GOOGLE PLAYSTORE REVIEW SENTIMENT ANALYSIS

Vaishnavi Dudhankar*1, Nitu Sen*2, Atmja Langde*3, Vaishnavi Kupade4

*1,2,3,4Bramhadevdada Mane Institude Of Technology, Belati, Solapur, India.

ABSTRACT

Our day-to-day life has always been influenced by what people think. Ideas and opinions of others have always affected our own opinions. As the Web plays an increasingly significant role in people's social lives, it contains more and more information concerning their opinions and sentiments. The distillation of knowledge from this huge amount of unstructured information, is also known as opinion mining and sentiment analysis. Nowadays, with the rapid evolution of smart phones, mobile applications (Mobile Apps) have become essential parts of our lives. However, it is difficult for consumers to keep track and understand the app sphere because new apps are entering market every day. So, sentiment analysis of application reviews on Google play store will help the developers of the applications to keep their particular applications up to date in order to keep their particular application in the top lists and also help the customers to select the most popular application. Our proposed architecture will show this scheme had good scale-out scalability and the upper limit of the total number of files that the system was able to store could be greatly increased by adding a number of Name Nodes.

Keywords: Mobile Apps, Reviews, Ratings, Sentiment Analysis, Learning Approaches And Polarity.

I.   INTRODUCTION

Introduction Sentiment analysis is the process of classifying whether a block of text is positive, negative, or, neutral. Sentiment analysis is contextual mining of words which indicates the social sentiment of a brand and also helps the business to determine whether the product which they are manufacturing is going to make a demand in the market or not. The goal which Sentiment analysis tries to gain is to analyses people's opinion in a way that it can also help the businesses expand. It focuses not only on polarity (positive, negative & neutral) but also on emotions (happy, sad, angry, etc.). Sentiment analysis is a technique with which we can identify and determine if data indicates a positive, negative or neutral emotion. It aids in understanding customer feedback and is mostly used by several companies to analyses brand and product reviews. It also helps in finding out the underlying sentiment in a text. There are different types of sentiment analysis and in each type, information can be extracted in different ways. Posting reviews online has become an increasingly popular way for people to express opinions and sentiments towards the products bought or services received. Analyzing the large value of online reviews would produce useful actionable knowledge that could be of economic values to vendors and other interested parties. Sentiment Analysis is the process of computationally determining whether a piece of writing is positive, negative or neutral and deriving the attitude of a speaker. Sentiment analysis is the computational study of people's opinions, attitudes and emotions toward an entity. The entity can represent individuals, events or topics. These topics are most likely to be covered by reviews. Generally speaking, sentiment analysis aims to determine the attitude of a speaker or a writer with respect to some topic or the overall contextual polarity of a document. The attitude may be his or her judgment or evaluation, affective state that is to say, the emotional state of the author when writing, or the intended emotional communication that is to say, the emotional effect the author wishes to have on the reader. Everything the project will be expected to do the earlier stage will comprise of the extraction of the user reviews for the purpose of analysis on it. The user reviews will consist of the text feedback. The sentiment analysis will be carried on which will create polarity for each text review generated. The polarity calculated from these reviews will help the developer to for future evaluation and any changes to their application. The developer of the application may be keen to know -What do people think about the particular application? How is the overall impact of the application whether it's positive or negative? What are people suggestions for improving the application or what does people prefer the future update for the application should be like? The main need to collect opinions for particular applications and draw conclusions is to be aware that what are the people like/dislike, is been the most. Sentiment Analysis identifies the sentiment expressed in a text then analyses it. Therefore, the target of Sentiment Analysis is to find opinions, identify the sentiments they express, and then classify their polarity as shown in Figure 1.1
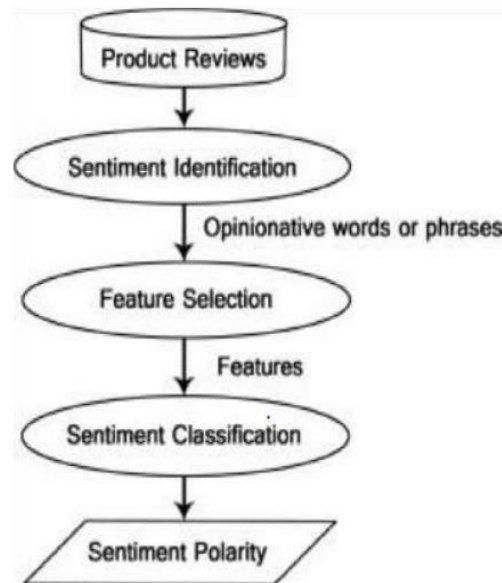
Figure 1: Sentiment analysis process on product reviews.

Main classification levels in Sentiment Analysis: Document-level, Sentence-level, and Aspect-level Sentiment Analysis. Document-level Sentiment Analysis aims to classify a Sentiment document as expressing a positive, negative or neutral sentiment. It considers the whole document a basic information unit. Sentence-level Sentiment Analysis aims to classify sentiment expressed in each sentence. The first step is to identify whether the sentence is subjective or objective. If the sentence is subjective, Sentence-level Sentiment Analysis will determine whether the sentence expresses positive negative or neutral reviews. Have pointed out that sentiment expressions are not necessarily subjective in nature. However, there is no fundamental difference between document and sentence level classifications because sentences are just short documents. Classifying text at the document level or at the sentence level does not provide the necessary detail needed Sentiment on all aspects of the entity which is needed in many applications, to obtain these details; we need to go to the aspect level. Aspect-level Sentiment Analysis aims to classify the sentiment with respect to the specific aspects of entities. The first step is to identify the entities and their aspects. The holders can give different opinions for different aspects of the same entity like this sentence "The voice quality of this phone is not good, but the battery life is long". This survey tackles the first two kinds of Sentiment Analysis. For example- If we want to analyses whether a product is satisfying customer requirements, or is there a need for this product in the market? We can use sentiment analysis to monitor that product's reviews. Sentiment analysis is also efficient to use when there is a large set of unstructured data, and we want to classify that data by automatically tagging it.Net Promoter Score (NPS) surveys are used extensively to gain knowledge of how a customer perceives a product or service. Sentiment analysis also gained its popularity due to its feature to process large volumes of NPS responses and obtain consistent results quickly.

Problem Definition:

This project aims to extract the features of play store reviews and analyses the opinion of play store app reviews. Sentiment Analysis is classifying the polarity of given text in a document in a sentence is positive, negative or neutral. To collecting data and categorizing into different sets for different purposes. Making sentiment tool for measuring all sentiments by one tool. Increase the accuracy of the result which is measured by sentiment tool.

Overall Scenario:

Important aspect in scenario, in this project, we try to implement a Google play store sentiment analysis model that helps to overcome the challenges of identifying the sentiment of the Google play store apps reviews. The objective of this paper is to discuss concept of sentiment analysis of application reviews on Google play store and perform comparative study of its various techniques.

Requirement Specifications:

Software requirements:

OS- Windows / Linux, Visual Studio Code, Python 3, XAMPP Control Panel, Browser (Google, Mozilla Firefox, Explorer… etc)

Hardware requirements:

System 32/64 bit, HDD – 1 TB, RAM- Minimum 4 GB required.

II.   LITERATURE REVIEW

There has been a constant growth in the public and private information stored within the internet. This includes textual data expressing people's opinions on review sites, forums, blogs, and other social media platforms. Review based prediction systems allow this unstructured information to be automatically transformed into structured data reflecting public opinion. These structured data can be used subsequently as a measure of users' sentiments about specific applications, products, services, and brands. They can hence provide important information for product and services refinement. This kind of sentiment analysis was conducted in the following studies.

Kumara and other researchers [8, 9, 10, and 5] used the Naïve Bayes (NB) classifier to classify opinions as positive, negative, or neutral. Wang and others [11] argued that a rating is not entirely determined by a review content. For example, a user may well intend to give a positive review by employing positive words, and yet issue a comparatively lower rating.

Dave and others [12] proposed a method for extracting the polarity in user reviews of products, expressed as poor, mixed, or good. The classifier used was Naïve Bayes (NB). According to Pang et al [13], although machine learning approaches perform far better for traditional topic-based categorization, they're less successful for sentiment analysis. Information extraction technologies have also been exploring do identify and organize opinions contained.

A recent study [17] investigated the application of a machine learning algorithm to dataset covering, for example, the app category, the numbers of reviews and downloads, the size, type, and Android version of an app, and the content rating, to predict a Google app ranking. Decision trees, linear regression, logistic regression, support-vector machine, NB classifiers, means clustering, k-nearest neighbors, and artificial neural networks were studied for that purpose.

Other authors [15] suggested adopting a statistical analysis based on a spin model, to extract the semantic orientations of words. Mean field approximations were used to compute the approximate probability in the spin model. Semantic orientations are then evaluated as desirable or undesirable. A smaller number of seed words for the proposed model produce highly accurate semantic orientations based on the English lexicon.

In contrast to the above-cited studies, other authors [20] investigated the nature of sentiments expressed in Google app

However, the above studies are unsatisfactory in various respects and are unsuitable for predicting numeric ratings of Google apps. First, text-mining techniques are ineffective when applied to app reviews, as it has Unicode supported language with a limited number of words. Second, those studies are based either on rating predictions made using inherent app features or on external features (e.g., price, bug report, etc.). None of those studies investigated the possible discrepancies between users' numeric ratings and reviews. To our knowledge, this study is the first to investigate such discrepancies and to base numeric-rating predictions for Google apps.

Software Development Life Cycle:

A development process consists of various phases, each phase ending with a defined output. A phased process allows proper checking for quality and progress at some defined points during the development. Problem solving in software most consist of requirement specification for understanding and clearly stating the problem, design for deciding a plan for a solution, coding for implementing the planed solution, and testing for verifying the programs (Fig.2).
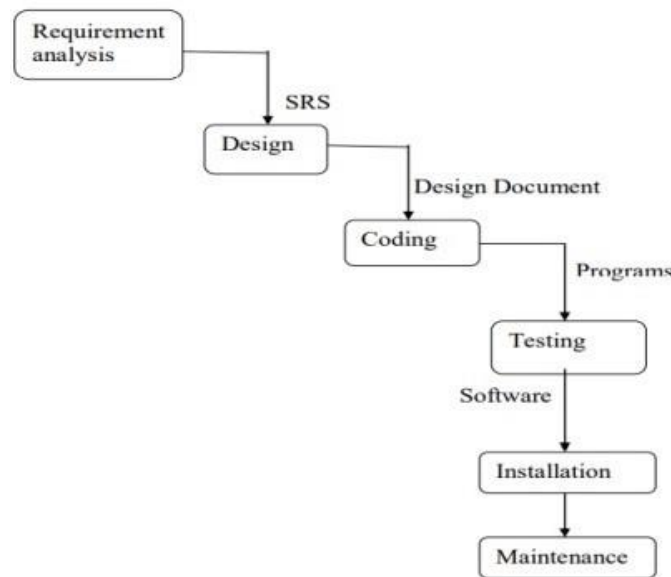
Fig 2: Software development life cycle

III.   SYSTEM DESIGN

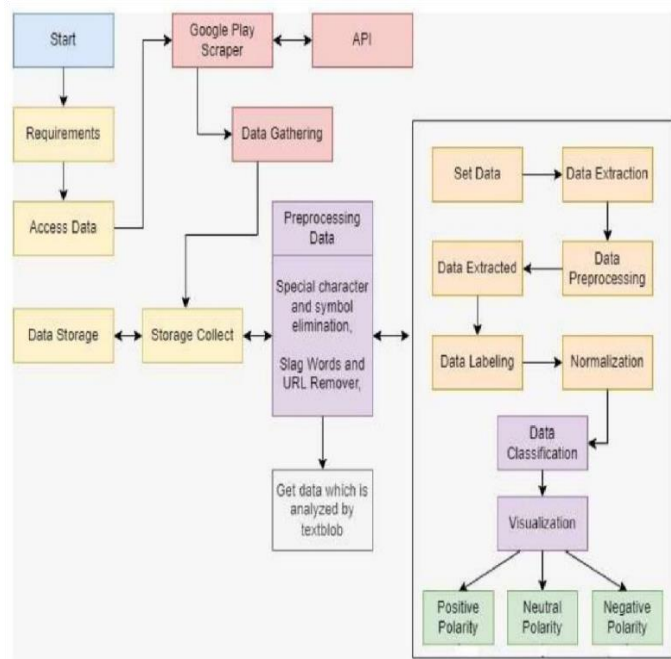**3.1 System Architecture:**

Figure 3: System Architecture

IV.   SOFTWARE DEVELOPMENT APPROACH

In software development, the focus is primarily on processes, which were referred to as systematic approach. The systematic approach is given by the software development life cycle.

Requirement's analysis is done in order to understand the problem the software system is to solve. The emphasis in requirements analysis is on identifying what is needed from the system, not how the system will achieve its goals. The goal of the requirements activity is to document the requirements in a software requirements specification document.

The purpose for the design phase is to plan a solution of the problem specified by the requirement document. The design activity often results in three separate outputs- architecture design, high level design, and detailed design. Architecture focuses on looking at a system as a combination of many different components, and how

they interact with each other to produce desired results. The high-level design identifies the modules that should be built for developing the system and the specification of these modules. In detailed design, the internal logic of the modules is specified.

The goal of the coding phase is to translate the design of the system into code in a given programming language. The coding phase affects both testing and maintenance profoundly. Well-written code can reduce the testing and maintenance effort.

Testing is the major quality control measure used during software development. Its basic function is to detect defects in the software.

Software Development:

In the software process we focus on the activities directly related to production of the software, for example, design, coding, and testing. Development process specifies the major development and quality control activities. Due to the importance of the development process, various models have been proposed.

For our project we are using the Iterative development model. Its basic idea is that the software should be developed in increments, each increment adding some functional capability to the system until the full system is implemented. At each step, extensions & design modifications can be made. An advantage of this approach is that it can result in better testing because testing each increment is likely to be easier than testing the entire system. Fig 3 shows the Iterative model.
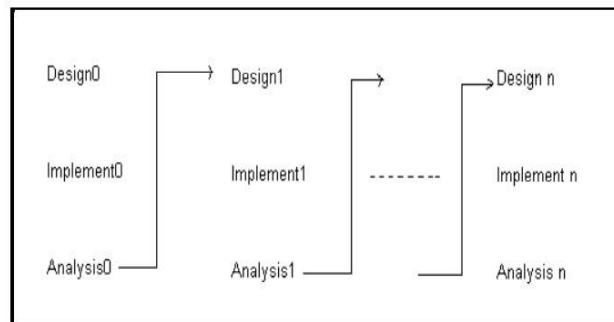

Fig 4: Iterative model

Testing:

As the basic goal of the software development process is to produce software that has no errors or very few errors. To have the few errors testing is performed at the end of each phase. Here we have tested the functionality of each of the model & component. Testing is usually relied upon to detect the faults remaining from earlier stages. Fig 4 shows the different levels of testing, which are described below:
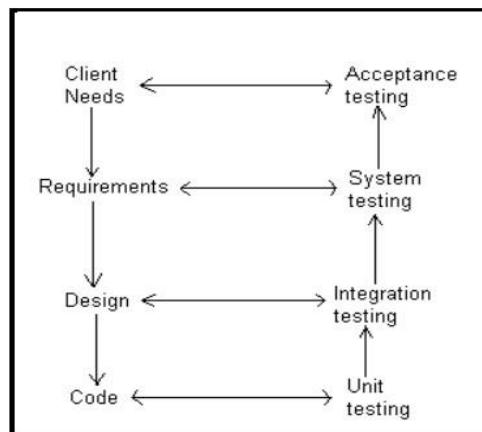

Fig 5: Levels of testing

Unit Testing:

In this testing different modules are tested against the specifications produced during design for the modules. Unit testing is essentially for verification of the code produced during the coding phase, and hence the goal is to

test the internal logic of the modules. A module is considered for integration and use by others only after it has been unit tested satisfactorily. In this testing, we have tested individual component in the system.

Integration Testing:

The next level of testing is the integration testing. In this, many units tested modules are combined into subsystems, which are then tested. The goal here is to see if the modules can be integrated properly. Hence, the emphasis is on testing interfaces between modules. Here we have integrated all the component of the system & then entire system is tested.

Validation Testing:

Here the entire software system is tested. The reference document for this process is the requirement document, and goal is to see if the software meets its requirements. This is essentially a validation exercise, and in many situations, it is the only validation activity. By considering the system architecture, we have checked whether the all requirements are satisfied or not.

High-Order Testing:

High order testing is sometimes performed with realistic data of the client to demonstrate that the software is working satisfactorily.

V.   TECHNOLOGY & LANGUAGE USED

In our project we are going to use Machine Learning technology and Python language.

Machine Learning:

Machine learning is a subfield of artificial intelligence (AI). The goal of machine learning generally is to understand the structure of data and fit that data into models that can be understood and utilized by people.

Although machine learning is a field within computer science, it differs from traditional computational approaches. In traditional computing, algorithms are sets of explicitly programmed instructions used by computers to calculate or problem solve. Machine learning algorithms instead allow for computers to train on data inputs and use statistical analysis in order to output values that fall within a specific range. Because of this, machine learning facilitates computers in building models from sample data in order to automate decision-making processes based on data inputs.

The goal of machine learning technology is to optimize the performance of a system when handling new instances of data through user defined programming logic for a given environment. To accomplish this goal effectively and efficiently, machine learning draws heavily on statistics and computer science.

Statistical methods provide machine learning algorithms ways to infer conclusions from data. Computer science methods, on the other hand, give machine learning algorithms (MLAs) the computing power to solve problems, including useful large-scale computational architectures and algorithms for capturing, manipulating, indexing, combining and performing predictions with data.

The objective of the present research work is to make the reader understand the importance of MLAs in daily life. This will help the readers get familiar with applying MLAs when addressing classification and regression problems. The evaluation parameter used in rating the performance of a regression algorithm is the mean squared error (MSE), whereas, for rating the performance of a classification algorithm, we utilized precision, recall, and F-score as evaluation metrics, which are constructed using a confusion matrix.

Additionally, the possible directions in evaluating the performance of both regression and classification models are analyzed using 15 evaluation parameters. The finding in our research is that machine learning algorithms can be applied to analyses the data to extract the hidden patterns from it, which in turns helps in effective decision making.

Python Language:

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built-in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity

and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.

Python is commonly used for developing websites and software, task automation, data analysis, and data visualization. Since it's relatively easy to learn, Python has been adopted by many non-programmers such as accountants and scientists, for a variety of everyday tasks, like organizing finances. Here, we are specifying application areas where Python can be applied.

1. Web Applications
2. Desktop GUI Applications
3. Console-based Application
4. Software Development
5. Scientific and Numeric
6. Business Applications
7. Audio or Video-based Applications
8. 3D CAD Applications
9. Enterprise Applications
10. Image Processing Application

For the creating the rapid development of secure and maintainable websites we also use the Django Framework:

Django Framework in Python:

Django is a high-level Python web framework that encourages rapid development and clean, pragmatic design. Built by experienced developers, it takes care of much of the hassle of web development, so you can focus on writing your app without needing to reinvent the wheel. It's free and open source.

As Django assists in building and maintaining quality web applications. Django helps eliminate repetitive tasks making the development process an easy and time saving experience.

The following are some features of Django:

1. Ridiculously fast - Django was designed to help developers take applications from concept to completion as quickly as possible.
2. Reassuringly secure - Django takes security seriously and helps developers avoid many common security mistakes.
3. Exceedingly scalable - Some of the busiest sites on the web leverage Django's ability to quickly and flexibly scale.

Web Development Languages:

HTML:

HTML stands for Hyper Text Markup Language. It is used to design web pages using a markup language. HTML is the combination of Hypertext and Markup language. Hypertext defines the link between the web pages. A markup language is used to define the text document within tag which defines the structure of web pages. This language is used to annotate (make notes for the computer) text so that a machine can understand it and manipulate text accordingly. Most markup languages (e.g., HTML) are human-readable. The language uses tags to define what manipulation has to be done on the text.

HTML is a markup language used by the browser to manipulate text, images, and other content, in order to display it in the required format.

CSS:

Cascading Style Sheets, fondly referred to as CSS, is a simply designed language intended to simplify the process of making web pages presentable. CSS allows you to apply styles to web pages. More importantly, CSS enables you to do this independent of the HTML that makes up each web page. CSS is easy to learn and understand, but it provides powerful control over the presentation of an HTML document.
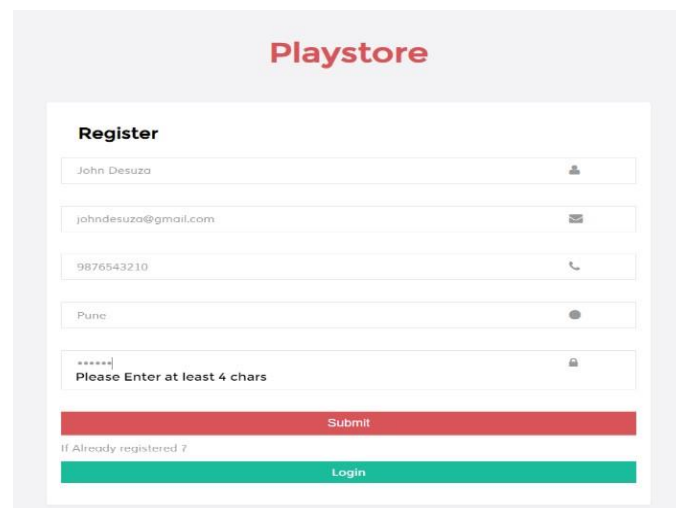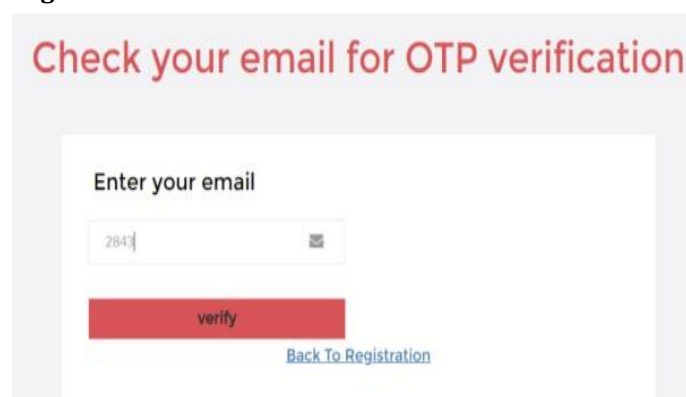
JS:

JavaScript is a lightweight, cross-platform, and interpreted scripting language. It is well-known for the development of web pages; many non-browser environments also use it. JavaScript can be used for Client-side developments as well as Server-side developments. JavaScript contains a standard library of objects, like Array, Date, and Math, and a core set of language elements like operators, control structures, and statements.
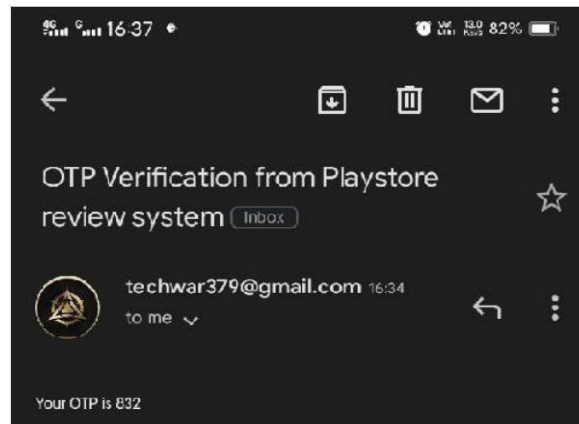
VI.   IMPLEMENTATION DETAILS

Mobile app distribution platform such as Google play store gets flooded with millions of new applications uploaded by developers every day. So, in this project, we aim on analyzing Google play store that provides a particular app description and data such as reviews, ratings, price and number of downloads. The objective of this is to analyses the desire of the customer through the reviews provided in the feedback section and apps trend in the market to help the organization & developers.

To this end, we provide an idea about app that managed to get maximum and minimum number of downloads and predicting the category of apps that is most likely to be downloaded in the coming years. Moreover, doing sentimental analysis on the apps that generated most positive and negative sentiments, Sustainability of app in market on basis of previous data and current market situation. Furthermore, also analyzing the apps that has maximum downloads have they managed to get average rating so that concluding the co relation between numbers of downloads and ratings received. There will be different tabs available for different operations and analysis that user wants to view. We have used the below given algorithm for implementation: -
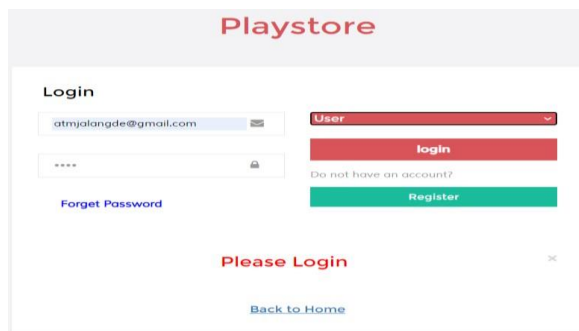
The high-level features and information can be extracted on the basis of the information generated and sentiment scores are evaluated.

VII.   USER MANUAL (OUTPUT)

7.1 Registration

7.2 Email verification using OTP
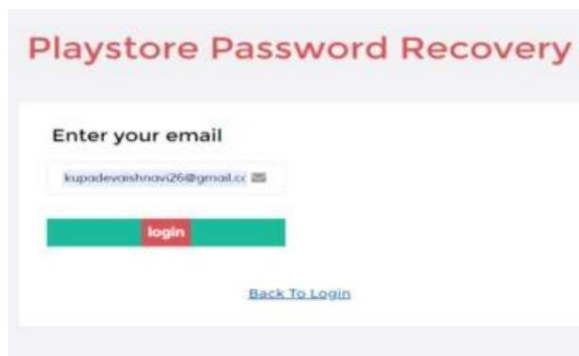
7.3 OTP on User Side
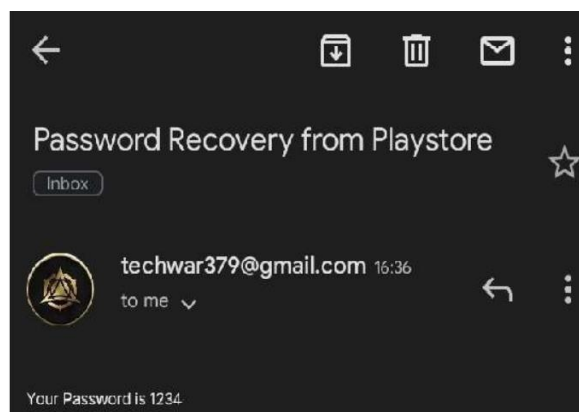

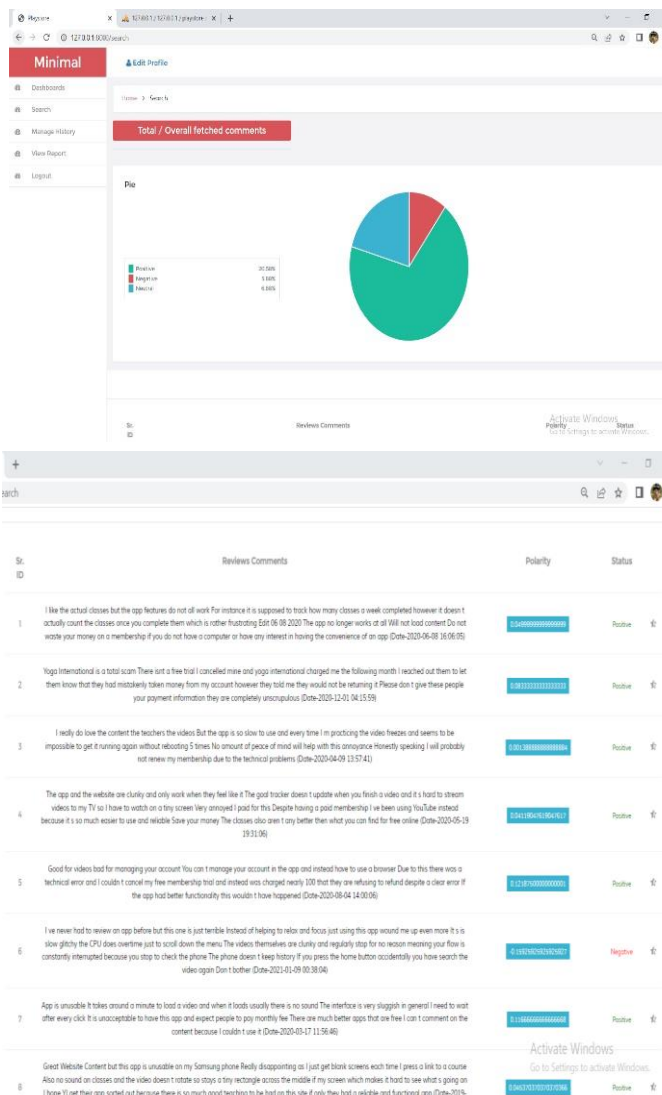**7.4 Login**

7.5 Forgot Password

7.6 Password Recovery

7.7 To Search the App Using App Id

7.8 App comment with graph and in tabular form with polarity

VIII.   SCOPE AND FUTURE WORK

- Going to continue to dig deeper, far past the surface of the number of likes, comments and shares, and aim to reach, and truly understand, the significance of social media interactions and what they tell us about the consumers behind the screens.
- Updating dictionary for synonyms and antonyms: Update the dictionary to easily recognize the what user can say actually means, the shortcuts of any word or any word related to that word and all can update into that dictionary.
- Web application can be converted into mobile application: Now, it is on website application, but in future it can converted into mobile application for flexibility to mobile users.

- Can be created in other languages using Data sets and algorithm: Now, it is on US English, but for the user understanding or user-friendly purpose it will make it on other languages also by using Data sets and Google Translator.
- In future it may be an automate feature by the Google play store.

IX.   CONCLUSION

Today's world Google play store has become the major hub for downloading and uploading Android application. The Android application users download applications for their personal use. Each user of the application has their own experience with the application. Users download and use these applications and express the experience of the application in the shape of comments or reviews, also give a rating to this application on the scale of 0-5. As an outcome of the project, the comments can be sent to owner. The owner is either delete the bad comments or make a correction in the app. We studied all required approaches of these project and proposed a new product to classify the polarity behind the emotions of the customers/users. We have been seen that it is easy to implement sentiment analysis.

X.   REFERENCES

[1]   Rishi Meridian, "Analysis of Google Play Store Data set and predict the popularity of an app on Google Play Store", Texas a&M University College Station, Texas, June-2020. Jong, J. (2011). Predicting rating with sentiment analysis.[Online]

http://cs229.stanford.eddo/proj2011/JongPredictingRatingwithSentimentAnalysis.pdf.

[2]   https://en.wikipedia.org/wiki/Sentiment_analysis

[3]   Harman, M., Jiao, Y., and Zhang, Y. (2012). App store mining and analysis: Mrs for app Stores. In 2012 9th IEEE Working Conference on Mining Software Repositories (MSR), Pages 108–111.

[4]   https://developer.mozilla.org/en-US/docs/Learn/Serverside/Django/Introduction

[5]   https://jovian.ai/learn/data-analysis-with-python-zero-topandas

[6]   In the Google play store. Empirical Software Engineering, 21(3), 1346-1370. Islam, M. R. (2014). Numeric rating of apps on Google play store by sentiment Analysis on user reviews. In 2014 International Conference on Electrical Engineering And Information Communication Technology, pages 1–4.