocessing steps.



Step 2: Exploring and Processing Text Data

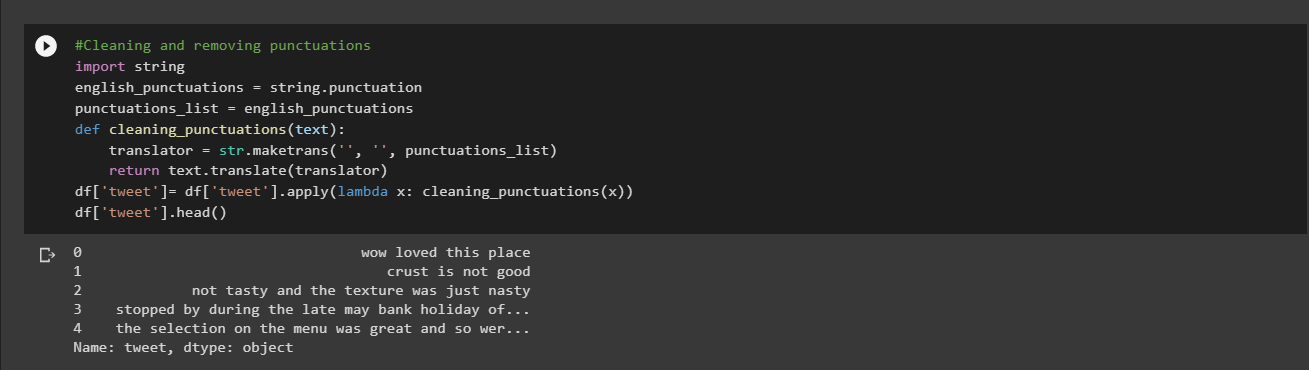
1. *Converting Text Data to Lowercase:*

I used the lower() method to convert all uppercase characters in the tweets into lowercase characters



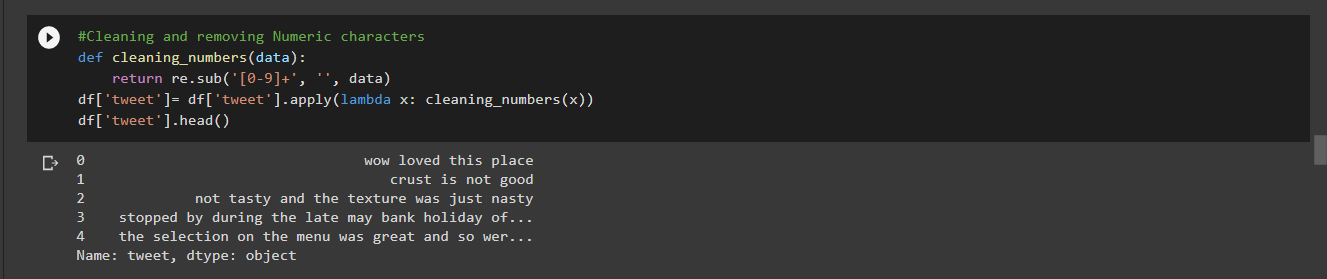
1. *Removing Punctuation*

Next, I moved on to remove any punctuations present in the tweets



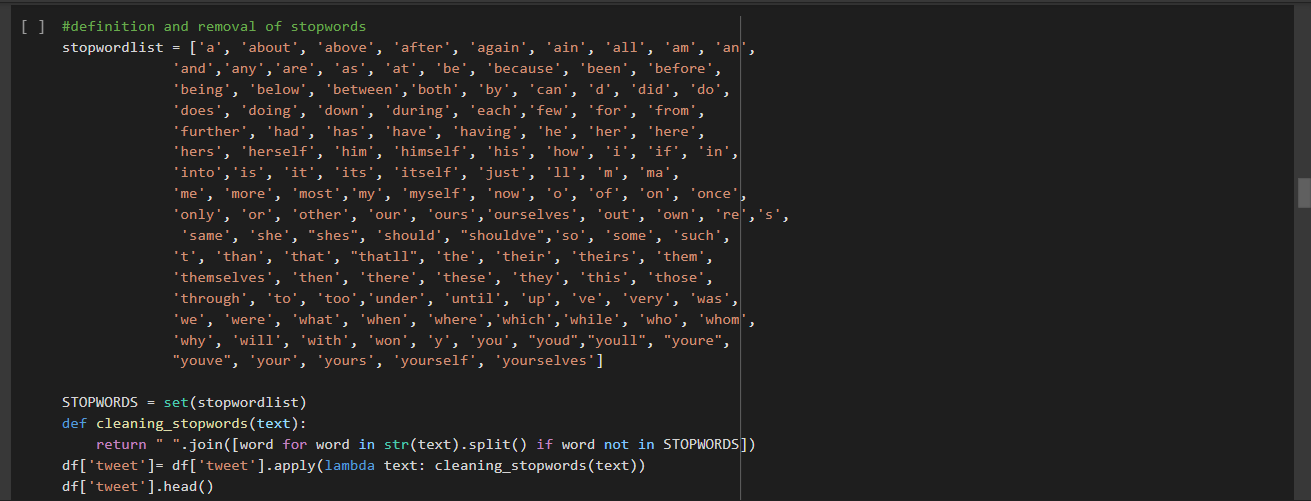
1. *Removing numbers*

I then used the re library to get rid of numeric characters



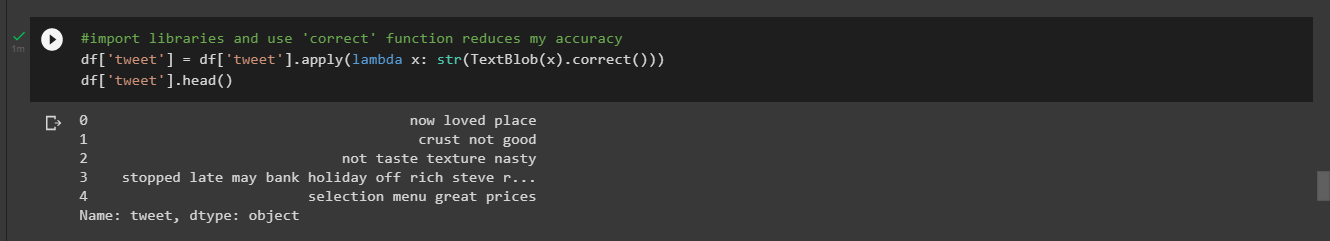
1. *Removing Stop Words*

I made my own list of stop words to remove the words that are less commonly used, so that I can focus on the important keywords instead.



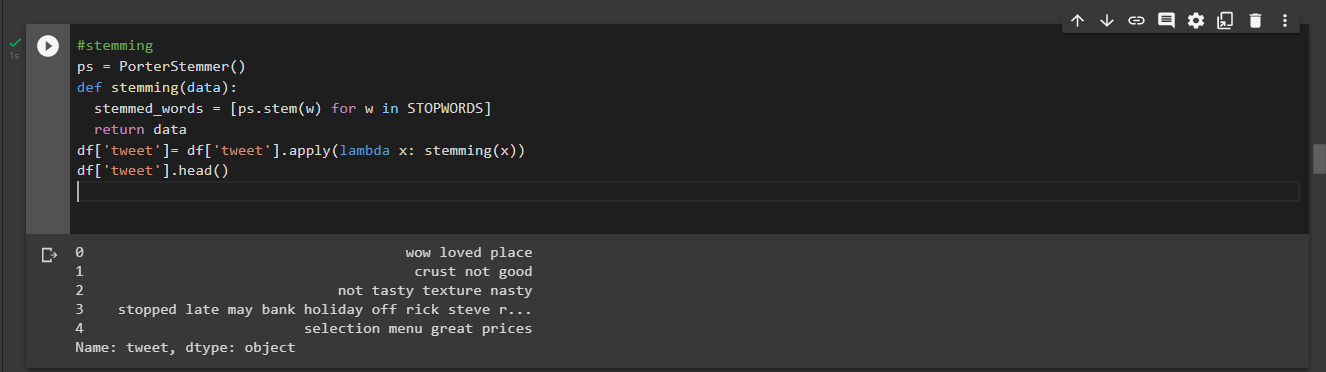
1. *Correcting Spelling:*

Most of the text data in the form was tweets, where there is a high chance of people using short words and making typo errors. So, I corrected the spellings so as to reduce multiple copies of words, which represents the same meaning. I used the correct() function.



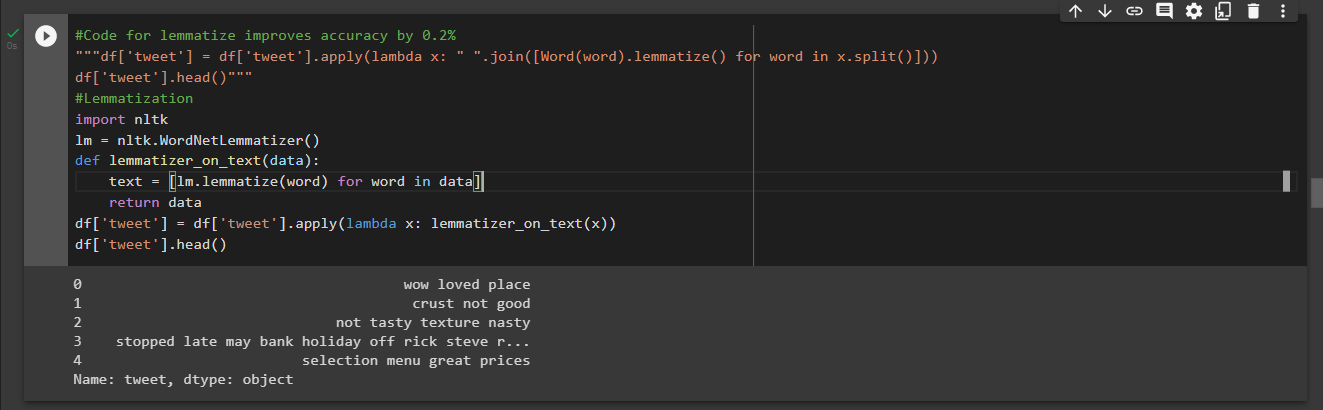
1. *Stemming*

I then used Porter Stemmer() function to perform stemming which eliminates affixes from a word in order to obtain a word stem



1. *Lemmatizing Text*

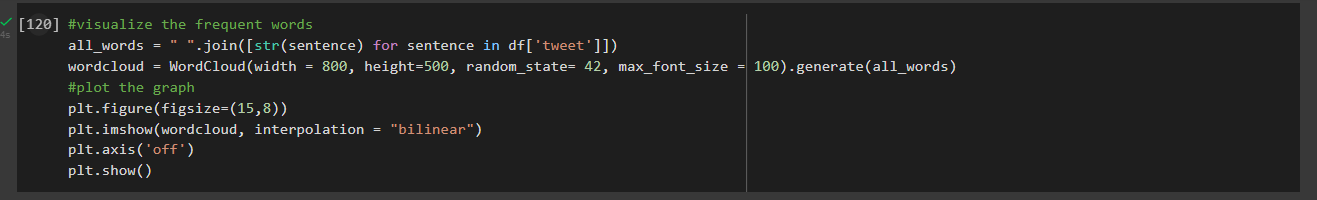
Since stemming sometimes loses meaning of a word I carried out lemmatization which reduces the inflicted word by ensuring its morphological analysis.



1. *Build Wordcloud*

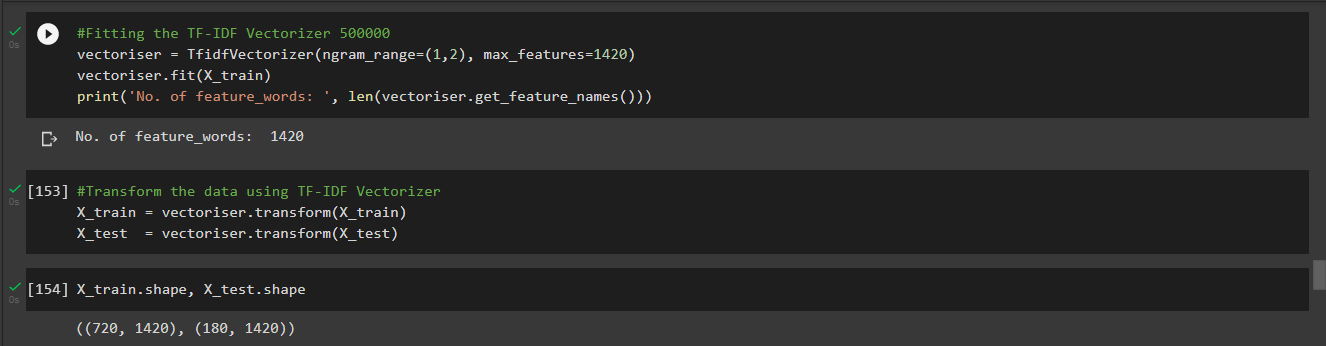
I then build a Wordcloud which is the pictorial representation of the most frequently repeated

words representing the size of the word. So here we can see the frequently used by customers.



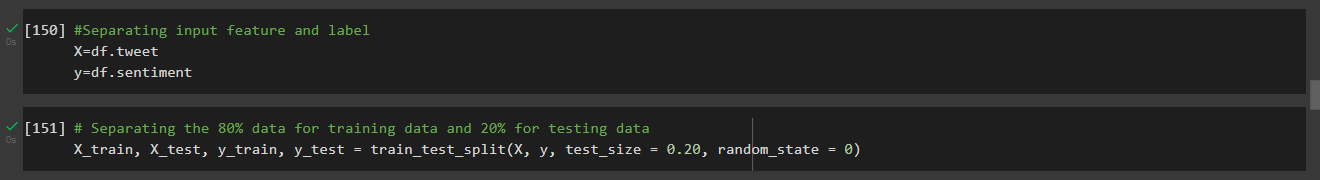
Step 3: Converting Text to Features Using TF-IDF

I then chose TF-IDF for transforming the text into features since algorithms cannot understand the characters/words or sentences, they can only take numbers as input that also includes binaries. I chose my maximum features as 1420 using hit and trial.



Step 4: Splitting into Training and Test set.

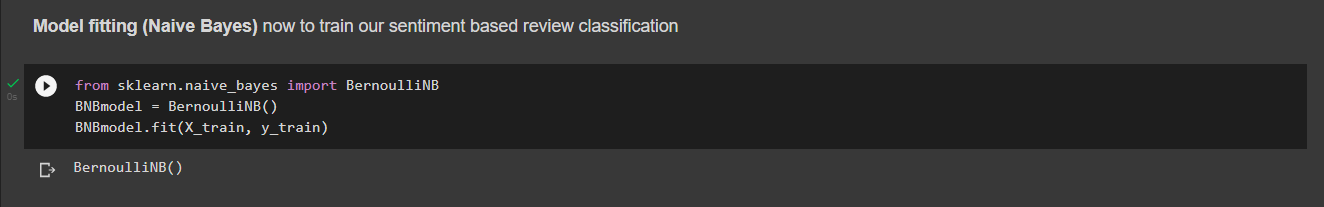
I split my training data and testing data into 80% and 20% respectively.



Step 5: Fitting a Binary Classification Model

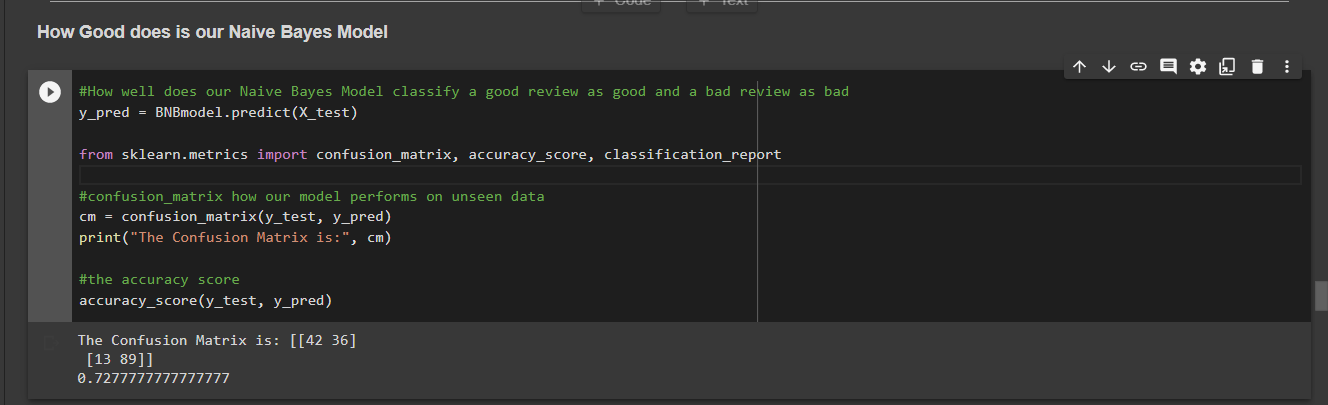
I used three models to carry out the classification so that I could chose the model with the highest accuracy.

*Naive Bayes Binary Classification Model*

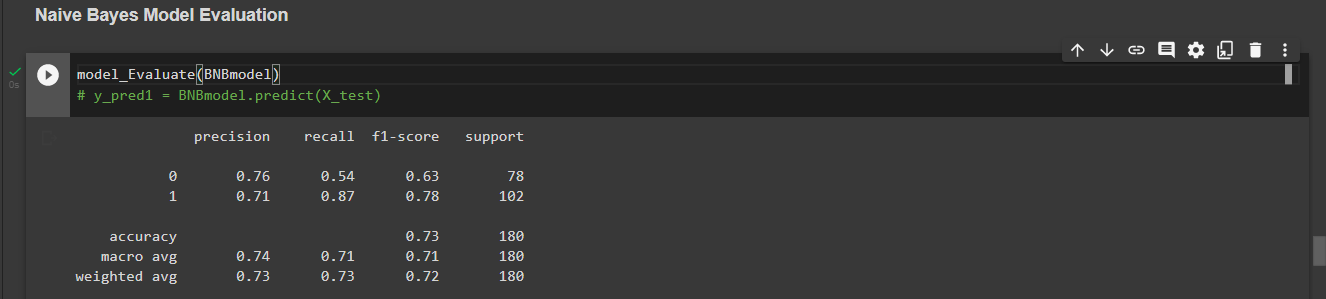


*Accuracy of the Naive Bayes Binary Classification Model*

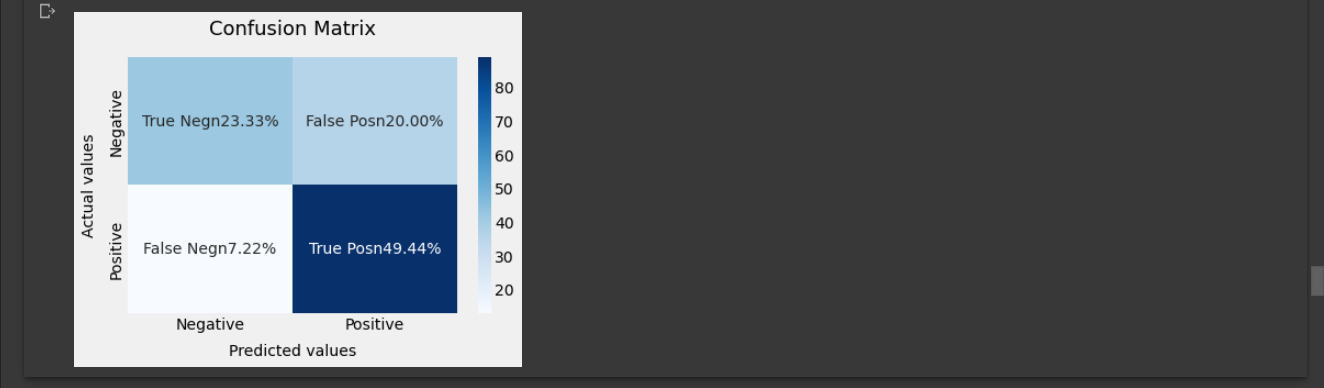
The accuracy of the Naïve Bayes Model was at approximately 73%



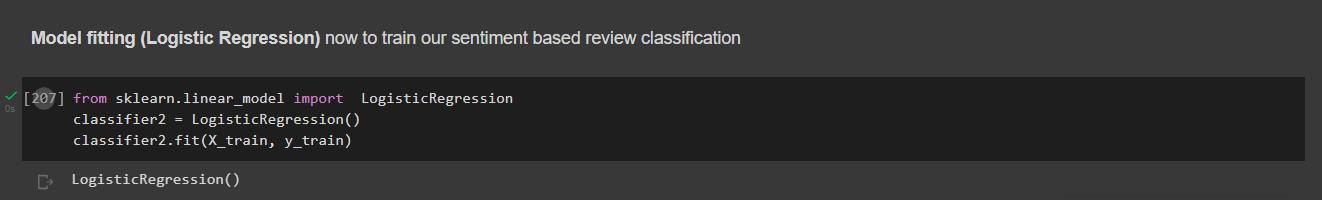
*Model Evaluation of Naive Bayes Binary Classification Model*



*Confusion Matrix of Naive Bayes Binary Classification Model*

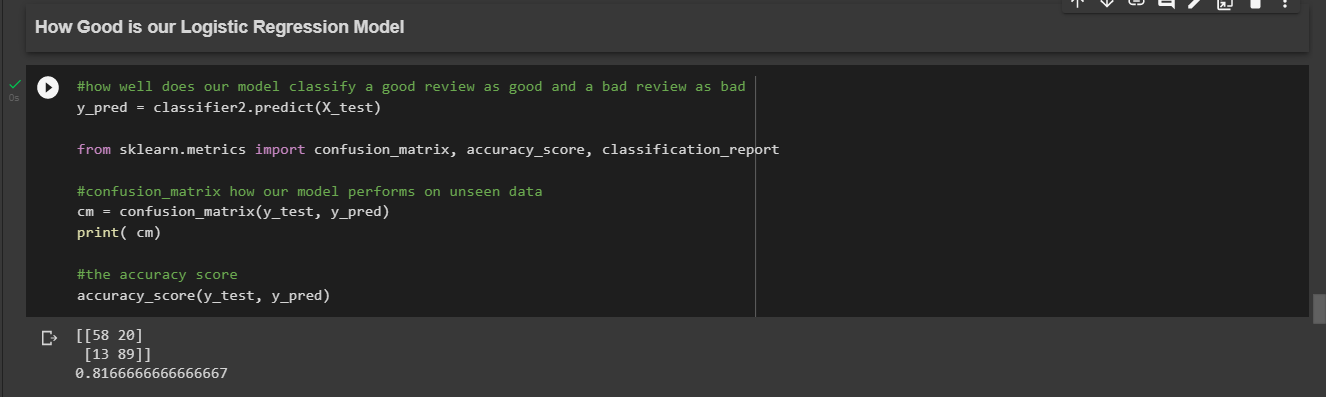


*Logistic Regression Binary Classification Model*

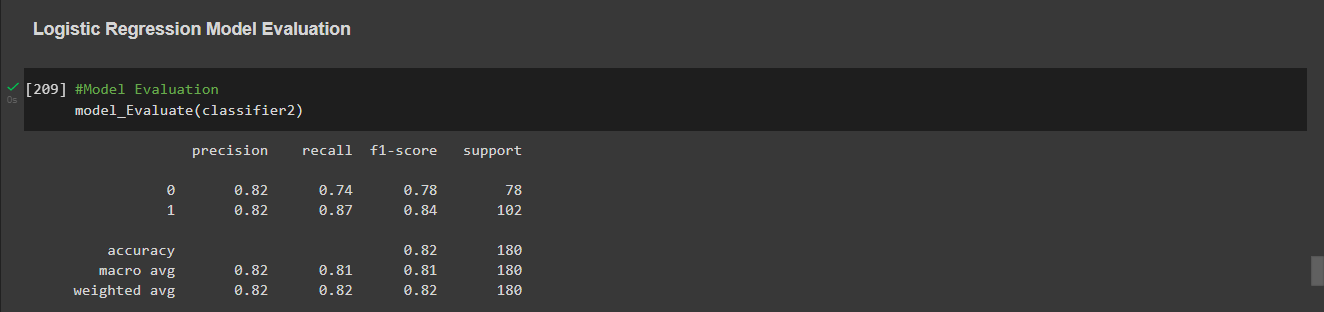


*Accuracy of the Logistic Regression Binary Classification Model*

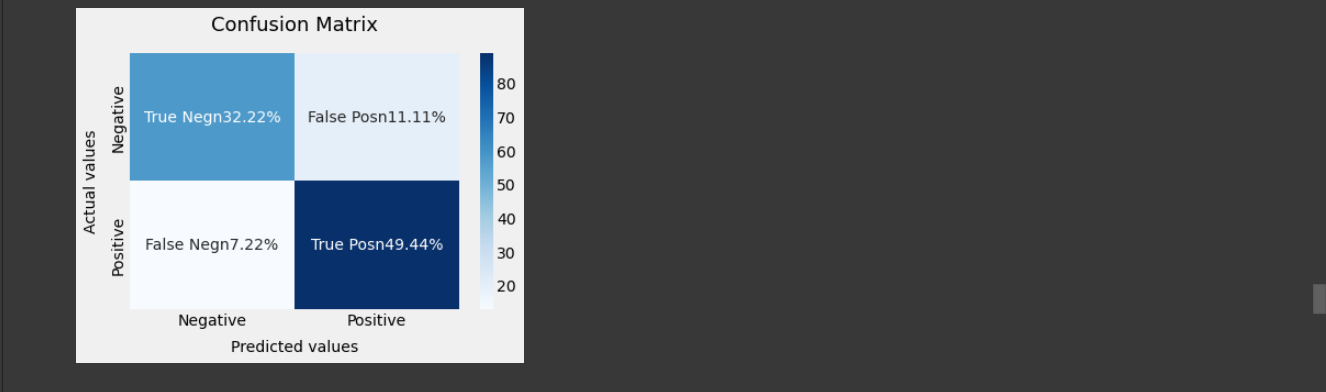
The accuracy of the Logistic Regression Model was at approximately 82%.



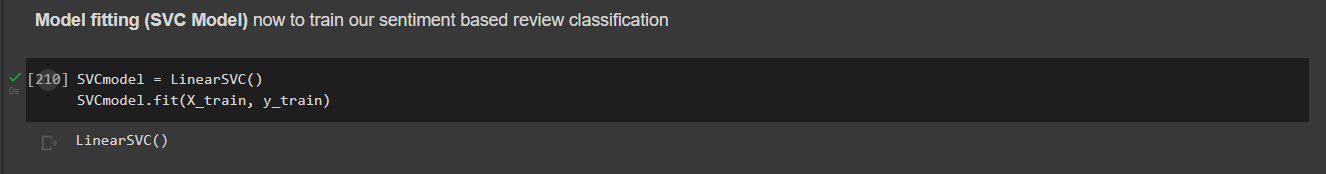
*Model Evaluation of the Logistic Regression Binary Classification Model*



*Confusion Matrix of the Logistic Regression Binary Classification Model*

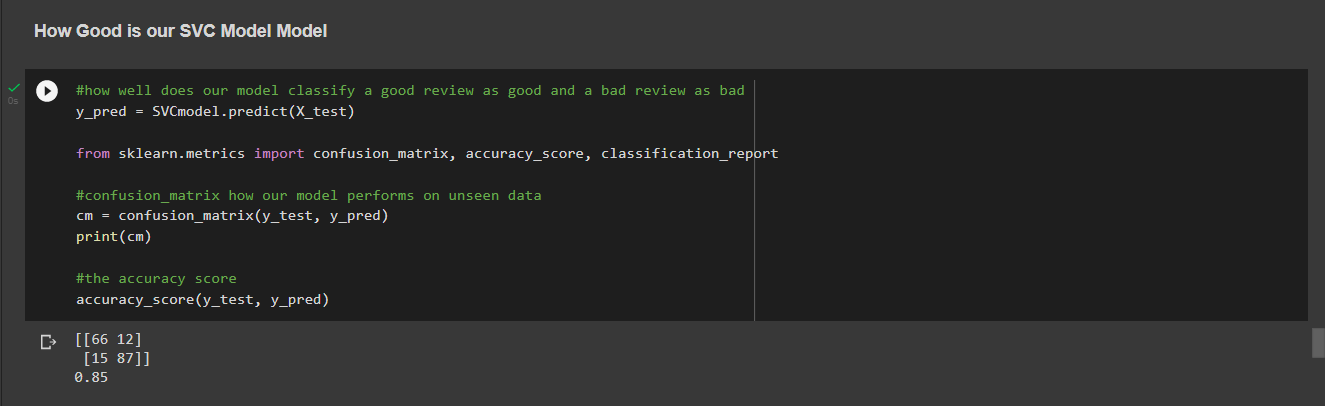


*Support Vector Machine Binary Classification Model*

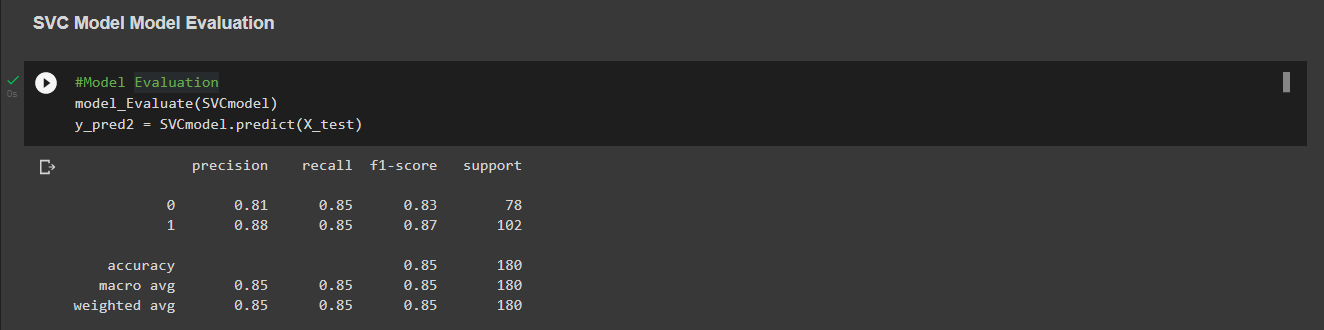


*Accuracy of the Support Vector Machine Binary Classification Model*

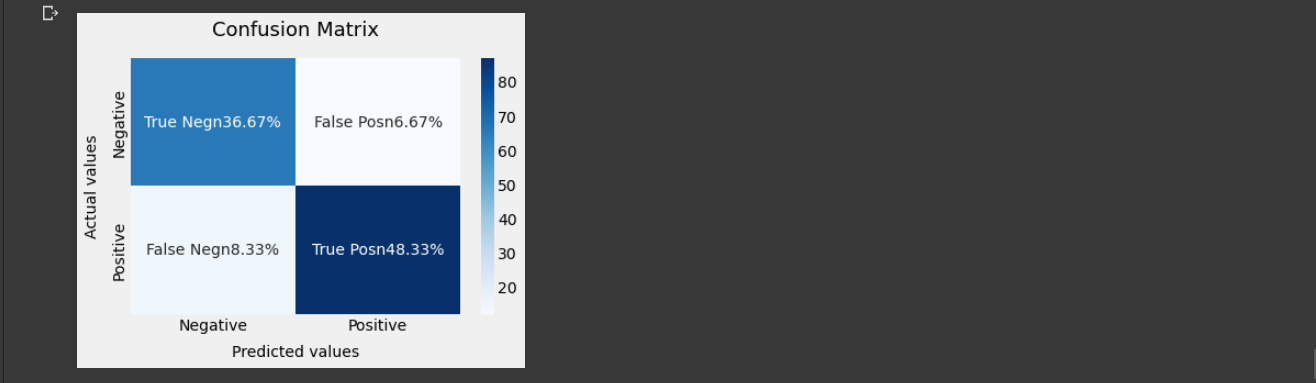
The accuracy of the SVC Model was at approximately 85%



*Model Evaluation of the Support Vector Machine Binary Classification Model*



*Confusion Matrix of the Support Vector Machine Binary Classification Model*



After evaluating all the three above models I chose the SVC model to use for prediction since it has the best accuracy of 85%.

6.2 Test Plan (test data, test cases, test results)

**Test methodology chosen**

**Verification and Validation Methodology (V-Model)**

This method is a step-by-step model of software testing. In this approach the development process happens in parallel with the testing process. As soon as a particular stage of development finishes, i immediately start testing a ready-made part of the product. This approach allows changes to the product at an early stage and saves time and resources in the future.

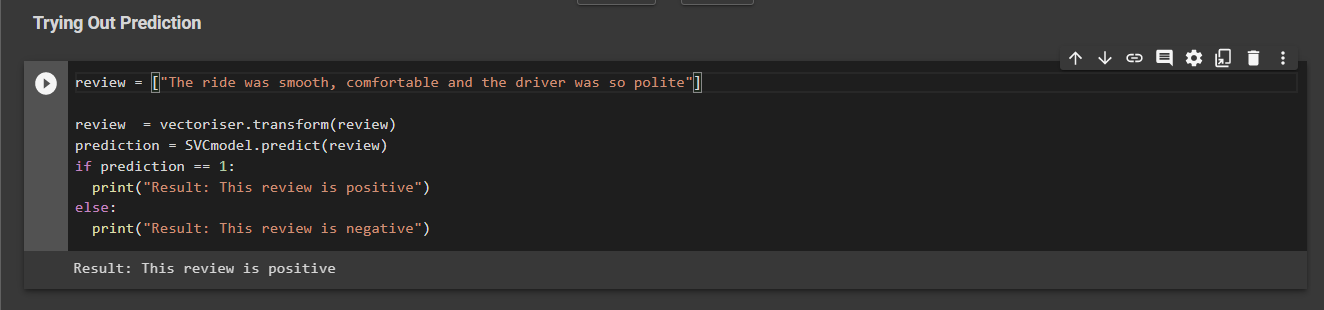
**Test Cases.**

I had two tests, testing the binary classification model to see if it works as intended and the UI test to also see if the application carries out sentiment analysis as intended.

*Testing the SVC Model*

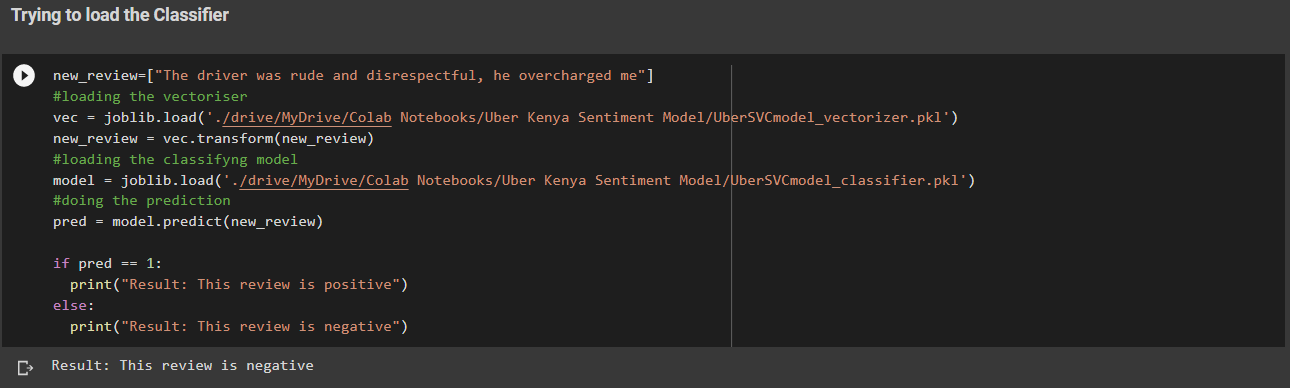
So, I tested the SVC Model to see how well it Predicts the sentiment of a word before taking it into deployment and the prediction was good and satisfying.

*Test Results*



After carrying out model serialization I tried using the SVC Model to predict sentiments and the result was well satisfying.

*Test Results*

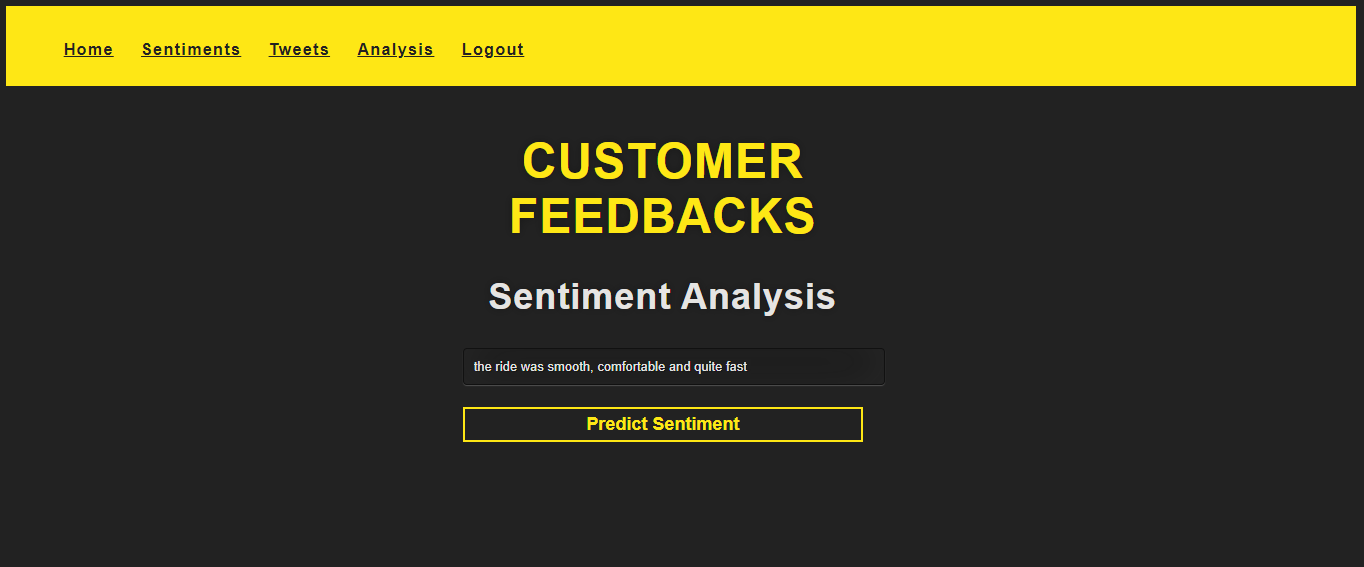


*Testing the Web Application for Sentiment Analysis*

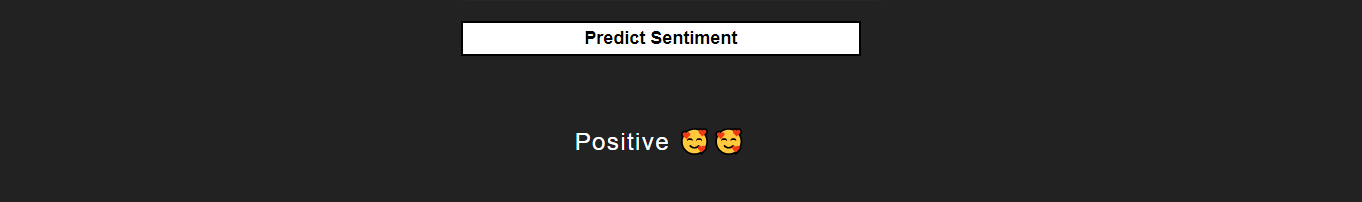
I ran a test on the web application to see if it well predicts the sentiments of a given customer feedback.

**Testing a Positive Sentiment**

**Positive Text Entered:** The ride was smooth, comfortable and quite fast

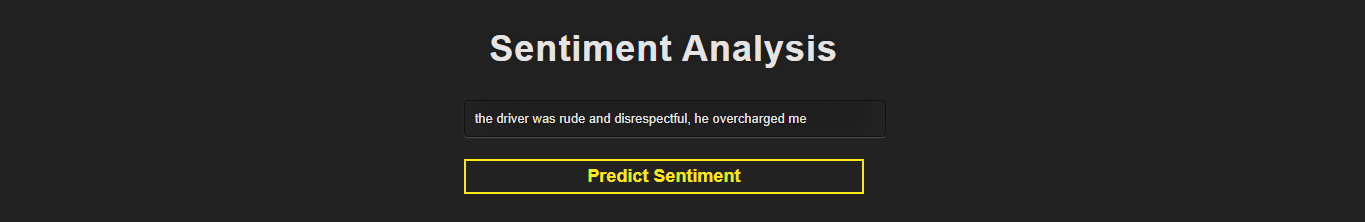


*Test Results*

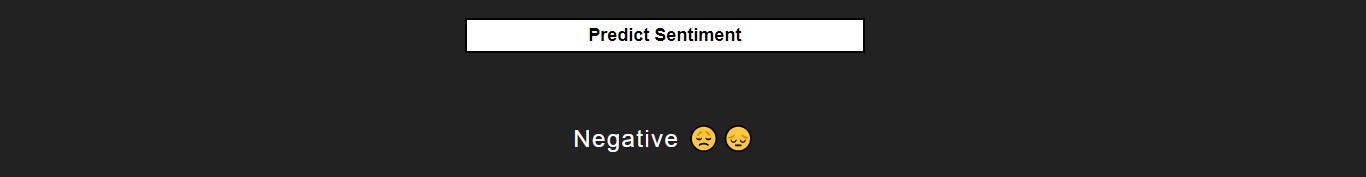


**Testing a negative sentiment**

**Negative Text Entered:** The driver was rude and disrespectful, he overcharged me



*Test Results*



# **CHAPTER 7: RESULTS & CONCLUSION**

7.1 Achievements and lessons learnt

My achievement in this project was to use sentiment analysis using machine learning as a tool for monitoring and understanding client sentiment as they share their opinions and emotions more openly on social media. Uber Kenya can know what makes customers satisfied or frustrated by automatically evaluating customer feedback, such as comments and dialogues on social media especially on Twitter. This will allow them to customize their transportation services and prices to match their customers' demands.

7.2 Conclusions

This system will replace the old systems such as NET PROMOTER SCORE, BRAND CONSIDERATION SCORE, that businesses use to analyze their user feedbacks.

The aim of this study was to build a binary sentiment analysis model using machine learning that can be able to classify whether customer reviews are positive or negative.

The insights gained from sentiment analysis can help Uber Kenya bring accurate changes and transformation of their transport business. It can be in areas that are either creating the most negative sentiment features, such as transportation price, return policies or customer service. Overall, these strategic measures will help the Uber Kenya:

* Become more competitive
* Attract new customers
* Retain present customers
* Make customers more comfortable
* Improve marketing messages and campaigns

7.3 Recommendations

* A more advanced system can be made where by the data collection methods can be increased to be able to fit on all social media platforms.
* The robustness of the system can further be advanced to add more functionalities such as reviewing competitor customer reviews to be able to know their strengths and hence strategize on how to better them in the market.
* Finally, an advanced system can make use of deep learning. This provides more precise results than traditional models and includes neural network models such as Convoluted Neural Network, Recurrent Neural Network, and Deep Neural Network.

# **REFERENCES**

Pang B and Lee L 2002 Sentiment Classification using Machine Learning Techniques Proceeding of ACL-02 Conference on Empirical Methods in Natural Language Processing

A. Kulkarni and A. Shivananda 2019, Natural Language Processing Recipes, Unlocking Text Data with Machine Learning and Deep Learning using Python

B. Bengfort, R. Bilbro, and T. Ojeda 2018, Applied Text Analysis with Python, Enabling Language-Aware Data Products with Machine Learning

T. Beysolow II 2018 Applied Natural Language Processing with Python, Implementing Machine Learning and Deep Learning Algorithms for Natural Language Processing

D. Sarkar 2016, Text Analytics with Python, A Practical Real-World Approach to Gaining Actionable Insights from Your Data

Da Silva NF, Hruschka ER, Hruschka ER Jr (2014) Tweet sentiment analysis with classifier ensembles. Decis Support Syst 66:170–179

Wang L, Niu J, Yu S (2019) SentiDiff: Combining textual information and sentiment diffusion patterns for twitter sentiment analysis. IEEE Trans Knowl Data Eng. https://doi.org/10.1109/TKDE.2019.2913641

Pai P-F, Liu C-H (2018) Predicting vehicle sales by sentiment analysis of Twitter data and stock market values. IEEE Access 6:57655–57662

Rane A, Kumar A (2018) Sentiment classification system of twitter data for US airline service analysis. In: 2018 IEEE 42nd annual computer software and applications conference (COMPSAC). IEEE, vol 1, pp 769–773

# Neethu M, S, Rajasree R,’Sentiment analysis in Twitter using Machine Learning Techniques’, 4th ICCCNT, 2013.

E. Boiy, P. Hens, K. Deschacht and M. Moens, ‘Automatic sentiment analysis in on-line text’, 11th International Conference on Electronic Publishing, vol. 349360, 2007.