

Sentiment analysis of mobile review

B.TECH. PROJECT

Submitted to Sant Gadge Baba Amravati University, Amravati in Partial Fulfillment of the
Requirements for the Degree of **BACHELOR OF TECHNOLOGY** in
INFORMATION TECHNOLOGY.

By

Bhavya Malviya (17007014)

Payal Ekre (17007026)

Digvijay Raut (17007035)

Aditi Linge (17007046)

Under the Guidance of

Prof. B. V. Wakode



DEPARTMENT OF INFORMATION TECHNOLOGY

Government College of Engineering, Amravati
(An Autonomous Institute of Government of Maharashtra)
“Towards Global Technological Excellence”

Government College of Engineering, Amravati
(An Autonomous Institute of Government of Maharashtra)
“Towards Global Technological Excellence”

DEPARTMENT OF INFORMATION TECHNOLOGY



This is to certify that the thesis entitled **“Sentiment Analysis on mobile review”** which is being submitted herewith for the award of the **‘Degree of Bachelor of Technology’ in ‘Information Technology’** of Government College of Engineering, Amravati. This is the result of the work and contribution by **‘Bhavya Malviya (17007014)’**, **‘Payal Ekre (17007026)’**, **‘Digvijay Raut (17007035)’**, **‘Aditi Linge (17007046)’** under my guidance and supervision in VIII Semester of course code ITU808. This work has not been produced earlier for the same award in any other examining body or university to the best of my knowledge and belief.

Prof. B. V. Wakode

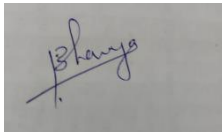
(Project Guide)

DECLARATION

We hereby declare that we have completed the project and written the Project report entitled “**Sentiment Analysis on mobile review**”. It has not previously submitted for the basis of the award of any degree or diploma or other similar title of this project for any other diploma/examining body or university.

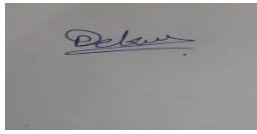
Place : Amravati

Date : 19/06/2021



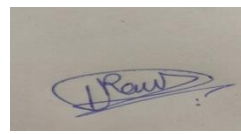
Bhavya Malviya

(17007014)



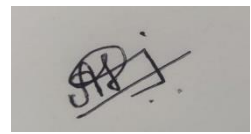
Payal Ekre

(17007026)



Digvijay Raut

(17007035)



Aditi Linge

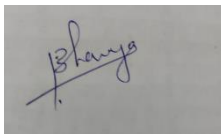
(17007046)

ACKNOWLEDGEMENT

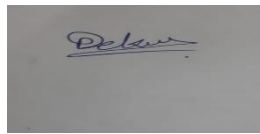
It is our immense pleasure to express a deep gratitude and indebtedness to respected guide Prof.B.V.Wakode for his valuable and inspiring guidance, constant encouragement and suggestion. We are very thankful to extending every facility and constant support for using computer laboratories. Last and not the least we are thankful to the Principal and all staff members of Information Technology Department who helped directly or indirectly during the course of this project successfully.

Place : Amravati

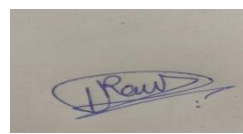
Date : 19/06/2021



Bhavya Malviya
(17007014)



Payal Ekre
(17007026)



Digvijay Raut
(17007035)



Aditi Linge
(17007046)

ABSTRACT

In this everchanging world, where all the things are slowly converting into different machines, we can say that the need of human is getting less. In this century the machines are now becoming capable of reading and understanding human emotions. In the project we have demonstrated how a machine is capable of understanding wither a review on mobile is positive or negative using NLP.

Natural language processing (NLP) is a branch of artificial intelligence that helps computers understand, interpret and manipulate human language. NLP draws from many disciplines, including computer science and computational linguistics, in its pursuit to fill the gap between human communication and computer understanding. While natural language processing isn't a new science, the technology is rapidly advancing thanks to an increased interest in human-to-machine communications, plus an availability of big data, powerful computing and enhanced algorithms

Sentiment analysis is an automated process capable of understanding the feelings or opinions that underlie a text. It is one of the most interesting subfields of NLP, a branch of Artificial Intelligence (AI) that focuses on how machines process human language. Sentiment analysis studies the subjective information in an expression, that is, the opinions, appraisals, emotions, or attitudes towards a topic, person or entity. Expressions can be classified as positive, negative, or neutral.

. In this project we have compared logistic regression algorithm, svc algorithm and random forest classifier to see the sentiment behind the review. This helps seller to identify the most loved product in his/her inventory. There is login page and registration page where if anyone new registers then he/she will be sent a confirmation mail. After login page the user can select a phone to review it and comment. The admin panel has five options of looking review of customers i.e., sentiment, camera, processor, storage, ram and battery for every mobile.

CONTENTS

| | |
|---|---------|
| List of Figures | 1 - 2 |
| 1. INTRODUCTION | 3 |
| 1.1 Introduction | 3 |
| 1.2 Need | 3 |
| 1.3 Objective | 4 |
| 1.4 Theme | 4 |
| 1.5 Organization | 5 |
| 2. LITERATURE SURVEY | 6 – 8 |
| 3. SYSTEM DEVELOPMENT | 9 |
| 3.1 General Information | 9 – 16 |
| 3.2 Hardware and Software Analysis | 16 |
| 3.2.1 Hardware Requirements | 16 |
| 3.2.2 Software Requirements | 16 |
| 3.3 Software Installation | 16 - 19 |
| 3.4 Issues and Some Challenges in Existing System | 19 |
| 3.5 Proposed Systems | 19 - 20 |
| 3.6 Diagram | 20 - 22 |
| 4. PERFORMANCE ANALYSIS | 23 |
| 4.1 IMPLEMENTATION..... | 23 - 38 |
| 4.1.1 Front-End Implementation | 23 - 33 |
| 4.1.2 Database Implementation | 34 - 38 |
| 5. CONCLUSION AND FUTURE SCOPE | 39 - 40 |

| | |
|--------------------------------------|----|
| 6. REFERENCES | 41 |
| 7. Appendix : How to run the project | 42 |

LIST OF FIGURES

| Figure No. | Figure Name | Page No. |
|------------|--|----------|
| 1 | System Development Life Cycle | 10 |
| 2 | Rating Vs Review Graph | 12 |
| 3 | Phases of Sentiment Analysis | 17 |
| 4 | Bokeh | 23 |
| 5 | Use Case Diagram | 25 |
| 6 | Basic working of ML Algorithm | 26 |
| 7 | Login Page | 27 |
| 8 | Confirmation Email | 28 |
| 9 | User Login Template | 29 |
| 10 | Mobile page | 29 |
| 11 | Admin Panel | 30 |
| 12 | Admin Report 1- Sentiment Analysis | 30 |
| 13 | Admin Report 2- Sentiment Analysis - Camera | 31 |
| 14 | Admin Report 3- Sentiment Analysis- Battery | 31 |
| 15 | Admin Report 4- Sentiment Analysis- Storage | 32 |
| 16 | Admin Report 5- Sentiment Analysis- Processor | 32 |
| 17 | Local Server | 33 |
| 18 | Training and Testing dataset | 33 |
| 19 | Accuracy of Random Forest Classification | 34 |
| 20 | Accuracy of Random Logistic Regression(TFDIF) | 34 |
| 21 | Accuracy of Random Logistic Regression(Count Vectorizer) | 35 |
| 22 | Accuracy of Random SVM | 35 |
| 23 | Accuracy of Random Logistic Regression(N- Gram) | 36 |
| 24 | Feature Name | 36 |

| | | |
|----|----------------------------|----|
| 25 | User Table Description | 37 |
| 26 | Table Description | 37 |
| 27 | Positive Description | 38 |
| 28 | Negative Table Description | 38 |
| 29 | Negative Table Sentiment | 39 |
| 30 | Product Table | 39 |

1. INTRODUCTION

1.1 INTRODUCTION

Sentiment analysis is the automated process of understanding the sentiment or opinion of a given text. You can use it to automatically analyze reviews and sort them.

This project is to classify the positive and negative reviews of the customers over different mobile phones and build a supervised learning model to polarize large amounts of reviews.

We extracted the features of our dataset and built several supervised models based on that. These models include traditional algorithms such as Naive bayes, Random Forest and Logistic Regression. We compared the accuracy these models and got a better understanding of the polarized attitudes towards the mobiles. This project is giving the maximum accurate results which are calculated using a Kaggle dataset.

The aim of this project is to investigate if sentimental analysis is feasible for the classification of mobile reviews from different buyers and hence strategize the further business strategy. Therefore, we will compare the performance of different classification algorithms on the binary classification (positive vs. negative) of mobile reviews from different customers. This greatly impacts the sales data and customer engagement on a particular mobile.

Sentiment analysis thus helps the business to get customer engagement, feedback and do market research.

1.2 NECESSITY

Online shopping has been growing for 20 years and many e-commerce websites such as Amazon and Flipkart have been created to meet the increasing demand. As customers usually want the best quality which at the same time satisfy all their expectations and needs but can't directly check it, reviews from other customers seem to be the most reliable way to decide whether to buy the product or not. Therefore, sentiment analysis has proven essential to understand a product's popularity among the buyers all over the world.

The owners and brand can use such automated technologies and ways to fulfil the needs of their customers. The sellers also need the customer feedback and machine learning helps to get this done in the feature extracted way too. They can further make improvements and have a

good business plan and can also modify the mobile's features. Machine learning makes all this automated and easy.

1.3 GOALS AND OBJECTIVES

- Prediction of reviews of mobile phones
- Prediction of popularity of mobile phones with a dataset
- By analyzing these reviews based on various features of mobiles, we categorize the sentiment as positive and negative
- The analysis is projected using various algorithms that gives an easier outlook to the customers as well as sellers to strategize their business

1.4 Theme

This project provides an automated way and user-friendly interface for customers and sellers. There are two main users- Customers and Admin. Following Represents the components of the project.

- Launch the website.
- Homepage Provides options for Signup and Login.
- New User can register and get an email after successful registration.
- Once the user is verified, User can login using credentials.
- A user can add comment to a mobile and view its price and other comments.
- An admin can review all the sentiments and sentiment to specific feature in his/her panel.

1.5 ORGANIZATION

Chapter 1 is Introduction, which consists of brief introduction about the project. It also contains the need of the project in the real world. The main objectives are also included in this section.

Chapter 2 is Literature Review, which consists of related information available in standard Books, Journals, Transactions, and various resources and how our project is needed in today's world.

Chapter 3 is of System Development, which includes the method used for development of the project. This section gives an understanding of the model development.

Chapter 4 is Performance Analysis, which consists of details, results at various stages and comparison between various results along with screenshots of the application.

Chapter 5 is of Conclusion, which includes the future scope of this project and applications of the project.

2. LITERATURE SURVEY

In this section, we study about the research done on the Sentiment Analysis of mobile phones and product reviews. Sentiment analysis deals with the classification of texts based on the sentiments they contain. Sentiment analysis of product reviews has recently become very popular in text mining and computational linguistics research. The applications of sentiment analysis are broad and powerful. The ability to extract insights from social data is a practice that is being widely adopted by organisations across the world. Shifts in sentiment on online

platform have been shown to correlate with shifts in the market. This acts as an essential part of market research and customer service approach. Customers are important for any market or product(mobile) to achieve profits and success. The overall customer experience of your users can be revealed quickly with sentiment analysis, but it can get far more granular too. Sentiment analysis is a classification process whereby machine learning techniques are applied on text-driven datasets in order to analyze its sentiment, e.g. a message being positive or negative about a certain topic.

Understanding such sentiments involves several tasks. Firstly, evaluative terms expressing opinions must be extracted from the review. Secondly the polarity, of the opinions must be determined. For instance, “slow” and “brilliant” respectively carry a negative and a positive opinion. Thirdly, the opinion strength, or the intensity, of an opinion should also be determined. For instance, both “brilliant” and “good” indicate positive opinions, but “brilliant” obviously implies a stronger preference. Finally, the review is classified with respect to sentiment classes, such as Positive and Negative, based on the polarity of the opinions it contains.

Feature specific sentiment analysis for mobile phone reviews is another way to understand the experience of users on the mobile phones. The objective is realized by identifying a set of potential features in the review and extracting opinion expressions about those features by exploiting their associations. Capitalizing on the view that more closely associated words come together to express an opinion about a certain feature, dependency parsing is used to identify relations between the opinion expressions. The system learns the *set of significant relations* to be used by dependency parsing and a *threshold parameter* which allows us to merge closely associated opinion expressions. The data requirement is minimal as this is a *one time learning* of the *domain independent parameters*.

Most existing sentiment analysis algorithms were designed for binary classification, meaning that they assign opinions or reviews to bipolar classes such as Positive or Negative (Turney, 2002; Pang et al., 2002; Dave et al., 2003). Some recently proposed algorithms extend binary sentiment classification to classify reviews with respect to multi-point rating scales, a

problem known as rating inference (Pang and Lee, 2005; Goldberg and Zhu, 2006; Leung et al., forthcoming). Rating inference can be viewed as a multi-category classification problem, in which the class labels are scalar ratings such as 1 to 5 “stars”. Some sentiment analysis algorithms aim at summarizing the opinions expressed in reviews towards a given product or its features (Hu and Liu, 2004a; Gamon et al., 2005).

Depending on the purpose, sentiment analysis algorithm can be used at the following scopes:

- Document-level - for the entire text.
- Sentence-level - obtains the sentiment of a single sentence.
- Sub-sentence level - obtains the sentiment of sub-expressions within a sentence.

Broadly, are two major Sentiment Analysis methods.

Rule based approach : It is based on an algorithm with a clearly defined description of an opinion to identify. Includes identify subjectivity, polarity, or the subject of opinion.

Automatic Sentiment Analysis : It involves supervised machine learning classification algorithms. In fact, sentiment analysis is one of the more sophisticated examples of how to use classification to maximum effect. In addition to that, unsupervised machine learning algorithms are used to explore data.

Sentiment analysis may involve the following types of classification algorithms:

- Linear Regression
- Naive Bayes
- Support Vector Machines
- RNN derivatives LSTM and GRU.

Sentiment analysis has evolved over time. The old approach was Bayesian sentiment which requires simple sums and logarithms. However, it makes strong assumptions and also rely on representative datasets Another approach is sentence-level training and classification.

These simple changes had a massive impact in reducing the number of overrides that our customers produce every month. In particular, overrides on news

documents reduced by 58% on average across the 16 supported languages. The analysis involved around 450M news documents and 4.2M overrides produced by 7,193 customers.

There is a continuous improvement to train the models in a better way. Logistic Regression is one of the ways to help to get the maximum possible accuracy. It is a classification algorithm used to solve binary classification problems. The logistic regression classifier uses the weighted combination of the input features and passes them through a sigmoid function. Sigmoid function transforms any real number input, to a number between 0 and 1. It can be used on both Bag-of-triGrams and Tf-Idf features to compare their accuracy scores. It can be build the models on the default parameters . N-gram approach has helps us to get good accuracy with less computational costs.

Sentiment analysis has a lot of companies such as Apple, KFC ,IMDB to understand their audience better. As a result, this can be a significant factor in the product's successful establishment on the market.

At the later stages, the use of sentiment analysis in product analytics merges with brand monitoring and provides a multi-dimensional view of the product and its brand:

- How the brand/product is perceived by various target audience segments?
- Which elements of the product or its presentation are the points of contention and in what light?

Accurate target audience segmentation and subsequent value proposition formulation are amongst the key elements of effective business operation. You need to know where are you aiming at with what.

There are some challenges to Sentiment Analysis that occur because of variety of customers and their opinions. Irony and Sarcasm ,tone, subjectivity and context are some of the challenges. While it's difficult to speculate how a relatively immature system might evolve in the future, there is a general assumption that sentiment analysis needs to move beyond a one-dimensional positive to negative scale. For the future, to truly understand and capture the broad range of emotions that humans express as written word, we need a more sophisticated multidimensional scale that can perform text analytics measure skepticism, hope, anxiety, excitement ,etc.

Some apps such as Appbot sentiment analysis tool (<https://appbot.co/app-review-sentiment-analysis/>) allows you to visualize your app review sentiment so you can understand how users feel. Appbot also surfaces bugs, feature requests and more in your app reviews to help you find opportunities to improve your app review sentiment and star rating. Such progressions in technologies help for future scope in sentiment analysis. These apps helps to perform sentiment analysis easily.

Our project helps to derive the sentiments based on features on mobiles such as RAM, processor, camera, battery which are the essentials of any mobile. Thus, this helps the brands and markets to find the pros and cons and demands of the market

3. SYSTEM DEVELOPMENT

System development is the process of defining, designing a new application. It could include the internal development of customized systems, the creation of database systems and updation of older version. Under this section, we have presented all model and databases, which we have considered for the development of the system, and all the steps, which are used for the development of system.

3.1 General Information

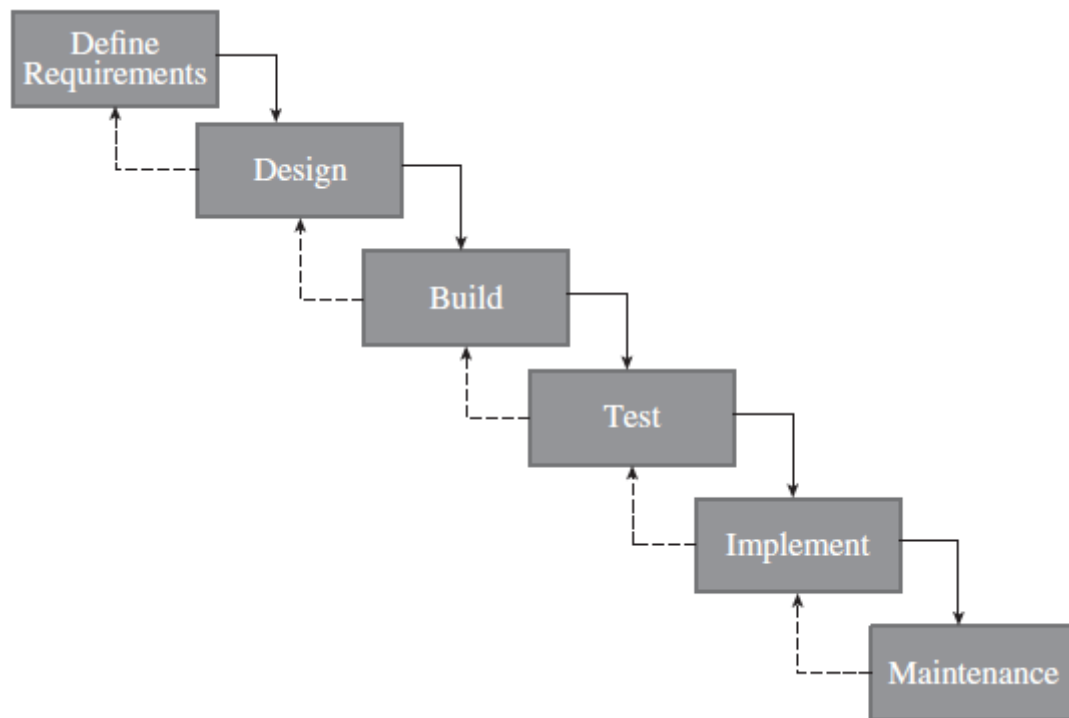


Figure 1: System development Lifecycle

Analysis:-

The Analysis stage of the is mainly comprised of the discovery and understanding of system needs. The problem domain must be evaluated, information gathered, and system requirements diagramed, prioritized, and documented. We analyze, refine, and scrutinize the gathered requirements to make consistent and unambiguous requirements. This activity reviews all requirements and may provide a graphical view of the entire system. After the completion of the analysis, it is expected that the understandability of the project may improve significantly.

Sentiment Analysis is the process of determining whether a piece of writing is positive, negative or neutral. A sentiment analysis system for text analysis combines natural language processing (NLP) and machine learning techniques to assign weighted sentiment scores to the entities, topics, themes and categories within a sentence or phrase.

Design:-

In this system, we are finding ratings for feature reviews. Since on Amazon reviews there are ratings available for each review, the sentiment for the product review and service review will be equivalent to the review given by the customer so the computational task of finding the

sentiments is reduced in case of service and product reviews and as we are reviewing only for positive and negative sentiment we have allocated positive sentiment for the review with 3 or more stars and negative review for less than three stars.



Our first step for data scraping is done already as we have downloaded data from Kaggle. The data was already pre-processed and we only have to drop some null values and allot the sentiment according to reviews.

After all the cleaning the data is stored in a variable and is used to summarize and visualize data. Finally, the data is trained and used to predict the sentiment of review.

The diagram Fig 2 shows that how many people have rated all the reviews, as we can see there are many review voters for 5-star review product which are more than 500 at a point.

The algorithm used in our model is logistic regression and its three different features namely:

- Count vectorizer
- TFIDF
- N-gram

Along with logistic regression we also used SVC and Random forest classifier.

Logistic regression:

In general, there are two different types of classification models: generative models (Naive Bayes, Hidden Markov Models, etc.) and discriminative models (Logistic Regression, SVM, etc.). Ultimately, both models try to compute $p(\text{class}|\text{features})$, or $p(y|x)$. The key difference is that a generative model tries to model the joint probability distribution $p(x, y)$ first and then compute the conditional probability $p(y|x)$ using Baye's Theorem, whereas a discriminative one

directly models $p(y|x)$. We want to classify all review texts into one of the two categories: positive sentiment and negative sentiment. Therefore, we will first need to transform the ‘stars’ value into the two categories. Here, we can treat ‘4.0’ and ‘5.0’ as positive sentiment and ‘1.0’ and ‘2.0’ as negative sentiment. We can treat ‘3.0’ as neutral or even treat each star as its own sentiment category, which would make it a multi-class classification problem. However, for the sake of a simply binary classification, we can take the ones with ‘3.0’ out.

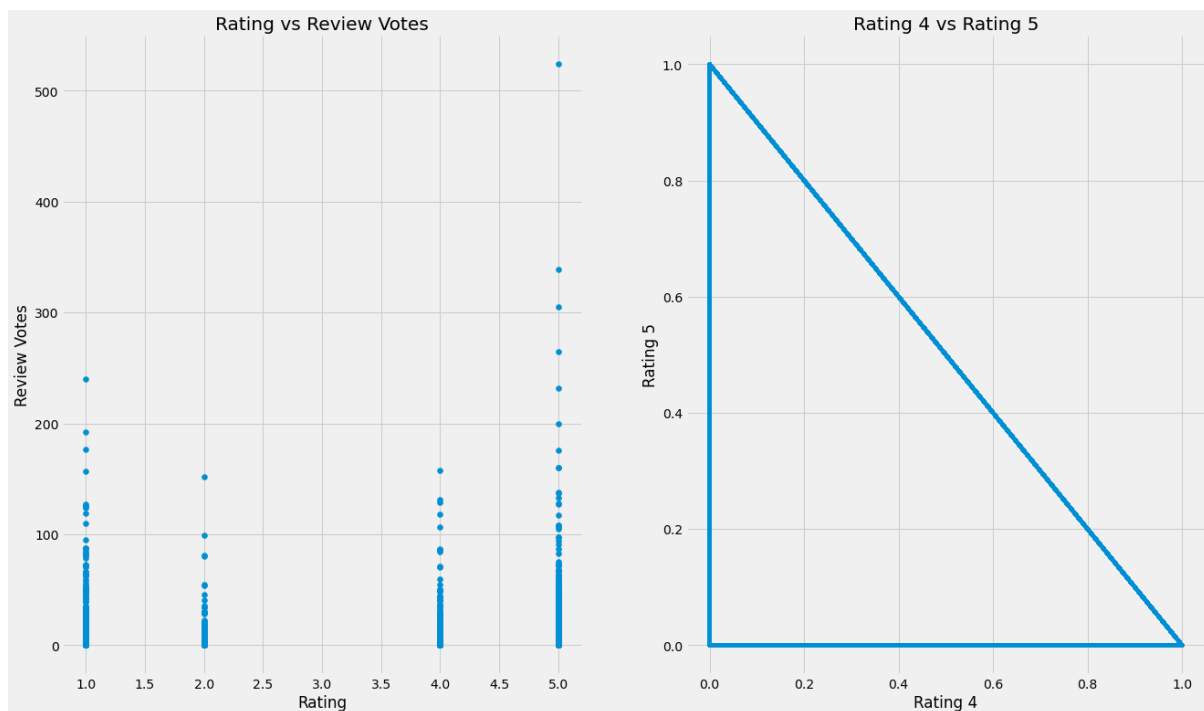


FIG.2.

Vectorization:

For a model to be able to process text input, we would need to convert them into vectors. There are a few different ways to represent these text features and here are the most common ones: 1. Binary, e.g. whether the word “good” is present. 2. Count, e.g. how many times does the word “good” appear in this review, similar to the Bag of Word model in Naive Bayes. 3. TF-IDF, which is a weighted importance of each text feature relevant to the document

One nice thing about logistic regression is that we can find the importance of each feature easily. We can find out which are the words which impact positively and which words impact negatively. Note that a high positive feature importance correlates to high possibility of Class 1 and a low negative (high absolute value) feature importance correlates to high possibility of Class 0.

1. **Count vectorizer:** CountVectorizer tokenizes(tokenization means breaking down a sentence or paragraph or any text into words) the text along with performing very basic preprocessing like removing the punctuation marks, converting all the words to lowercase, etc. The vocabulary of known words is formed which is also used for encoding unseen text later. An encoded vector is returned with a length of the entire vocabulary and an integer count for the number of times each word appeared in the document. Let's take an example to see how it works.

Out of all the countries of the world, some countries are poor, some countries are rich, but no country is perfect.

Table A.

| | out | of | all | the | countries | world | some | are | poor | rich | but | no | country | is | perfect |
|-----|-----|----|-----|-----|-----------|-------|------|-----|------|------|-----|----|---------|----|---------|
| doc | 1 | 2 | 1 | 2 | 3 | 1 | 2 | 2 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

Table B.

| index | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|-------|---|---|---|---|---|---|---|---|---|---|----|----|----|----|----|
| doc | 1 | 2 | 1 | 2 | 3 | 1 | 2 | 2 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

The row of the above matrix represents the document, and the columns contain all the unique words with their frequency. In case a word did not occur, then it is assigned zero corresponding to the document in a row.

2. **TFIDF:** TF-IDF stands for “Term Frequency — Inverse Document Frequency”. This is a technique to quantify a word in documents, we generally compute a weight to each word which signifies the importance of the word in the document and corpus. If I give you a sentence for example “This building is so tall”. Its easy for us to understand the sentence as we know the semantics of the words and the sentence. But how will the computer understand this sentence? The computer can understand any data only in the form of numerical value. So, for this reason we vectorize all of the text so that the computer can understand the text better.

Term frequency (TF) measures the frequency of a word in a document. This highly depends on the length of the document and the generality of word. TF is individual to each document and word; hence we can formulate TF as follows.

$$tf(t,d) = \text{count of } t \text{ in } d / \text{number of words in } d$$

Dominance frequency (DF) measures the importance of document in whole set of corpus, this is very similar to TF. The only difference is that TF is frequency counter for a term t in document d , whereas DF is the count of occurrences of term t in the document set N . In other words, DF is the number of documents in which the word is present.

Count Vectorizer

| | blue | bright | sky | sun |
|------|------|--------|-----|-----|
| Doc1 | 1 | 0 | 1 | 0 |
| Doc2 | 0 | 1 | 0 | 1 |

TD-IDF Vectorizer

| | blue | bright | sky | sun |
|------|----------|----------|----------|----------|
| Doc1 | 0.707107 | 0.000000 | 0.707107 | 0.000000 |
| Doc2 | 0.000000 | 0.707107 | 0.000000 | 0.707107 |

Here , we can see clearly that Count Vectorizer give number of frequency with respect to index of vocabulary where as *tf-idf* consider overall documents of weight of words.

$$tf(t, d) = \sum_{x \in d} fr(x, t)$$

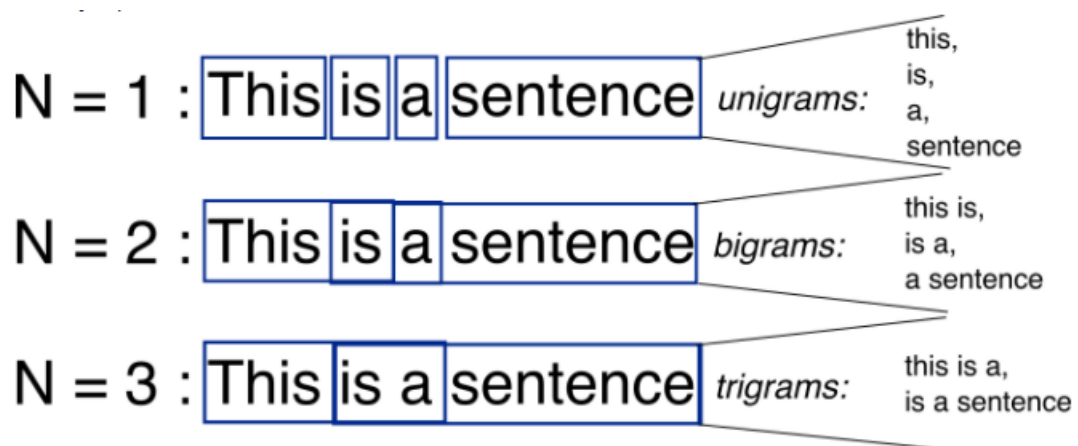
where the $fr(x, t)$ is a simple function defined as:

$$fr(x, t) = \begin{cases} 1, & \text{if } x = t \\ 0, & \text{otherwise} \end{cases}$$

3. **N-gram:** N-grams are simply all combinations of adjacent words or letters of length n that you can find in your source text. For example, given the word fox, all 2-

grams (or “bigrams”) are fo and ox. You may also count the word boundary – that would expand the list of 2-grams to ‘#f’, ‘fo’, ‘ox’, and ‘x#’, where # denotes a word boundary. You can do the same on the word level. As an example, the hello, world! text contains the following word-level bigrams: ‘# hello’, ‘hello world’, ‘world #’. The basic point of n-grams is that they capture the language structure from the statistical point of view, like what letter or word is likely to follow the given one. The longer the n-gram (the higher the n), the more context you have to work with. Optimum length really depends on the application – if your n-grams are too short, you may fail to capture important differences. On the other hand, if they are too long, you may fail to capture the “general knowledge” and only stick to particular.

n-grams are used for a variety of things. Some examples include auto completion of sentences (such as the one we see in Gmail these days), auto spell check (yes, we can do that as well), and to a certain extent, we can check for grammar in a given sentence. We’ll see some examples of this later in the post when we talk about assigning probabilities to n-grams. Using these n-grams and the probabilities of the occurrences of certain words in certain sequences could improve the predictions of auto completion systems.



These are the methods in logistic regression algorithm which we used in the model to train the mobile review data.

Now let's see about SVC and Random Forest Classifier

Support Vector Classifier:

The objective of the support vector machine algorithm is to find a hyperplane in an N-dimensional space (N — the number of features) that distinctly classifies the data points.

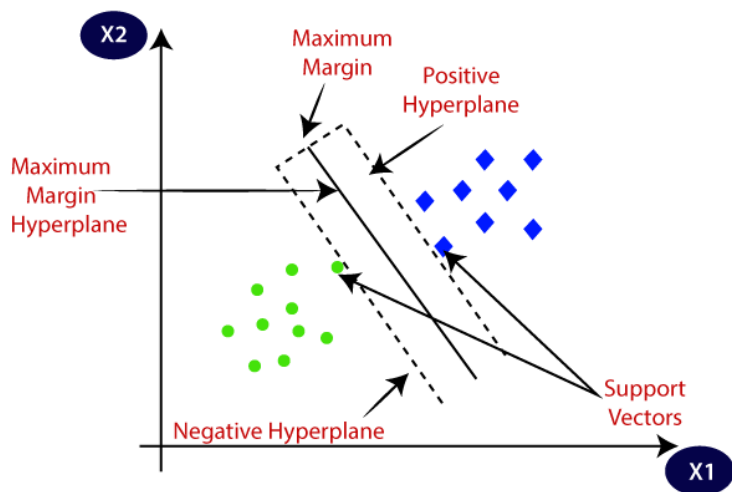
Support Vector Machine (SVM) is a relatively simple **Supervised Machine Learning Algorithm** used for classification and/or regression. It is more preferred for classification but is sometimes very useful for regression as well. Basically, SVM finds a hyper-plane that creates a boundary between the types of data. In 2-dimensional space, this hyper-plane is nothing but a line.

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. Next, find the optimal hyperplane to separate the data. So by this, you must have understood that inherently, SVM can only perform binary classification (i.e., choose between two classes). However, there are various techniques to use for multi-class problems.

Support Vector Machine for Multi-Class Problems

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.
- **Non-linear SVM:** Non-Linear SVM is used for non-linearly separated data, which means if a dataset cannot be classified by using a straight line, then such data is termed as non-linear data and classifier used is called as Non-linear SVM classifier.

Hyperplane: There can be multiple lines/decision boundaries to segregate the classes in n-dimensional space, but we need to find out the best decision boundary that helps to classify the data points. This best boundary is known as the hyperplane of SVM.

The dimensions of the hyperplane depend on the features present in the dataset, which means if there are 2 features (as shown in image), then hyperplane will be a straight line. And if there are 3 features, then hyperplane will be a 2-dimension plane.

We always create a hyperplane that has a maximum margin, which means the maximum distance between the data points.

Support Vectors:

The data points or vectors that are the closest to the hyperplane and which affect the position of the hyperplane are termed as Support Vector. Since these vectors support the hyperplane, hence called a Support vector.

```
In [17]: from sklearn.svm import SVC
          from sklearn.metrics import roc_auc_score
          model1 = SVC(kernel='linear', random_state=0)
          model1.fit(X_train_vectorized,y_train)
          predictions = model1.predict(vect.transform(X_test))
          print('AUC: ',roc_auc_score(y_test,predictions))
```

```
AUC:  0.8975711995090037
```

Random Forest Classifier

Here, individual decisions trees are combined to make a random forest. Decision trees as they are the building blocks of the random forest model. It technically is an ensemble method (based on the divide-and-conquer approach) of decision trees generated on a randomly split dataset. This collection of decision tree classifiers is also known as the forest. The individual decision trees are generated using an attribute selection indicator such as information gain, gain ratio, and Gini index for each attribute. Each tree depends on an independent random sample. In a classification problem, each tree votes and the most popular class is chosen as the final result. In the case of regression, the average of all the tree outputs is considered as the final result. It is simpler and more powerful compared to the other non-linear classification algorithms.

It works in four steps:

1. Select random samples from a given dataset.
2. Construct a decision tree for each sample and get a prediction result from each decision tree.
3. Perform a vote for each predicted result.
4. Select the prediction result with the most votes as the final prediction.

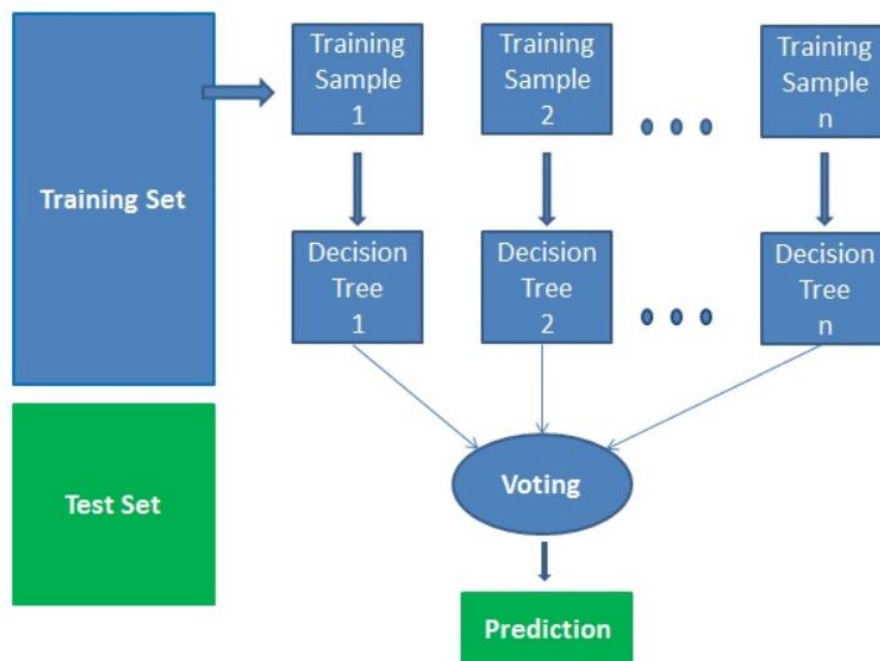


Figure : Depicting Random Forest Classifier

Disadvantages:

- Random forests is slow in generating predictions because it has multiple decision trees. Whenever it makes a prediction, all the trees in the forest have to make a prediction for the same given input and then perform voting on it. This whole process is time-consuming.
- The model is difficult to interpret compared to a decision tree, where you can easily make a decision by following the path in the tree.

```
In [20]: from sklearn.ensemble import RandomForestClassifier
model2 = RandomForestClassifier(n_estimators=300, max_features="auto")
model2.fit(X_train_vectorized,y_train)
predictions = model2.predict(vect.transform(X_test))
print('AUC: ',roc_auc_score(y_test,predictions))
```

AUC: 0.867141692815005

Implementation

After the design is completed, the system must be implemented. All coding for programs is completed, software components are constructed, and data from the old system is converted into the new system.

Now the model is completed we have to make the user interface and databases to interact and store data. We have used HTML, CSS, JS for front-end and flask for back-end.

The model was saved in .pkl format using pickle library in python.

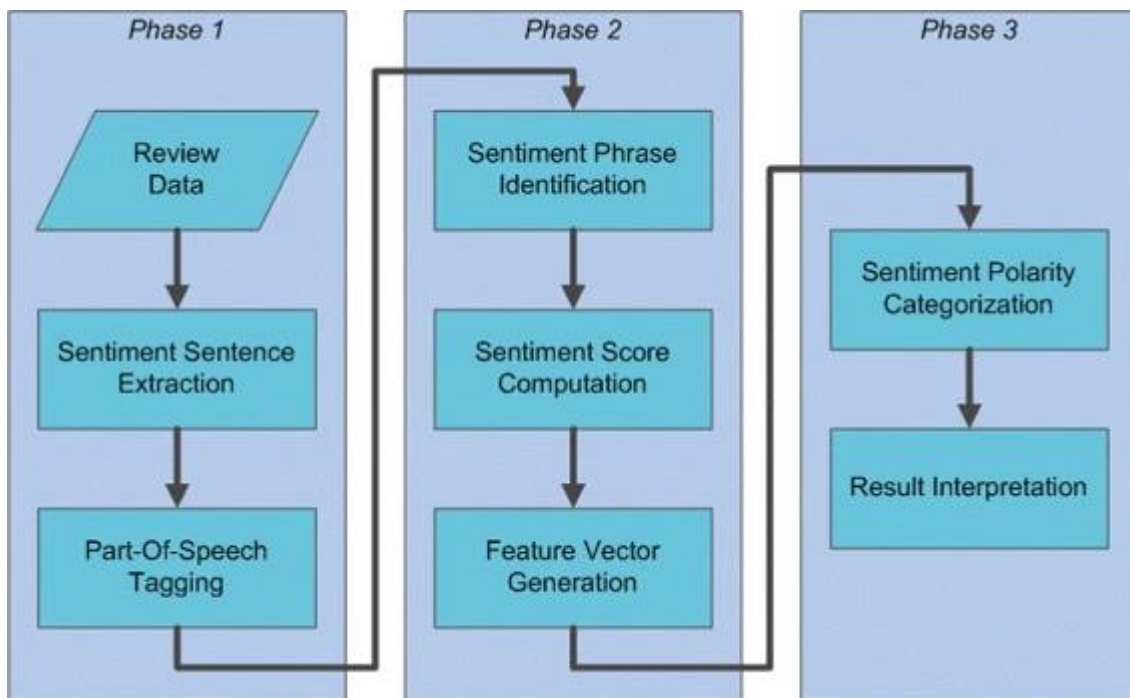


Fig 3 Phases of Sentiment Analysis

Python pickle module is used for serializing and de-serializing python object structures. The process to convert any kind of python objects (list, dict, etc.) into byte streams (0s and 1s) is called pickling or serialization or flattening or marshall. We can convert the byte stream (generated through pickling) back into python objects by a process called as unpickling.

Below are some of the common exceptions raised while dealing with pickle module –

- `Pickle.PicklingError`: If the pickle object doesn't support pickling, this exception is raised.
- `Pickle.UnpicklingError`: In case the file contains bad or corrupted data.
- `EOFError`: In case the end of file is detected, this exception is raised.

Pros:

- Comes handy to save complicated data.
- Easy to use, lighter and doesn't require several lines of code.
- The pickled file generated is not easily readable and thus provide some security.

Cons:

- Languages other than python may not be able to reconstruct pickled python objects.
- Risk of unpickling data from malicious sources.

This has implications both for recursive objects and object sharing. Recursive objects are objects that contain references to themselves. These are not handled by marshal, and in fact, attempting to marshal recursive objects will crash your Python interpreter. Object sharing happens when there are multiple references to the same object in different places in the object hierarchy being serialized. pickle stores such objects only once, and ensures that all other references point to the master copy. Shared objects remain shared, which can be very important for mutable objects.

We have compared the following three methods in logistic regression.

```
In [12]: from sklearn.metrics import roc_auc_score,roc_curve

#predict the transform test document.
predictions = model.predict(vect.transform(X_test))
print('AUC: ',roc_auc_score(y_test,predictions))

AUC:  0.8974332776669326
```

```
In [15]: X_train_vectorized = vect.transform(X_train)

model = LogisticRegression()
model.fit(X_train_vectorized, y_train)

predictions = model.predict(vect.transform(X_test))

print('AUC: ', roc_auc_score(y_test, predictions))

AUC: 0.889951006492175
```

```
In [20]: model = LogisticRegression()
model.fit(X_train_vectorized, y_train)

predictions = model.predict(vect.transform(X_test))

print('AUC: ', roc_auc_score(y_test, predictions))

C:\Users\LENOVO\anaconda3\lib\site-packages\sklearn\line
(status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the
https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative s
https://scikit-learn.org/stable/modules/linear\_model.html
n_iter_i = _check_optimize_result(

AUC: 0.9104640361714084
```

As seen in above screenshots we can see that the N-gram method has the highest accuracy.

At last, this model is saved in pickle format using pickle library as follows:

```
In [54]: import pickle
pickle.dump(model, open("vectorizer.pickle", "wb"))
```

```
In [46]: import pickle
pickle.dump(vect, open("vect.pickle", "wb"))
```

The first one is to make model and second one is to make vectorizer in pickle format.

These model and vector is used in predicting the review given by user in backend. Then the sentiment is predicted by this model and update the database.

Project Types

There are various types of requirements required for the development of the website.

The website has a Front end developed using HTML,CSS and JavaScript. The backend is managed by Flask and the Database used is MySQL. Along with this Bokeh is used for data visualization of the sentiment.

1. Front end

HTML provides the basic structure of sites, which is enhanced and modified by other technologies like CSS and JavaScript.

CSS is used to control presentation, formatting, and layout.

JavaScript is used to control the behaviour of different elements.

The front end has

- Login and Register page
- Home Page displaying all the mobile phones
- Mobile page(Individual)
- Admin Page to view Sentiments
- Customer Page to enter reviews

2. Backend

FLASK

Flask is a web application framework written in Python.

A Web-Application Framework or Web Framework is the collection of modules and libraries that helps the developer to write applications without writing the low-level codes such as protocols, thread management, etc. Flask is based on WSGI(Web Server Gateway Interface) toolkit and Jinja2 template engine.

Web Server Gateway Interface (WSGI) has been adopted as a standard for Python web application development. WSGI is a specification for a universal interface between the web server and the web applications.

It is a WSGI toolkit, which implements requests, response objects, and other utility functions. This enables building a web framework on top of it. The Flask framework uses Werkzeug as one of its bases.

Jinja2 is a popular templating engine for Python. A web templating system combines a template with a certain data source to render dynamic web pages.

Flask is often referred to as a micro framework. It aims to keep the core of an application simple yet extensible. Flask does not have built-in abstraction layer for database handling, nor does it have form a validation support. Instead, Flask supports the extensions to add such functionality to the application

Python 2.6 or higher is usually required for installation of Flask. Although Flask and its dependencies work well with Python 3 (Python 3.3 onwards), many Flask extensions do not support it properly.

Install virtual env for development environment

virtualenv is a virtual Python environment builder. It helps a user to create multiple Python environments side-by-side. Thereby, it can avoid compatibility issues between the different versions of the libraries.

The following command installs virtualenv

```
pip install virtualenv
```

This command needs administrator privileges. Add sudo before pip on Linux/Mac OS. If you are on Windows, log in as Administrator

On Windows, following can be used

```
venv\scripts\activate
```

We are now ready to install Flask in this environment.

```
pip install Flask
```

The above command can be run directly, without virtual environment for system-wide installation.

We can Send Form Data to the HTML File of Server:
A Form in HTML is used to collect the information of required entries which are then forwarded and stored on the server. These can be requested to read or modify the form. The flask provides this facility by using the URL rule. In the given example below, the ‘/’ URL renders a web page(student.html) which has a form. The data filled in it is posted to the ‘/result’ URL which triggers the result() function. The results() function collects form data present in request.form in a dictionary object and sends it for rendering to result.html.

Http protocol is the foundation of data communication in world wide web.

GET

Sends data in unencrypted form to the server. Most common method.

POST

Used to send HTML form data to server. Data received by POST method is not cached by server.

3. Database

MySQL is a relational database management system based on SQL – Structured Query Language. The application is used for a wide range of purposes, including data warehousing, e-commerce, and logging applications. The most common use for MySQL however, is for the purpose of a web database.

MySQL is a database system that runs on a server. MySQL is ideal for both small and large applications. It is very fast, reliable, and easy to use.

MySQL is developed, distributed, and supported by Oracle Corporation.

4. Visualization

Bokeh is a Python library for interactive visualization that targets web browsers for representation. This is the core difference between Bokeh and other visualization libraries.

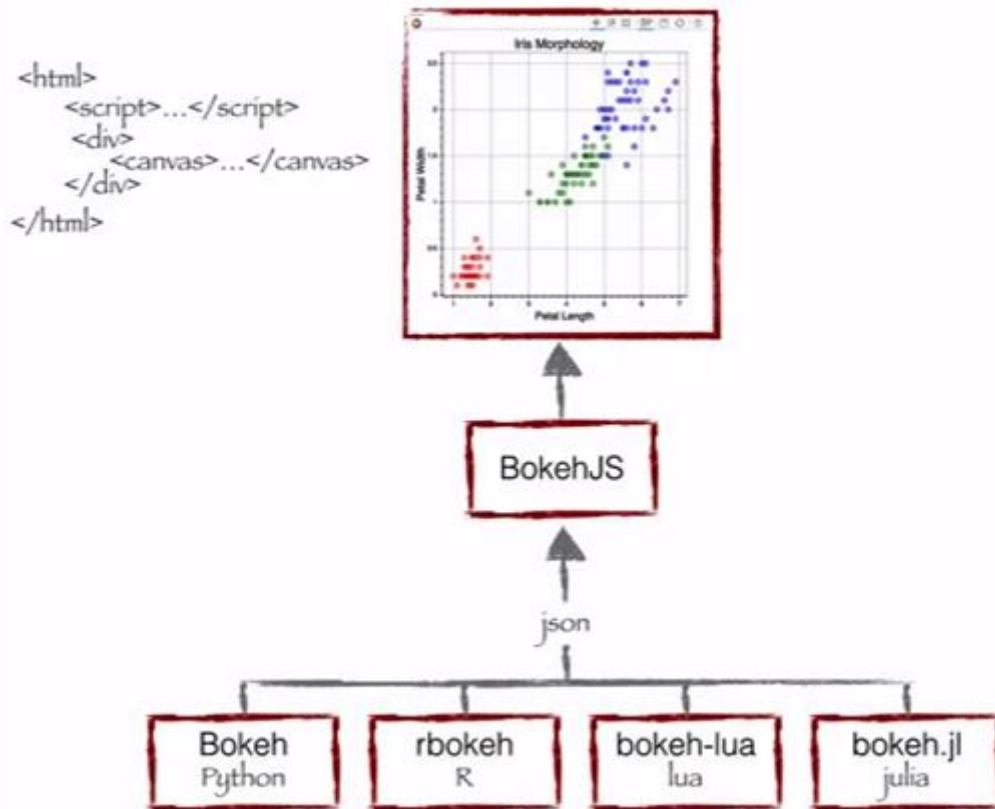


Fig 4 Bokeh

As you can see, Bokeh has multiple language bindings (Python, R, lua and Julia). These bindings produce a JSON file, which works as an input for BokehJS (a Javascript library), which in turn presents data to the modern web browsers.

Bokeh can produce elegant and interactive visualization like D3.js with high-performance interactivity over very large or streaming datasets. Bokeh can help anyone who would like to quickly and easily create interactive plots, dashboards, and data applications.

Advantages Of Bokeh

- Bokeh allows you to build complex statistical plots quickly and through simple commands
- Bokeh provides you output in various medium like html, notebook and server
- We can also embed Bokeh visualization to flask and django app
- Bokeh can transform visualization written in other libraries like matplotlib, seaborn, ggplot
- Bokeh has flexibility for applying interaction, layouts and different styling option to visualization

It is a high level interface used to present information in standard visualization form. These forms include box plot, bar chart, area plot, heat map, donut chart and many others. You can generate these plots just by passing data frames, numpy arrays and dictionaries.

The common methodology to create a chart:

1. Import the library and functions/ methods
2. Prepare the data
3. Set the output mode (Notebook, Web Browser or Server)
4. Create chart with styling option (if required)
5. Visualize the chart.

3.6 Diagram

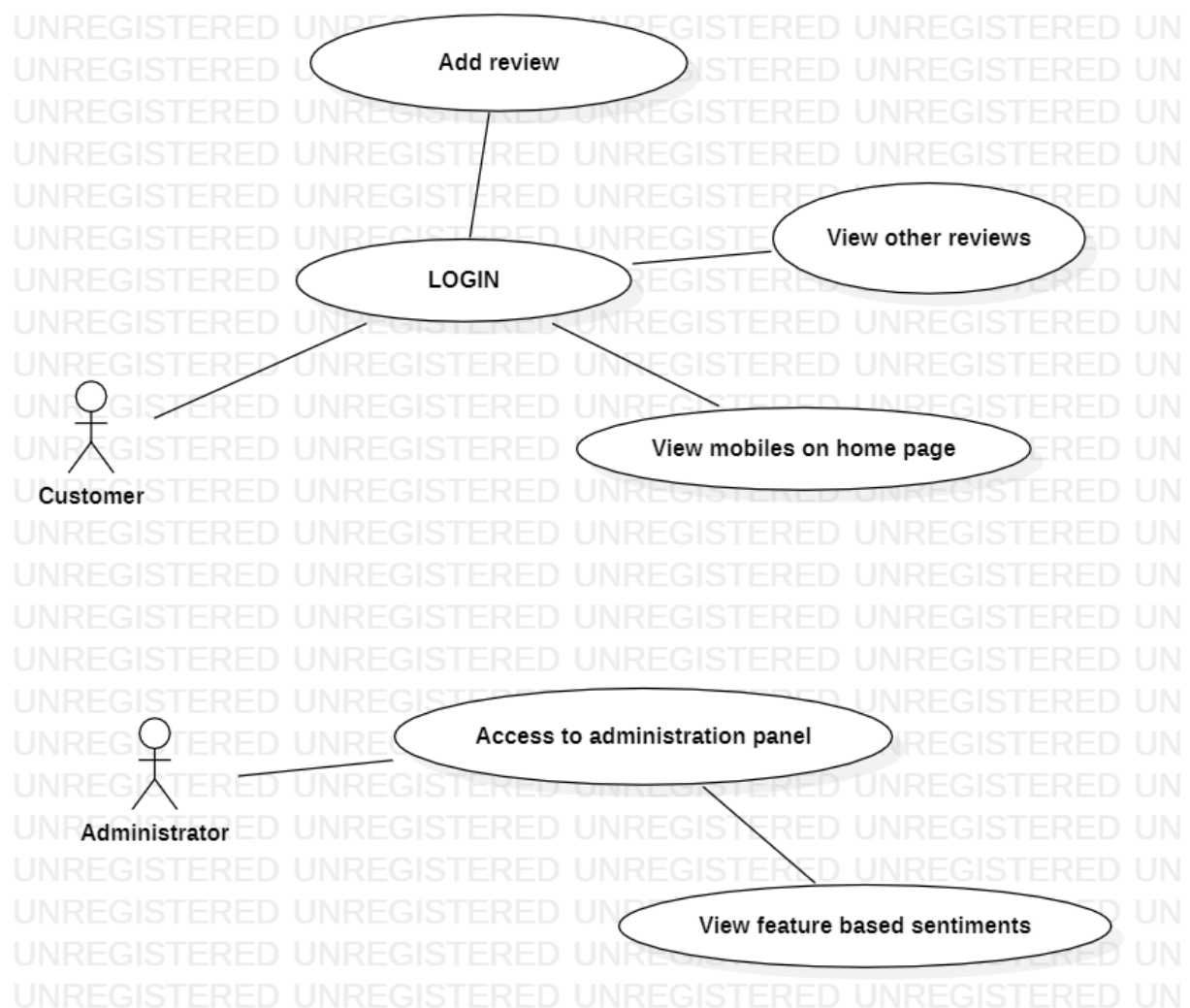


Fig 5. Use case diagram

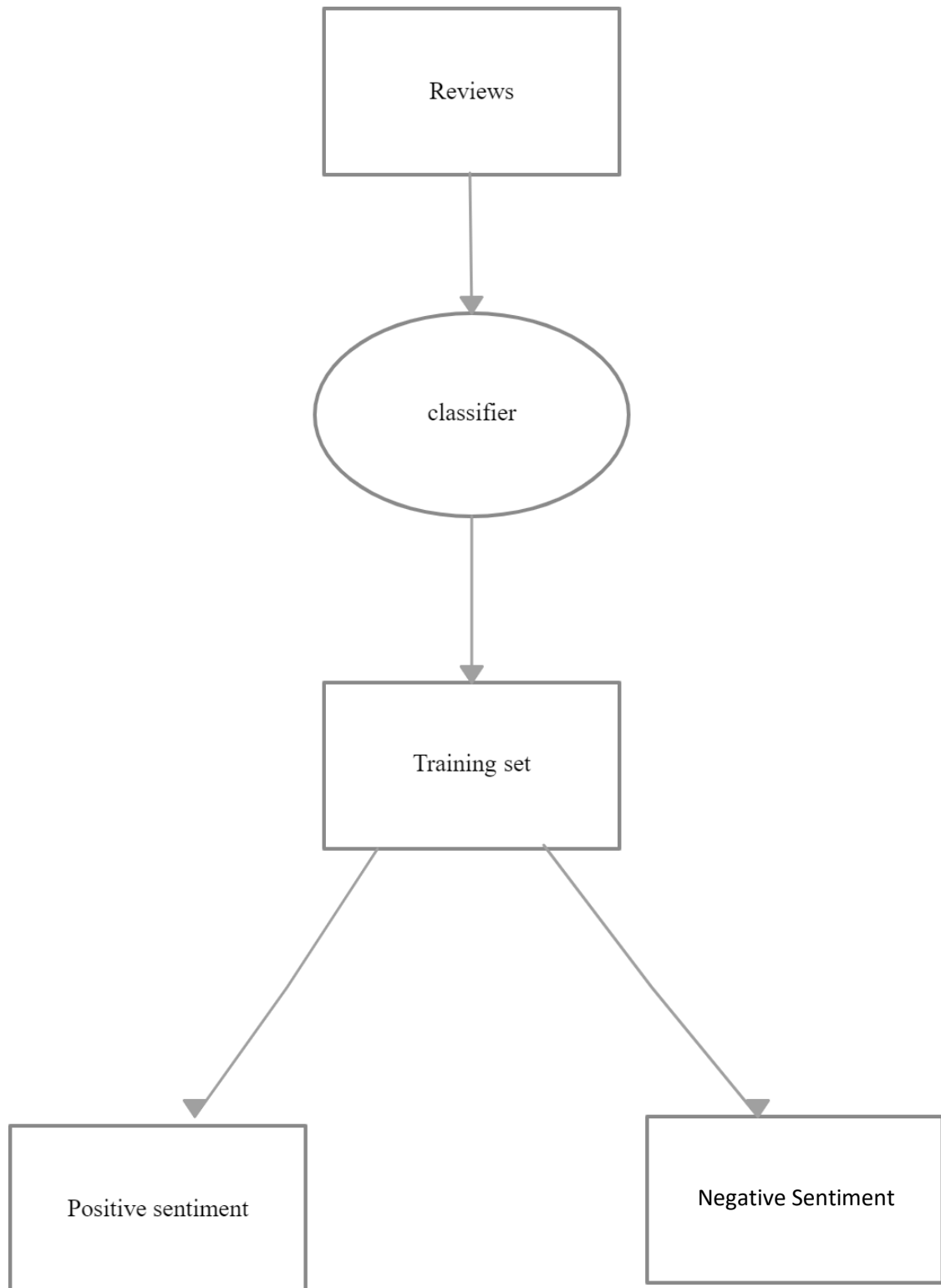


Fig 6 Basic Working of Machine Learning Algorithm

4. PERFORMANCE ANALYSIS

4.1 IMPLEMENTATION

4.1.1 Front-End Implementation

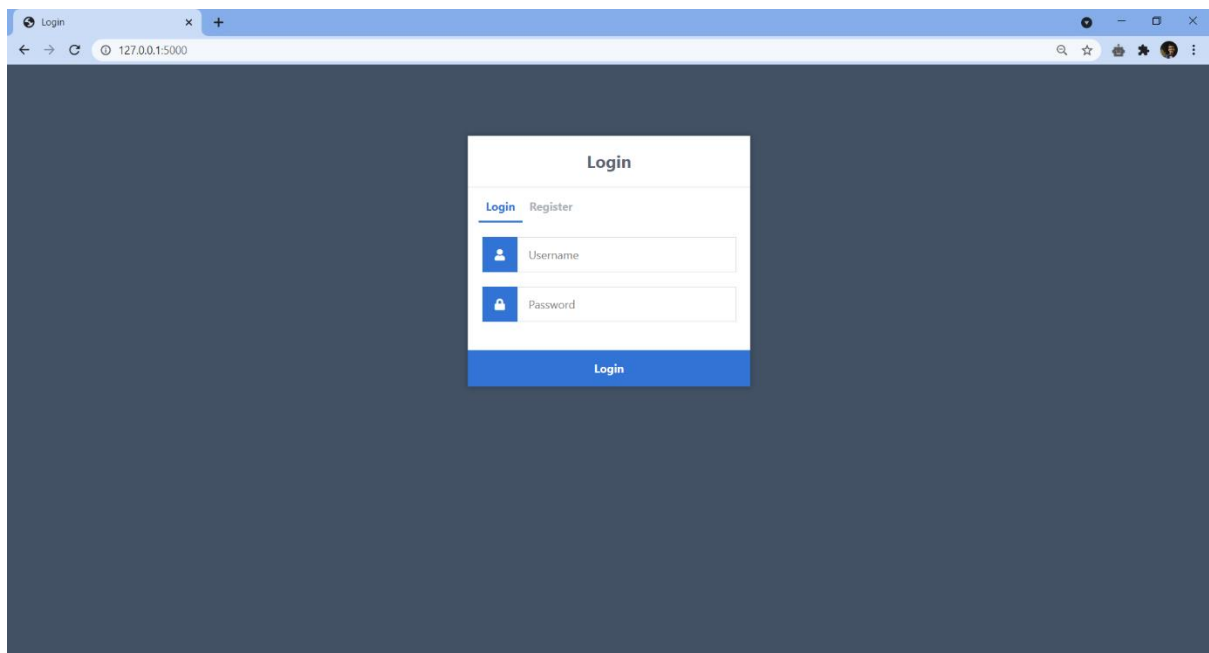


Figure 7 Login Page

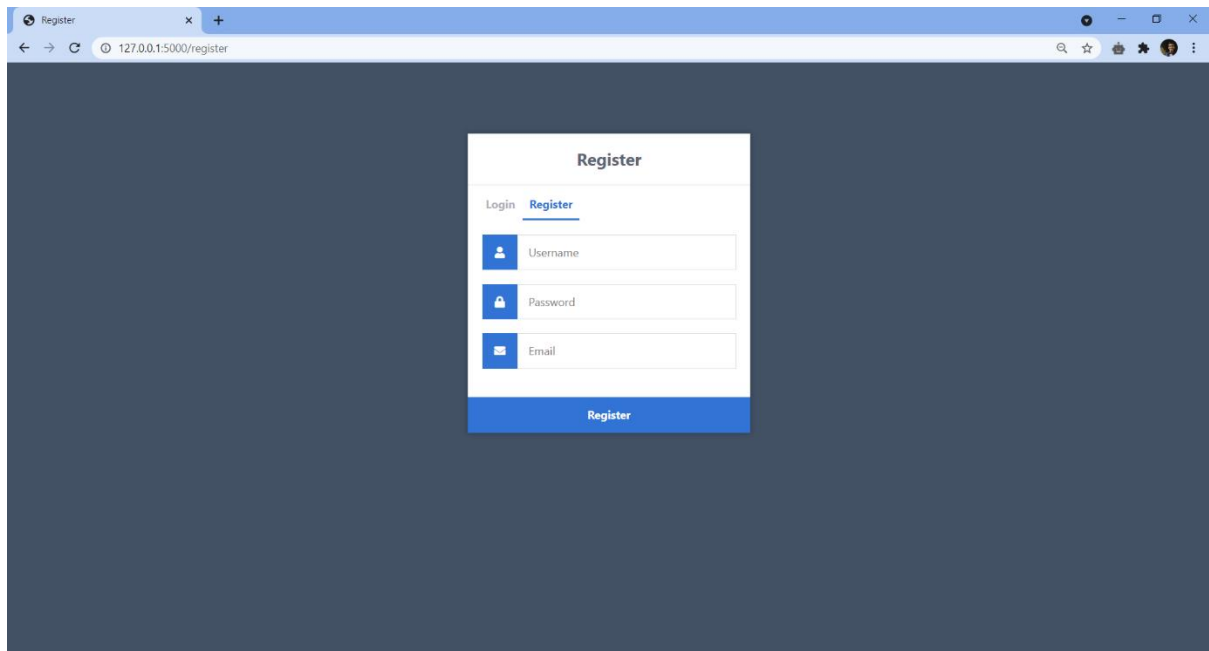


Figure 8. User Registration Page

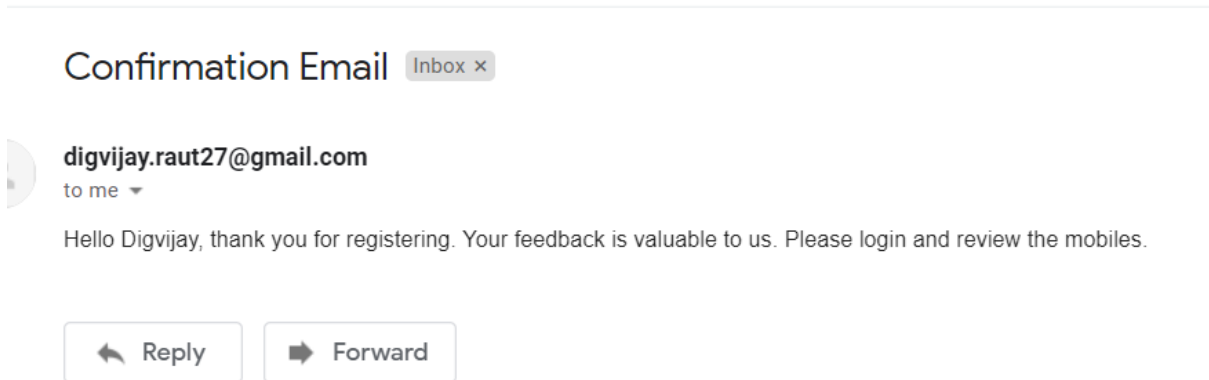


Figure 9 Confirmation Email

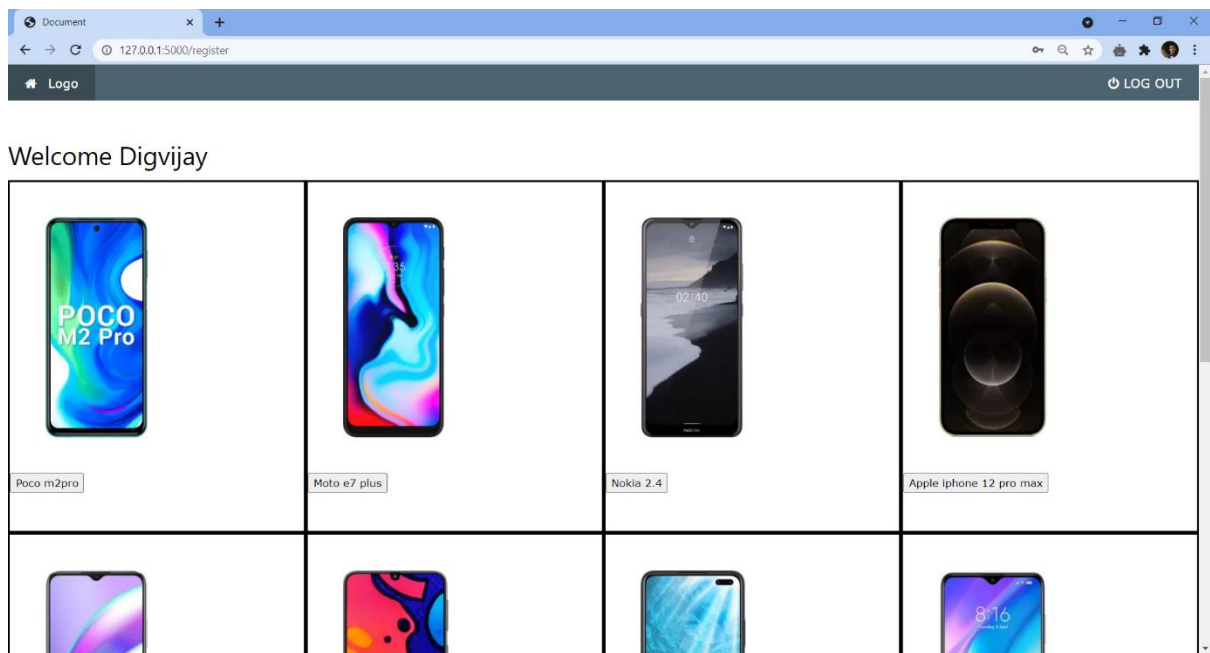


Figure 10. User login template

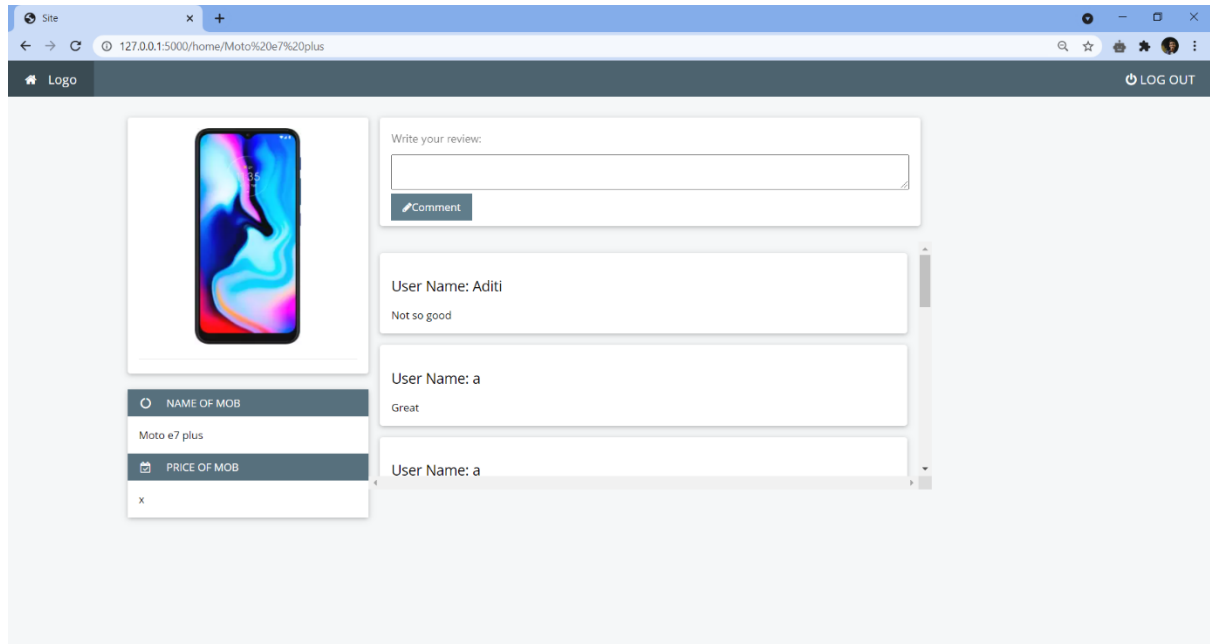


Figure 11. Mobile page

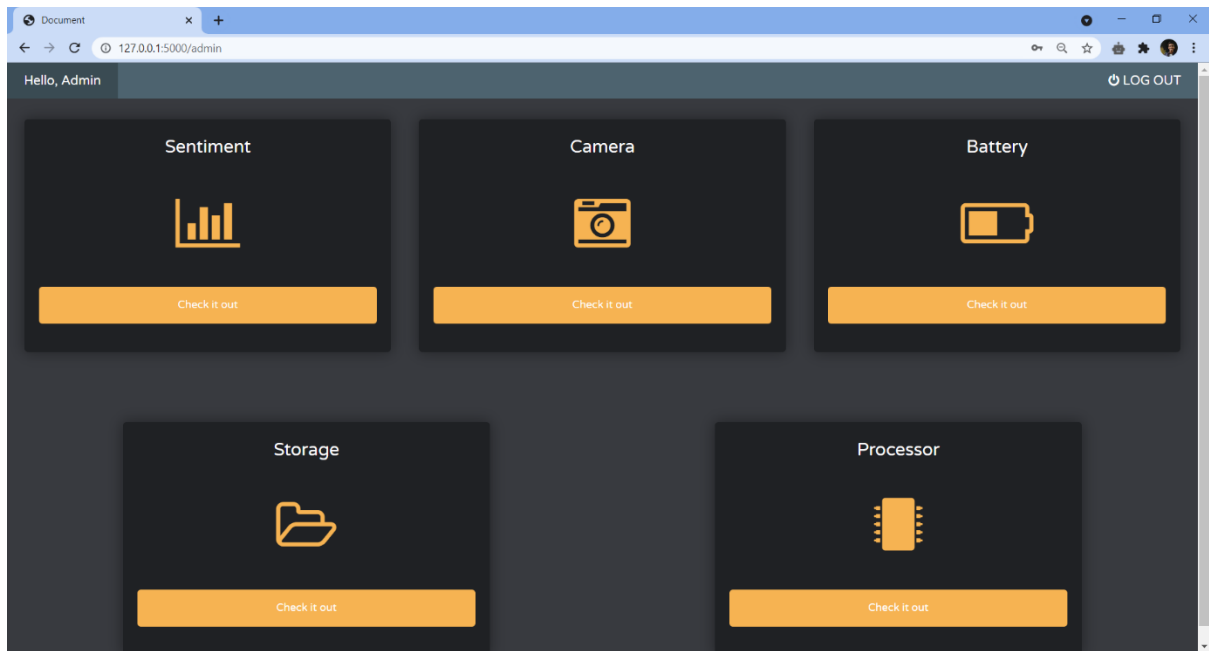


Figure 12. Admin panel

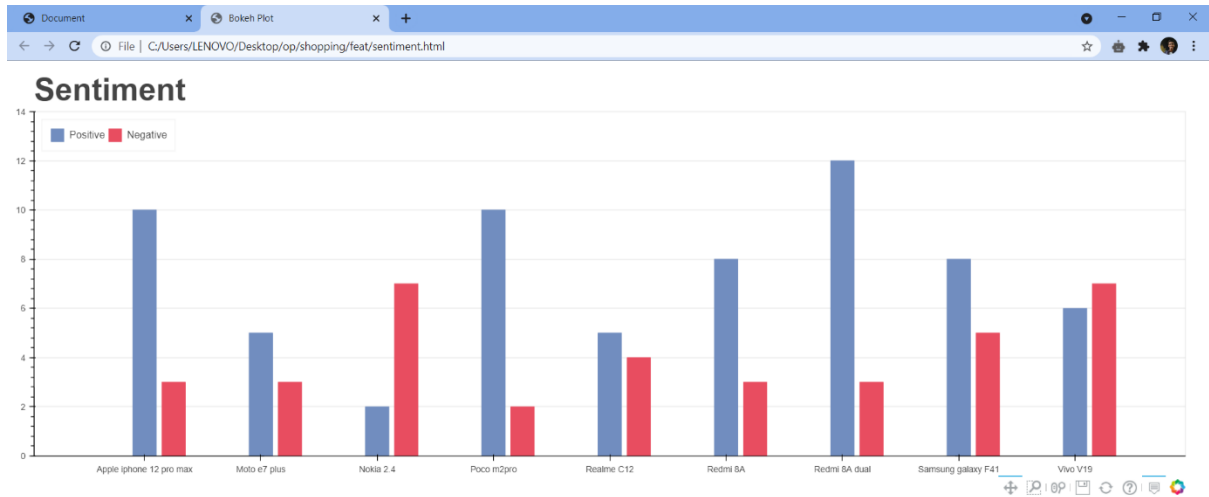


Figure 13. Admin – Reports1 – sentiment analysis

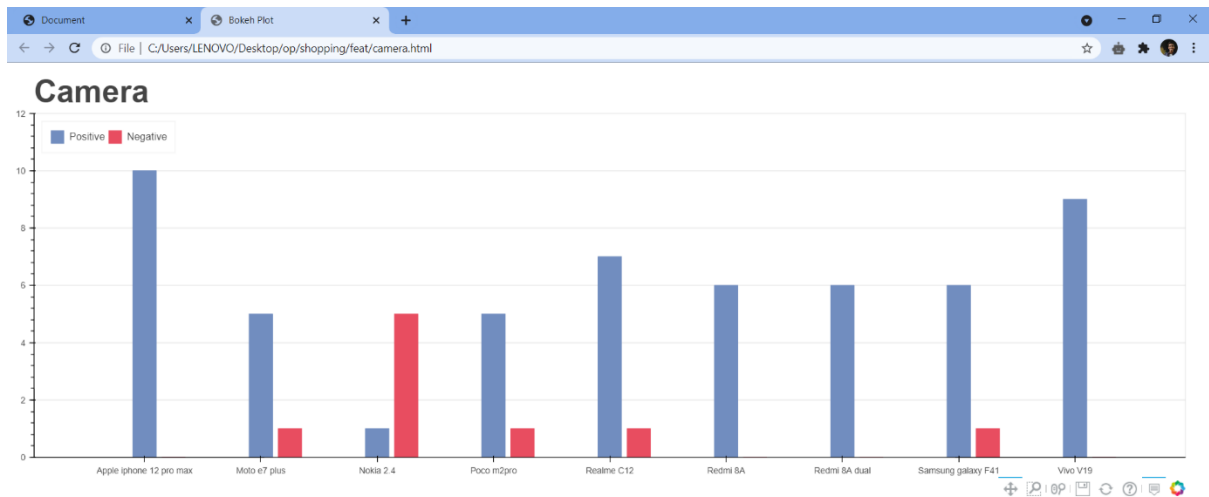


Figure 14. Admin – Reports2 – Camera sentiment analysis

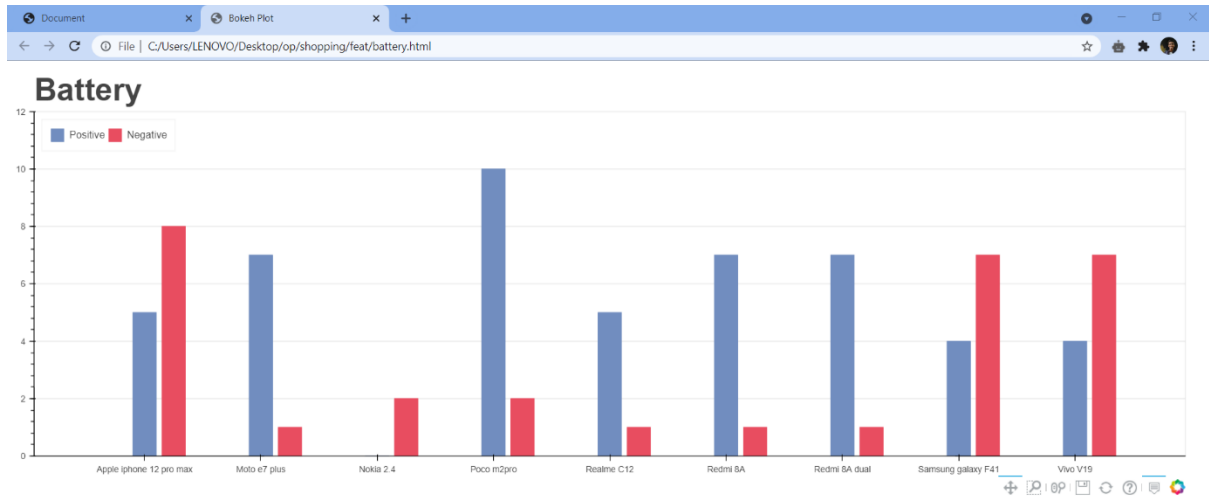


Figure 15 Admin – Reports3 – Battery sentiment analysis

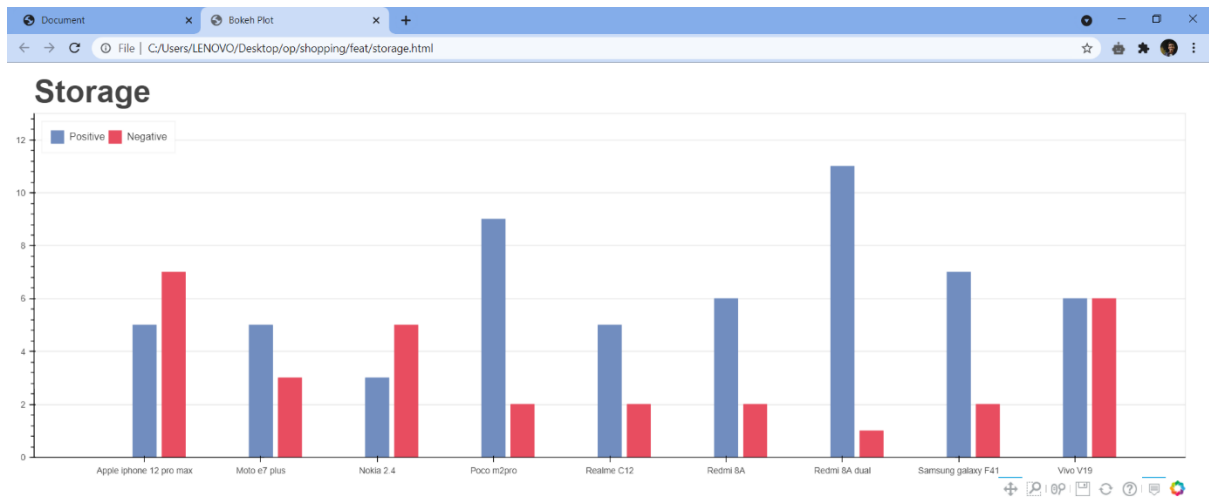


Figure 16. Admin – Reports4 – Storage sentiment analysis

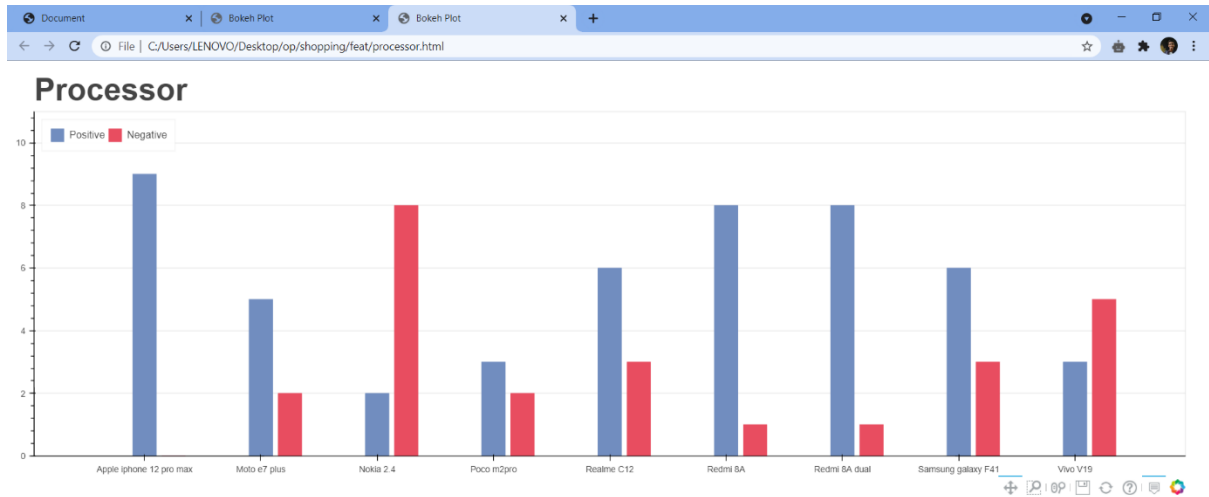


Figure 17 Admin – Reports5 – Processor sentiment analysis


```
Select C:\Windows\System32\cmd.exe - flask run
C:\Users\LENOVO\Desktop\op\shopping>flask run
* Environment: production
WARNING: This is a development server. Do not use it in a production deployment.
Use a production WSGI server instead.
* Debug mode: off
Total rows in db: 10
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
127.0.0.1 - - [14/Jun/2021 00:47:19] "[37mGET / HTTP/1.1+[0m" 200 -
127.0.0.1 - - [14/Jun/2021 00:47:19] "[36mGET /static/style.css HTTP/1.1+[0m" 304 -
127.0.0.1 - - [14/Jun/2021 00:47:20] "[33mGET /favicon.ico HTTP/1.1+[0m" 404 -
127.0.0.1 - - [14/Jun/2021 00:48:31] "[37mGET /register HTTP/1.1+[0m" 200 -
127.0.0.1 - - [14/Jun/2021 00:48:45] "[37mGET / HTTP/1.1+[0m" 200 -
127.0.0.1 - - [14/Jun/2021 00:48:48] "[37mGET /register HTTP/1.1+[0m" 200 -
No file
127.0.0.1 - - [14/Jun/2021 00:49:35] "[37mPOST /register HTTP/1.1+[0m" 200 -
127.0.0.1 - - [14/Jun/2021 00:50:03] "[32mGET /logout HTTP/1.1+[0m" 302 -
127.0.0.1 - - [14/Jun/2021 00:50:03] "[37mGET / HTTP/1.1+[0m" 200 -
127.0.0.1 - - [14/Jun/2021 00:50:19] "[32mPOST / HTTP/1.1+[0m" 302 -
127.0.0.1 - - [14/Jun/2021 00:50:19] "[37mGET /home HTTP/1.1+[0m" 200 -
127.0.0.1 - - [14/Jun/2021 00:50:25] "[37mGET /home/Moto%20e7%20plus HTTP/1.1+[0m" 200 -
127.0.0.1 - - [14/Jun/2021 00:51:31] "[37mGET /home HTTP/1.1+[0m" 200 -
127.0.0.1 - - [14/Jun/2021 00:51:33] "[32mGET /logout HTTP/1.1+[0m" 302 -
127.0.0.1 - - [14/Jun/2021 00:51:33] "[37mGET / HTTP/1.1+[0m" 200 -
127.0.0.1 - - [14/Jun/2021 00:51:40] "[32mPOST / HTTP/1.1+[0m" 302 -
127.0.0.1 - - [14/Jun/2021 00:51:41] "[37mGET /admin HTTP/1.1+[0m" 200 -
127.0.0.1 - - [14/Jun/2021 00:52:02] "[32mGET /admin/sentiment HTTP/1.1+[0m" 302 -
127.0.0.1 - - [14/Jun/2021 00:52:02] "[37mGET /admin HTTP/1.1+[0m" 200 -
127.0.0.1 - - [14/Jun/2021 00:52:22] "[32mGET /admin/camera HTTP/1.1+[0m" 302 -
127.0.0.1 - - [14/Jun/2021 00:52:22] "[37mGET /admin HTTP/1.1+[0m" 200 -
127.0.0.1 - - [14/Jun/2021 00:52:34] "[32mGET /admin/battery HTTP/1.1+[0m" 302 -
```

Figure 18. Local Server

| Product Name | Brand | Price | Rating | Reviews | Review Votes |
|---------------------|---------|--------|--------|---------------------------------------|--------------|
| "CLEAR CLE/ Samsung | Samsung | 199.99 | 5 | I feel so LUCKY to have found th | 1 |
| "CLEAR CLE/ Samsung | Samsung | 199.99 | 4 | nice phone, nice up grade from r | 0 |
| "CLEAR CLE/ Samsung | Samsung | 199.99 | 5 | Very pleased | 0 |
| "CLEAR CLE/ Samsung | Samsung | 199.99 | 4 | It works good but it goes slow s | 0 |
| "CLEAR CLE/ Samsung | Samsung | 199.99 | 4 | Great phone to replace my lost i | 0 |
| "CLEAR CLE/ Samsung | Samsung | 199.99 | 1 | I already had a phone with probl | 1 |
| "CLEAR CLE/ Samsung | Samsung | 199.99 | 2 | The charging port was loose. I g | 0 |
| "CLEAR CLE/ Samsung | Samsung | 199.99 | 2 | Phone looks good but wouldn't s | 0 |
| "CLEAR CLE/ Samsung | Samsung | 199.99 | 5 | I originally was using the Samsur | 0 |
| "CLEAR CLE/ Samsung | Samsung | 199.99 | 3 | It's battery life is great. It's very | 0 |
| "CLEAR CLE/ Samsung | Samsung | 199.99 | 3 | My fiance had this phone previo | 0 |
| "CLEAR CLE/ Samsung | Samsung | 199.99 | 5 | This is a great product it came al | 0 |
| "CLEAR CLE/ Samsung | Samsung | 199.99 | 5 | These guys are the best! I had a | 2 |
| "CLEAR CLE/ Samsung | Samsung | 199.99 | 1 | I'm really disappointed about my | 1 |
| "CLEAR CLE/ Samsung | Samsung | 199.99 | 5 | Ordered this phone as a replace | 0 |
| "CLEAR CLE/ Samsung | Samsung | 199.99 | 2 | Had this phone before and loved | 1 |
| "CLEAR CLE/ Samsung | Samsung | 199.99 | 5 | I was able to get the phone I pre | 6 |
| "CLEAR CLE/ Samsung | Samsung | 199.99 | 5 | I brought this phone as a replace | 0 |
| "CLEAR CLE/ Samsung | Samsung | 199.99 | 4 | I love the phone. It does everyth | 1 |
| "CLEAR CLE/ Samsung | Samsung | 199.99 | 3 | unfortunately Sprint could not a | 0 |
| "CLEAR CLE/ Samsung | Samsung | 199.99 | 4 | The battery was old & had been | 0 |
| "CLEAR CLE/ Samsung | Samsung | 199.99 | 4 | pros-beautiful screen,capable of | 0 |
| "CLEAR CLE/ Samsung | Samsung | 199.99 | 1 | I purchased this phone in Decem | 19 |
| "CLEAR CLE/ Samsung | Samsung | 199.99 | 4 | Phone good just a little slow ph | 0 |
| "CLEAR CLE/ Samsung | Samsung | 199.99 | 4 | Phone's speaker little low. Over | 0 |
| "CLEAR CLE/ Samsung | Samsung | 199.99 | 5 | the phone was great and in gooc | 0 |
| "CLEAR CLE/ Samsung | Samsung | 199.99 | 3 | the reasons for the 3 star rating | 0 |
| "CLEAR CLE/ Samsung | Samsung | 199.99 | 5 | Phone works great. No problem | 0 |

Figure 19. Train & test data set

```
In [20]: from sklearn.ensemble import RandomForestClassifier
model2 = RandomForestClassifier(n_estimators=300, max_features="auto")
model2.fit(X_train_vectorized,y_train)
predictions = model2.predict(vect.transform(X_test))
print('AUC: ',roc_auc_score(y_test,predictions))|

AUC: 0.867141692815005
```

Figure 20. Accuracy of Random Forest Classifier

```
In [15]: X_train_vectorized = vect.transform(X_train)

model = LogisticRegression()
model.fit(X_train_vectorized, y_train)

predictions = model.predict(vect.transform(X_test))

print('AUC: ', roc_auc_score(y_test, predictions))

AUC: 0.889951006492175
```

Figure 21 Accuracy of Logistic regression(TFIDF)

```
In [12]: from sklearn.metrics import roc_auc_score,roc_curve

#predict the transform test document.
predictions = model.predict(vect.transform(X_test))
print('AUC: ',roc_auc_score(y_test,predictions))

AUC: 0.8974332776669326
```

Figure 22 . Accuracy of Logistic regression(Count vectorizer)

```
In [17]: from sklearn.svm import SVC
from sklearn.metrics import roc_auc_score
model1 = SVC(kernel='linear', random_state=0)
model1.fit(X_train_vectorized, y_train)
predictions = model1.predict(vect.transform(X_test))
print('AUC: ', roc_auc_score(y_test, predictions))
```

AUC: 0.8975711995090037

Figure 23. Accuracy of SVC

```
In [20]: model = LogisticRegression()
model.fit(X_train_vectorized, y_train)

predictions = model.predict(vect.transform(X_test))

print('AUC: ', roc_auc_score(y_test, predictions))
```

C:\Users\LENOVO\anaconda3\lib\site-packages\sklearn\linear_model_logistic.py:763: ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
<https://scikit-learn.org/stable/modules/preprocessing.html>
Please also refer to the documentation for alternative solver options:
https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
n_iter_i = _check_optimize_result(

AUC: 0.9104640361714084

Figure 24 . Accuracy of Logistic Regression(N-gram)

```
feature_names = np.array(vect.get_feature_names())

sorted_coef_index = model.coef_[0].argsort()

print('Smallest Coefs:\n{}\n'.format(feature_names[sorted_coef_index[:10]]))
print('Largest Coefs: \n{}\n'.format(feature_names[sorted_coef_index[:-11:-1]]))
```

Smallest Coefs:
['no good' 'junk' 'poor' 'slow' 'worst' 'broken' 'not good' 'terrible'
'defective' 'horrible']

Largest Coefs:
['excellent' 'excelente' 'perfect' 'excelent' 'great' 'love' 'awesome'
'no problems' 'good' 'best']

Figure 25. Feature names

4.1.2 DATABASE Implementation

Database Name: peace

```
mysql> desc logins;
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| id         | int(100)      | NO   | PRI | NULL    | auto_increment |
| Name      | varchar(255)  | NO   |     | NULL    |                |
| email     | varchar(255)  | NO   |     | NULL    |                |
| password  | varchar(255)  | NO   |     | NULL    |                |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

Figure 26. Users Table description

```
mysql> desc comments;
```

| Field | Type | Null | Key | Default | Extra |
|-----------|--------------|------|-----|---------|----------------|
| id | int(11) | NO | PRI | NULL | auto_increment |
| name | varchar(255) | YES | | NULL | |
| commant | varchar(450) | YES | | NULL | |
| user_name | varchar(255) | YES | | NULL | |

```
4 rows in set (0.01 sec)
```

Figure 27. comments table description

```
mysql> desc positive;
```

| Field | Type | Null | Key | Default | Extra |
|-----------|--------------|------|-----|---------|-------|
| sentiment | int(20) | YES | | NULL | |
| camera | int(20) | YES | | NULL | |
| battery | int(20) | YES | | NULL | |
| storage | int(20) | YES | | NULL | |
| processor | int(20) | YES | | NULL | |
| name | varchar(255) | YES | | NULL | |

```
6 rows in set (0.02 sec)
```

Figure 28. Positive Table description

```
mysql> desc negative;
```

| Field | Type | Null | Key | Default | Extra |
|-----------|--------------|------|-----|---------|-------|
| sentiment | int(20) | YES | | NULL | |
| camera | int(20) | YES | | NULL | |
| battery | int(20) | YES | | NULL | |
| storage | int(20) | YES | | NULL | |
| processor | int(20) | YES | | NULL | |
| name | varchar(255) | YES | | NULL | |

6 rows in set (0.00 sec)

Figure 29. Negative Table description

```
mysql> select * from positive;
```

| sentiment | camera | battery | storage | processor | name |
|-----------|--------|---------|---------|-----------|-------------------------|
| 10 | 5 | 10 | 9 | 3 | Poco m2pro |
| 5 | 5 | 7 | 5 | 5 | Moto e7 plus |
| 2 | 1 | 0 | 3 | 2 | Nokia 2.4 |
| 10 | 10 | 5 | 5 | 9 | Apple iphone 12 pro max |
| 5 | 7 | 5 | 5 | 6 | Realme C12 |
| 8 | 6 | 4 | 7 | 6 | Samsung galaxy F41 |
| 6 | 9 | 4 | 6 | 3 | Vivo V19 |
| 12 | 6 | 7 | 11 | 8 | Redmi 8A dual |
| 8 | 6 | 7 | 6 | 8 | Redmi 8A |

9 rows in set (0.00 sec)

Figure 30. Positive sentiment Table

```
mysql> select * from negative;
```

| sentiment | camera | battery | storage | processor | name |
|-----------|--------|---------|---------|-----------|-------------------------|
| 3 | 0 | 1 | 2 | 1 | Redmi 8A |
| 3 | 0 | 1 | 1 | 1 | Redmi 8A dual |
| 7 | 0 | 7 | 6 | 5 | Vivo V19 |
| 5 | 1 | 7 | 2 | 3 | Samsung galaxy F41 |
| 4 | 1 | 1 | 2 | 3 | Realme C12 |
| 3 | 0 | 8 | 7 | 0 | Apple iphone 12 pro max |
| 7 | 5 | 2 | 5 | 8 | Nokia 2.4 |
| 3 | 1 | 1 | 3 | 2 | Moto e7 plus |
| 2 | 1 | 2 | 2 | 2 | Poco m2pro |

9 rows in set (0.00 sec)

Figure 31 Negative sentiment Table

| | A | B | C |
|----|-------------------------|---------------------|-------|
| 1 | Names | Images | Price |
| 2 | Poco m2pro | https://rukminim1.x | |
| 3 | Moto e7 plus | https://rukminim1.x | |
| 4 | Nokia 2.4 | https://rukminim1.x | |
| 5 | Apple iphone 12 pro max | https://rukminim1.x | |
| 6 | Realme C12 | https://rukminim1.x | |
| 7 | Samsung galaxy F41 | https://rukminim1.x | |
| 8 | Vivo V19 | https://rukminim1.x | |
| 9 | Redmi 8A dual | https://rukminim1.x | |
| 10 | Redmi 9A | https://rukminim1.x | |
| 11 | | | |
| 12 | | | |
| 13 | | | |
| 14 | | | |
| 15 | | | |
| 16 | | | |
| 17 | | | |
| 18 | | | |
| 19 | | | |
| 20 | | | |
| 21 | | | |
| 22 | | | |
| 23 | | | |
| 24 | | | |
| 25 | | | |
| 26 | | | |
| 27 | | | |
| 28 | | | |
| 29 | | | |

Figure 32. Products table

5. CONCLUSIONS

5.1 CONCLUSIONS

Sentimental analysis(also known as opinion mining is a process of determining the emotional tone behind a series of words, used to gain an understanding of the attitudes, opinions and emotions expressed within an online mention. It uses various techniques and algorithms so as to get maximum accuracy and hence best results. This project has undergone using a number of algorithms and finally reaching at the suitable process of Logistic Regression with a proper dataset and parameters. We get an accuracy of around 90%. Sentiment analysis is an incredibly valuable technology for businesses because it allows getting realistic feedback from your customers in an unbiased (or less biased) way. Done right, it can be a great value-added to your systems, apps, or web projects. This project helps the market to know about the customer feedback on various mobile phones. Customers have various expectations as mobile phones are an integral part of life and has evolved as a device a lot. Camera, Processor, battery , RAM are some of the features on which our project shows the opinions of their buyers. This helps the market to plan further workflow of management, get a general perception of project since this acts as an efficient way of market research. The voice of customer is projected easily and hence helps the business to grow. Such technologies can help every business and owner to improve business and customer engagement. This has a good scope in future to become better and bring out best uses.

5.2 FUTURE SCOPE

Considering the current issues and challenges of this project, it has a wide scope to extend and improve. This project currently focused on dealing with only reviews in English language. However, it can be extended to work as multilingual. In a market, there is customer engagement worldwide. Hence, this can be developed in such a way so as to be working on ever customer's native language too.

Also, in today's world , it is seen that the online world has its own language in the form of various short forms and slangs. With proper dataset, we can also try to train our model to fit in the evolving world. This needs more time and appropriate data because the trends keep changing. There is scope for improvement even in the accuracy if possible using a variety of

algorithms and a larger dataset. This project can be also further developed so that the business owners gets to know about each brand in detail and hence predict the business strategy.

5.3 APPLICATION

As already mentioned, this project is used for

Market Research

Customer engagement

Customer service

Monitoring

REFERENCES

- [1] <https://www.javatpoint.com/machine-learning-support-vector-machine-algorithm>
- [2] https://www.tutorialspoint.com/machine_learning_with_python/machine_learning_with_python_classification_algorithms_random_forest.htm.
- [3] Sentiment Analysis Of Product Reviews – A Survey Dishu Jain, Bitra Harsha Vardhan, Saravanakumar Kandasamy.
- [4] A Literature Review on Application of Sentiment Analysis Using Machine Learning Techniques.
https://www.researchgate.net/publication/343736541_A_Literature_Review_on_Application_of_Sentiment_Analysis_Using_Machine_Learning_Techniques

APPENDIX : How to run the project

This is Sentiment analysis developed using the latest flask framework.

To deploy this application

- 1) Import the project folder to your PC.
- 2) Create a database
- 3) Import the SQL file into your database
- 4) Open cmd in the folder where you have app.py file
- 5) Change the following values to match your database settings:

`'hostname' = 'localhost'`

`'username' = 'your database username',`

`'password' = 'your database password',`

`'database' = 'Your database name'`
- 6) Type flask run in cmd.
- 7) Open your web browser and navigate to <http://127.0.0.1:5000/>.