

VrishHub-II

Project Report

Submitted By:

Vinayak Rao Dikshit 201B308

Virad Chaurasia 201B354

Saurabh Singh 201B364

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

at



JAYPEE UNIVERSITY OF ENGINEERING & TECHNOLOGY

GUNA , MADHYA PRADESH (INDIA) – 473226

May-2023

DECLARATION

I hereby declare that the project entitled “VrishHub-II” submitted for the B. Tech. (CSE) degree is our original work and the project has not formed the basis for the award of any other degree, diploma, fellowship or any other similar titles.

Signature of the Students

Virad Chaurasia(201B354)

Vinayak Rao Dikshit (201B308)

Saurabh Singh (201B364)

Place: Jaypee University of Engineering & Technology Guna,
Madhya Pradesh (India) – 473226

Date: 15/May/2023

CERTIFICATE

This is to certify that the project titled “VrishHub-II” is the bona fide work carried out by Virad Chaurasia, Vinayak Rao Dikshit, Saurabh Singh, the students of B Tech (CSE) of Jaypee University of Engineering and Technology, Guna (M.P).

During the academic year 2022-23, in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology (Computer Science and Engineering) and that the project has not formed the basis for the award previously of any other degree, diploma, fellowship or any other similar title.

Signature of the Guide

Virad Chaurasia (201B354)

Vinayak Rao Dikshit (201B308)

Saurabh Singh (201B364)

Place: Jaypee University of Engineering & Technology Guna,
Madhya Pradesh (India) – 473226

Date: 15/May/2022

Abstract

Adaptive recommendation system that provides real-time personalized trading advice to the investors based on their profiles and trading information environment. A proposed system integrates Stochastic technical analysis and recurrent neural network that incorporates an adaptive user modelling. The user model is constructed and updated based on initial user profile and recorded user interactions with the system. The information presented to each individual user is also tailormade to fit the user's behaviour and preference.

Experiments used to evaluate the system's performance will be done on both human subjects and synthetic users. The results will show our proposed system is able to rapidly learn to provide appropriate advice to different types of users.

Acknowledgement

We would like to express our gratitude and appreciation to all those who gave us the opportunity to complete this project. Special thanks are due to our supervisor Dr. *Mahesh Kumar Sir* whose help, stimulating suggestions and encouragement helped us in all the time of development process and in writing this report. We sincerely thank him for the time spent proofreading and correcting my many mistakes.

We would also like to thank our parents and friends and who helped us a lot in finalizing this project within the limited period. Last but not the least I am grateful to all the team members of VrishHub-II.

List of Figures

Figure	Title	Page Number
3.2.1	Structure Chart	19
3.2.2	Context Diagram	20
3.2.3	Use Case Diagram	20
3.2.4	Dataflow Diagrams	21-22
3.2.5	Sequence Diagrams	23-34
3.2.6	State Diagram	25
3.2.7	Class Diagram	26
3.2.8	Flowchart of the Neural Network	27

Table of Contents

Title page	i
Declaration of the Student	ii
Certificate of the guide	iii
Abstract	iv
Acknowledgement	v
List of Figure	vi

Chapter-1 INTRODUCTION

- 1.1 Problem Definition
- 1.2 Project Overview
- 1.3 Requirement Engineering
- 1.4 Software Requirement Specification
- 1.5 Hardware Specification

Chapter-2 LITERATURE SURVEY

- 2.1 Existing System
- 2.2 Proposed System
- 2.3 Feasibility Study

Chapter-3 SYSTEM ANALYSIS & DESIGN

3.1 Requirement Specification

3.1.1 Frontend

- 3.1.1.1 HTML
- 3.1.1.2 CSS
- 3.1.1.3 Java Script

3.1.2 Backend

- 3.1.2.1 Firebase
- 3.1.2.2 MongoDB
- 3.1.2.3 Express
- 3.1.2.4 ReactJS

3.1.2.5 NodeJS

3.1.3 Deep Learning

3.1.3.1 Python

3.1.3.2 Google Colab

3.1.3.3 Tensor Flow

3.1.3.4 Matplotlib

3.1.3.5 NumPy

3.1.3.6 Pandas

3.1.3.7 Scikit Learn

3.2 Flowcharts /Diagrams

3.2.1 Structure Chart

3.2.2 Context Diagram

3.2.3 Use case Diagram

3.2.4 Data Flow Diagram

3.2.5 Sequence Diagrams

3.2.6 State Diagram

3.2.7 Class Diagram

3.2.8 Neural Network

Chapter-4 RESULTS/OUTPUTS

Chapter-5 CONCLUSIONS/RECOMMENDATIONS

Chapter-6 REFERENCE

INTRODUCTION

1.1 Problem Definition

These days stock brokers are prominent throughout the market, but their analysis is based on old school forecasting and mathematical models which are useful but lack reliability. Furthermore, they charge hefty prices even regardless of whether their suggestions are fruitful or not. Therefore, we thought about automating this process via the powerful computational capacity of Artificial neural networks. That is how the idea of VrishHub-II: (Automated) Personalized Stock Recommendations was born.

1.2 Project Overview

VrishHub-II is an ML-backed Web Platform that will help you in overcoming all the hassles of Stock Trading. VrishHub-II will analyze your Stock Portfolio and with your data of previous decisions and it will assess changes in the stocks after your sold or bought them for providing you with the best options to go for. Unlike other Algorithmic Trading platforms, VrishHub-II will give you personalized solutions.

VrishHub-II combines the traditional stock prediction methods with the power of modern-day computers to give fast, accurate and real time predictions of your favorite stocks – all for free. The results of the project are expressed in this report, and present how Recurrent Neural Networks and LSTMs are utilized in enabling our dear investors to make decisions which maximizes their return on investment.

1.3 Requirement Engineering

1. Requirement Elicitation

A) Normal (Functional requirements):

1. *Personalization* is a crucial requirement for a stock recommendation system to provide customized recommendations to individual users, which suits their portfolio.
2. User will be able to see list of all the available stocks at our website, in case they want to invest in a new stock sector.
3. Allow users to login using their existing broker ID. Since we are using a Third-Party API so we expect user to perform login with their broker's ID.

B) Expected Requirements

1. Recommending stocks in such a way that user's portfolio loss can be covered up or profit can be increased by the suggested stock.
2. *Predictive analytics* can be used to identify patterns and predict future performance of stocks. By using deep learning algorithms to analyze historical data and identify trends, a stock recommendation system can provide users with insights into which stocks are likely to perform well in the future.

C) Exciting Requirements

1. Social collaboration: Providing users feature of blogs and updated business news.

1.4 Software Requirement Specification

1.4.1 Introduction

1.4.1.1 Purpose:

This software requirement specification document has been written to provide to the developers, a clear vision of what to develop and what functionalities and features to integrate so as make this project- "VrishHub" complete.

Completing this SRS will give the development team a point of view from the customer's side. The developers will also get an idea about our input/output data, the various interfaces and scenarios etc.

Once the SRS has been written, the requirements would have been finalised and no further changes shall be entertained.

1.4.1.2 Scope:

The various types of users are mentioned below:

Engaged Users: These are our permanent users, having an active Demat account, spending a significant amount of time on our website, engaging with their personalised portfolio dashboard and other multiple interfaces and utilising all of our services.

Returning Users: These are the casual users who visit our website for getting updated stock related information or news

1.4.1.3 Definition, Abbreviations, and Acronyms

1.4.1.3.1 Definitions:

Algorithmic Trading: It is a method of executing orders using automated pre-programmed trading instructions accounting for variables such as time, price, and volume.

LSTM: It is an artificial neural network used in the fields of artificial intelligence and deep learning. Unlike standard feedforward neural networks, LSTM has feedback connections.

Demat Account: A demat account helps investors hold shares and securities in an electronic format.

1.4.1.3.2 Abbreviations:

Demat Account: Dematerialisation account

1.4.1.3.3 Acronyms:

API : Application Programming Interface

ID: Individual description

ML: Machine Learning

DL: Deep Learning

LSTM: Long Short Term Memory Unit

RNN: Recurrent Neural Network

LAN: Local Area Network

WLAN: Wireless Local Area Network

1.4.1.4 Overview

Section 1- Introduction: It defines the purpose of creating the SRS, the various sources from various sources from where information has been taken, the project specific acronyms and abbreviations that may not be known to the common person.

Section 2- Overall Description: It describes the various hardware/software interfaces required in the project, traits required in the user, different limitations and assumptions made for the development of the project.

Section 3- Specific Requirement: States in details about the functionalities of the software, the interaction between users and software, attributes of the system and organisation of requirements.

1.4.1.5 The Overall Description

1.4.1.5.1 Product Perspective

System Interfaces: N.A.

User Interfaces: For the users, 4 screens will be there.

- The landing page
- The login page
- Portfolio Dashboard
- News Page

Hardware Interfaces: All the user interfaces require the connection of an external keyboard and external mouse with them at all times.

Software Interfaces: There are two screens where an API is connected:

- Login page (3rd party broker)
- News page (Money control news API)

Communication Interfaces: A stable (and preferable fast) internet connection either via LAN or WLAN is required at all user screens.

Memory Constraints: Minimum 120 MB RAM and 380 MB of hard drive space (to run in any modern web browser).

Operations: N.A.

1.4.5.2 Product functions:

1.Personalization is a crucial requirement for a stock recommendation system to provide customized recommendations to individual users, which suits their portfolio.

2. User will be able to see list of all the available stocks at our website, in case they want to invest in a new stock sector.
3. Allow users to login using their existing broker ID. Since we are using a Third-Party API so we expect user to perform login with their broker's ID.

1.4.5.3 User characteristics

User should be versed in English language.

User should know how to open and use the web browser.

User Should know basic working and terminologies associated with the stock market.

User should know how to interact with the website by reading instruction from the display.

1.4.5.4 Constraints

Time constraints: Even Semester 2023 before Presentation 3.

Cost Constraints: We are able to fetch, process and handle a limited amount of data because of No External monetary support.

Resource Constraints: Unable to train a large amount of data due to low system configuration and unable to fetch large user data because of restricted API support.

1.4.5.5 Assumptions and Dependencies

All the users know how to operate a computer.

We are assuming that the predicted stock prices are unaffected by sudden stock market disruptions and seasonal variations.

We are also assuming a large time interval of approximately 6hrs in our dataset to train the model. This is highly unrepresentative of the real world.

We are also assuming that our system will not be affected by external security breach.

1.4.5.6 Apportioning of Requirements

N.A. since we are using waterfall approach.

1.4.1.6 Specific Requirements

External Interfaces

- User will be able to see list of all the available stocks at our website, in case they want to invest in a new stock sector.
- Allow users to login using their existing broker ID. Since we are using a Third-Party API so we expect user to perform login with their broker's ID.

Functions

Personalization is a crucial requirement for a stock recommendation system to provide customized recommendations to individual users, which suits their portfolio. We have developed a model which analyses user's past history to give them recommendation of new stocks to invest in, furthermore our deep learning model will also recommend stocks in such a way that will reduce the users losses.

To run our website, a chromium-based browser or internet explorer version 11 is required. The operating system requirements for it are given below:

Windows/Mac/Linux

Windows 7, Windows 8, Windows 8.1, Windows 10 or later

Mac

macOS High Sierra 10.13 or later

Linux

64-bit Ubuntu 18.04+, Debian 10+, openSUSE 15.2+, or Fedora Linux 32+

Android

Android 7.0 Nougat

1.5 Hardware Specification

Desktop PCs

- An Intel Pentium 4 processor or later that's SSE3 capable.

LITERATURE SURVEY

2.1 Review of the Literature

Panda et al. (2011): He revealed that majority of investors depend on newspapers as the source of information. Mostly people who invested in shares they were ranked financial journals and business magazines next to newspapers. Many investors believed they were not able to get the required information from the company in time. Many investors were found to face problems while selling securities.[1]

Kasisomayajula (2012): He found that many investors are facing problems for trading in stock market due to lack of information of stock market. Investors are more founded to invest their money in post office schemes, banks etc. and they are very confined towards share trading. [2]

Abdul Rahim (2013): Analysed the problems and prospects of online share trading practices in India and found that the main benefit derived out of online share trading is wider choice followed by better value and source of information. Results also revealed that inadequate technology is the major problems faced by online share traders followed by lack of professional management. [3]

Prabakaran (2017): Found that all investors considered other relevant information in relating to share market before entering the online share trading. The higher income people make the investment in shares via online trading, because that they knew the all kinds of information as well as facing their risk.

2.1 Existing System

There have been many works on the development of Intelligent UI in several areas. For examples, Webber et al. [2] and Tasso et al. [3] employ user-modelling techniques based on student skill and background knowledge to develop Information Tutoring System. Brusilovsky [4] presents browsing-based access to information source area. The author also mentioned about link manipulation, e.g., hide, sort, annotate, and adaptive presentation technique

according to user modelling, interest, and knowledge respectively. Information Retrieval and Filtering is another area that is related. Shepherd et al. [5] presents a framework that helps users filtering news, and Kay [6] proposed the Movies Advisor Project that could suggest movies to a specific user based on their interest. For stock trading systems, most of the existing systems use machine learning and soft computing techniques to analyse a company's stock and suggest an action to the investors. Examples include an online financial informational web service, Tradetrek [7]. Achelis [8] proposed trading algorithm that works by using trading indicators as parameters in mathematics equations. Kuo [9] used Neurofuzzy approach for predicting financial time series. Tseng and Gmytrasiewicz [10] developed the real-time DSS system, based on object-oriented technology and using Bayesians Network, to produce investment recommendation. To the best of our knowledge, most of the works mentioned above do not consider the individual investor's profile into the recommendation process. Recently, Yoo et al. [11] proposed a leaning algorithm to personalize advice to different users. However, they did not emphasize on real time recommendation. The underlying motivation of our work is to build a real-time personalized stock recommendation and portfolio management system that is capable of customizing information based on user's profile. The system must also have ability to adapt user's model as well as information presentation based on the interactions of the user with the system[4]

2.2 Proposed System

We make use of the user's profile, usage data, and portfolio status to indicate what stocks that user should be interested with some actions. Briefly, when a user login to the system, a user model will be loaded. While user is interacting with the system, the system learns user behaviour and self-adapts the model so that the provided information fits the user's need. We use supervised induction algorithm of RNNs for improving leaning process. We divide the user's interaction tracking tasks into two major tasks as follows: First, a feedback session is provided to detect the user's acceptance and rejection. If user responses by accepting the advice, a positive example is created. Conversely, if user response is rejection, a negative example is created. These examples are fed back to the system for continuing learning process. Second, the relevant/interest data to the user is monitored. As mentioned before, user interaction is recorded. To help user coping with the problem of information overload, the system try to understand user's behaviour and preferences. Using history of interaction and

records of transactions made on a particular stock item to update user model, the system can tailor information that is interest or useful to the user.[5]

2.3 Feasibility Study

A significant consequence of starter examination is the affirmation that the framework request is feasible. This is possible just if it is viable inside limited resource and time. The various potential outcomes that must be dismembered are

- Operational Feasibility
- Economic Feasibility
- Technical Feasibility

Operational Feasibility:

Operational Feasibility deals with the examination of prospects of the framework to be made. This framework operationally assists customers in sufficiently foreseeing stock estimation of an association, with the objective that customers can settle on up their stock exchanging decisions similarly as improve the gauge model reliant on the comprehension. In perspective on the examination, the framework is wound up being operationally viable.

Economic Feasibility:

Monetary Feasibility or Cost-advantage is an assessment of the budgetary resistance for a PC based endeavor. As gear was presented from the most punctual beginning stage and for heaps of purposes along these lines the cost on the undertaking of hardware is low.

Technical Feasibility:

The system is made for the stage independent condition. Python code, Html, CSS, Javascript coming up short immediately of pycharm are used to develop the structure. The specific reachability has been finished. The structure is really down to earth for development and can be made with the present office

SYSTEM ANALYSIS & DESIGN

3.1.1 System Architecture

The Prototype of the system is developed based on client server architecture. The server contains 4 main components.

- **Trading Information Manager** gathers and manages various data necessary for trading decision such as current stock information, historical trading information, etc.
- **User Modeler Component** exploits a Recurrent Neural Network to construct an initial user model and to update and revise the model during the user's interaction with the system.
- **Personalized Recommendation Agent** makes appropriate suggestion for the company stock based on individual investor profile and trading environment.
- **Presentation Module** adaptively and dynamically generates the information to be presented to the user.

To create personalized recommendation, the system first constructs an initial user model from user profile using RNNs. Then converting trading constraints into decision rules for trading decision making. Next, the system selects and ranks appropriate information for the presentation based on individual user profile. The user's interactions and responses to previous recommendation are tracked and make these data become activate rule for dynamic version of the user model. This may in turn change the recommendation in the future.

3.1 Requirement Specification

3.1.1 Frontend

3.1.1.1 HTML

HTML (Hypertext Markup Language) is a text-based approach to describing how content contained within an HTML file is structured. This markup tells a web browser how to display text, images and other forms of multimedia on a webpage.

HTML is used to create webpages but does experience limitations when it comes to fully responsive components. Therefore, HTML should only be used to add text elements and structure them within a page. For more complex features, HTML can be combined with cascading style sheets (CSS) and JavaScript (JS).

In the early days of the world wide web, marking up text-based documents using HTML syntax was more than sufficient to facilitate the sharing of academic documents and technical memos. However, as the internet expanded beyond the walls of academia and into the homes of the general population, greater demand was placed on webpages in terms of formatting and interactivity.

3.1.1.2 CSS

Cascading Style Sheets, fondly referred to as CSS, is a simple design language intended to simplify the process of making web pages presentable.

CSS handles the look and feel part of a web page. Using CSS, you can control the color of the text, the style of fonts, the spacing between paragraphs, how columns are sized and laid out, what background images or colors are used, layout designs, variations in display for different devices and screen sizes as well as a variety of other effects.

CSS is used along with HTML and JavaScript in most websites to create user interfaces for web applications and user interfaces for many mobile applications.

3.1.1.3 Java Script

JavaScript is a dynamic computer programming language. It is lightweight and most commonly used as a part of web pages, whose implementations allow client-side script to interact with the user and make dynamic pages. It is an interpreted programming language with object-oriented capabilities.

JavaScript was first known as Live Script, but Netscape changed its name to JavaScript, possibly because of the excitement being generated by Java. JavaScript made its first appearance in Netscape 2.0 in 1995 with the name Live Script. The general-purpose core of the language has been embedded in Netscape, Internet Explorer, and other web browsers.

3.1.2 Backend

3.1.2.1 Firebase

Firebase is a development platform known originally for its real-time database that's still at its core a multi-node, key-value database optimized for synchronizing data, often between user machines or smartphones and centralized storage in the cloud. It's designed to make life easier for developers by handling much of the pushing and pulling of data. That relieves app developers of the programming burdens associated with managing versions or locations. They can write the new bits to Firebase and the data will be consistent throughout the system. Firebase is valued largely because it can constantly propagate and synchronize changes between local copies of information stored on users' machines with versions kept in the cloud. Firebase eliminates many of the challenges of mixing authentication, synchronization, and segregation by juggling multiple versions and ensuring the right bits are the same throughout the system.

3.1.2.2 MongoDB

MongoDB is an open-source document-oriented database that is designed to store a large scale of data and also allows you to work with that data very efficiently. It is categorized under the NoSQL (Not only SQL) database because the storage and retrieval of data in the MongoDB are not in the form of tables.

MongoDB is a database server and the data is stored in these databases. Or in other words, MongoDB environment gives you a server that you can start and then create multiple

databases on it using MongoDB. Because of its NoSQL database, the data is stored in the collections and documents.

3.1.2.3 ExpressJS

Express.js is a web framework for Node.js. It is a fast, robust and asynchronous in nature. Express is a fast, assertive, essential and moderate web framework of Node.js. You can assume express as a layer built on the top of the Node.js that helps manage a server and routes. It provides a robust set of features to develop web and mobile applications.

3.1.2.4 ReactJS

React.js, more commonly known as React, is a free, open-source JavaScript library. It works best to build user interfaces by combining sections of code (components) into full websites. Originally built by Facebook, Meta and the open-source community now maintain it. One of the good things about React is that you can use it as much or as little as you want! For example, you can build your entire site in React or just use one single React component on one page.

React.js is built using JSX – A combination of JavaScript and XML. Elements are created using JSX, then use JavaScript to render them on your site. While react has a steep learning curve for a junior developer, it's quickly shaping into one of the most popular and in-demand JavaScript libraries.

React is considered a JavaScript library rather than a framework, whereas the other options we'll consider today are considered frameworks. It helps to think of a library as a tool that developers could use in any project and a framework as a whole design.

There has been a massive rise in the popularity of React.js, as found in the studies by [State Of JS](#). This is partially due to its flexibility and speed of development but also helped by the fact that it is supported by Meta, making developers and companies feel safe in their decision to use React. Therefore, the demand for React developers is very high. As a result, there is a wide range of jobs for developers who know how to use React.

3.1.2.5 NodeJS

Node.js is an open source, cross-platform runtime environment and library that is used for running web applications outside the client's browser.

It is used for server-side programming, and primarily deployed for non-blocking, event-driven servers, such as traditional web sites and back-end API services, but was originally designed with real-time, push-based architectures in mind. Every browser has its own version of a JS engine, and node.js is built on Google Chrome's V8 JavaScript engine.

In simple terms, what this means is that entire sites can be run using a unified 'stack', which makes development and maintenance quick and easy, allowing you to focus on meeting the business goals of the project. The mechanics of Node.js are what contributes to its popularity with developers. Whereas most alternative runtime environments utilize multi-threaded processing models, Node.js does it all in a single thread. In multi-threaded processing setups, each server has a limited thread pool it can access. So every time a server receives a request, it pulls a thread from the pool and assigns it to that request, to take care of processing it. In this case, the processing is synchronous and sequential, which means that one operation is performed at a time. Node.js, however, uses single-threaded processing. The difference between the two is as you'd imagine: single-thread architectures process every request using a single main thread, utilizing event loops to run blocking Input/Output operations in a non-blocking way.

3.1.3 Deep Learning

3.1.3.1 Python

AI algorithms and machine learning models are complex predictive technologies that Python can simplify. There is a vast database of libraries and frameworks that Python uses for machine learning purposes. For eg: NumPy, Keras, Tensorflow, Matplotlib, Seaborn, Pytorch etc.

Software solutions developed with Python can be built and also can run on multiple operating system platforms. For instance, Linux, Windows, Mac, Solaris, and more. This makes python programming machine learning a lot more convenient. That's why developers enjoy Python in the process of developing ML apps.

3.1.3.2 Google Colab

Colab, or "Collaborator", allows you to write and execute Python in your browser. Colab notebooks allow you to combine executable code and rich text in a single document, along with images, HTML, LaTeX and more. Colab notebooks are Jupyter notebooks that are hosted by Colab. With Colab you can import an image dataset, train an image classifier on it, and evaluate the model hence, Colab is used extensively in the machine learning community.

3.1.3.3 Tensor Flow

TensorFlow is an end-to-end open-source platform for machine learning. It has a comprehensive, flexible ecosystem of tools, libraries, and community resources that lets researchers push the state-of-the-art in ML and developers easily build and deploy ML-powered applications.

TensorFlow was originally developed by researchers and engineers working on the Google Brain team within Google's Machine Intelligence Research organization to conduct machine learning and deep neural networks research. The system is general enough to be applicable in a wide variety of other domains, as well.

3.1.3.4 Numpy

NumPy stands for '*Numerical Python*'. It is an open-source Python library used to perform various mathematical and scientific tasks. It contains multi-dimensional arrays and matrices, along with many high-level mathematical functions that operate on these arrays and matrices.

Moreover, Numpy forms the foundation of the Machine Learning stack.

NumPy arrays are stored at one continuous place in memory unlike lists, so processes can access and manipulate them very efficiently. This behavior is called locality of reference in computer science. This is the main reason why NumPy is faster than lists. Also, it is optimized to work with latest CPU architectures. NumPy is written partially in Python, but most of the parts that require fast computation are written in C or C++.

3.1.3.5 Matplotlib

Matplotlib is an open-source plotting library in Python introduced in the year 2003. It is a very comprehensive library and designed in such a way that most of the functions for plotting in MATLAB can be used in Python.

It consists of several plots like the Line Plot, Bar Plot, Scatter Plot, Histogram etc. through which we can visualize various types of data.

3.1.3.6 Pandas

Pandas is one of the tools in Machine Learning which is used for data cleaning and analysis. It has features which are used for exploring, cleaning, transforming and visualizing from data.

Pandas is derived from the term “**Panel-data-s**” an econometrics term for data sets include observations over multiple time periods for the same individuals.

Pandas deals with three types of data structures:

- Series
- DataFrame
- Panel

3.1.3.7 Scikit Learn

Scikit-learn (Sklearn) is the most useful and robust library for machine learning in Python. It provides a selection of efficient tools for machine learning and statistical modeling including classification, regression, clustering and dimensionality reduction via a consistency interface in Python. This library, which is largely written in Python, is built upon NumPy, SciPy and Matplotlib.

Some of the most popular groups of models provided by Sklearn are as follows:

Supervised Learning algorithms (like Linear Regression, Support Vector Machine (SVM), Decision Tree etc.), Unsupervised Learning algorithms, Clustering, Cross Validation, Dimensionality Reduction, Ensemble methods, Feature extraction, Feature selection.

3.2 Flowcharts and Diagrams

3.2.1 Structure Chart

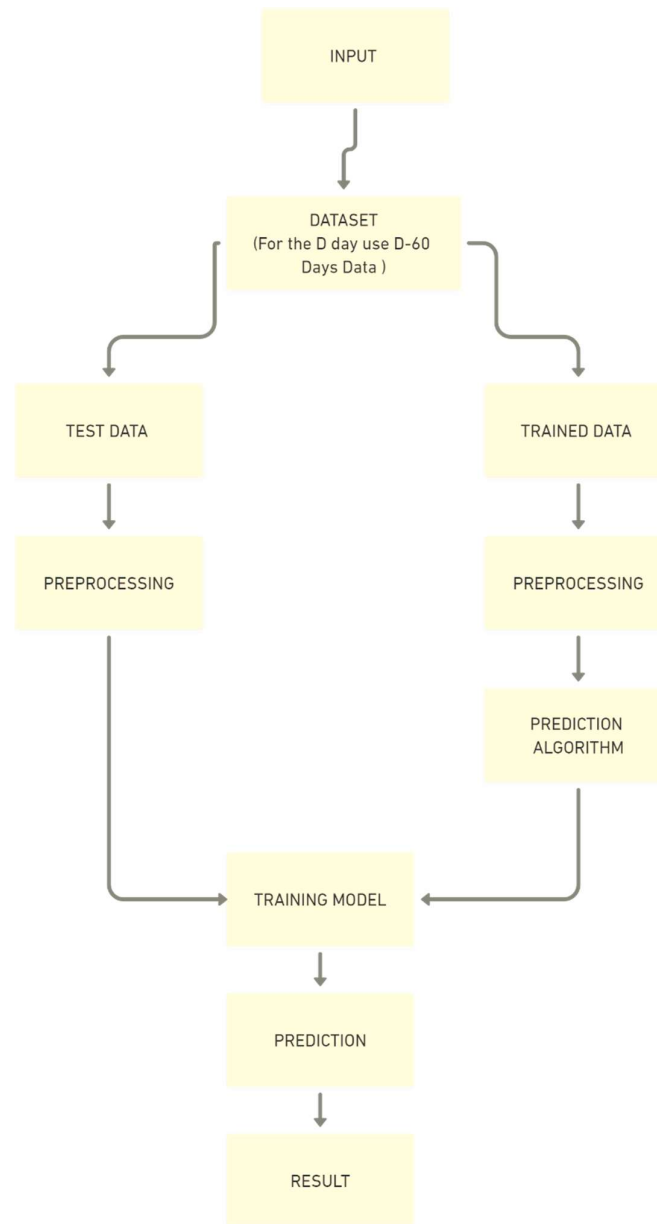


Fig.1 Structure Chart

3.2.2 Context Diagram

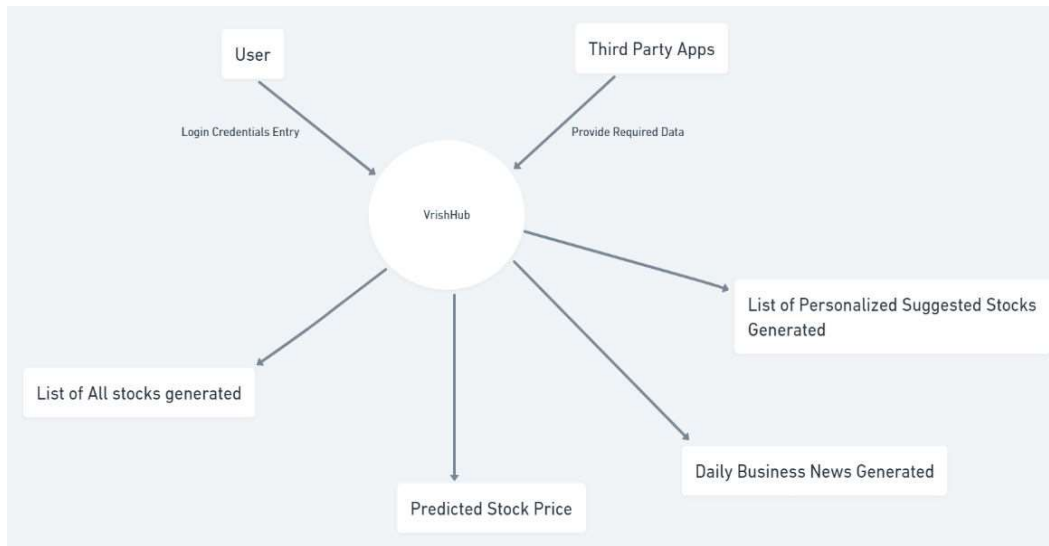


Fig.2.Context Diagram

3.2.3 Use case Diagram



Fig.3.Use Case Diagram

3.2.4 Data Flow Diagrams



Fig.4.Level 0 Data Flow Diagram

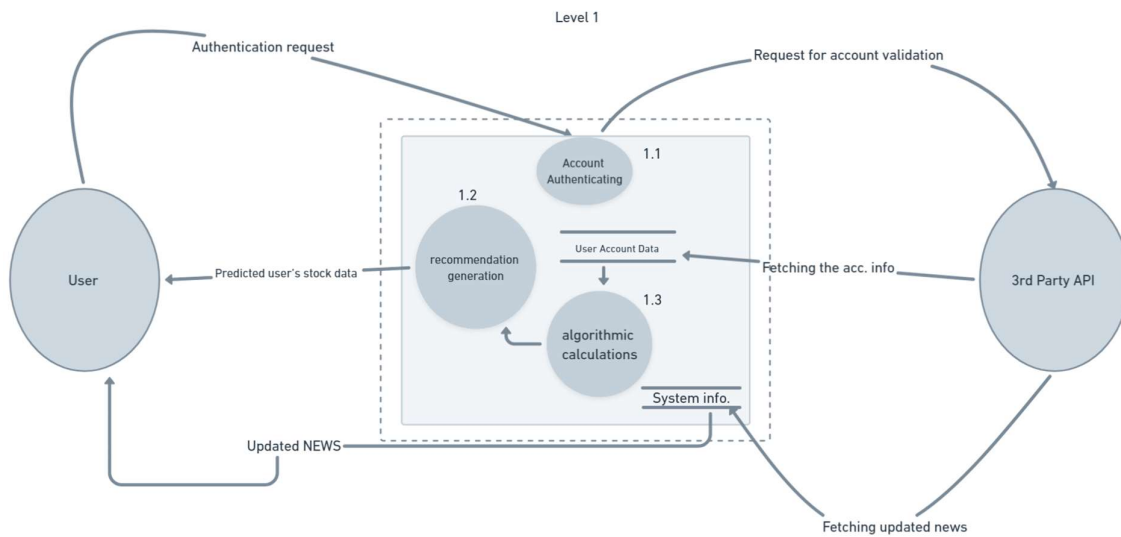


Fig.5. Level 1 Data Flow Diagram

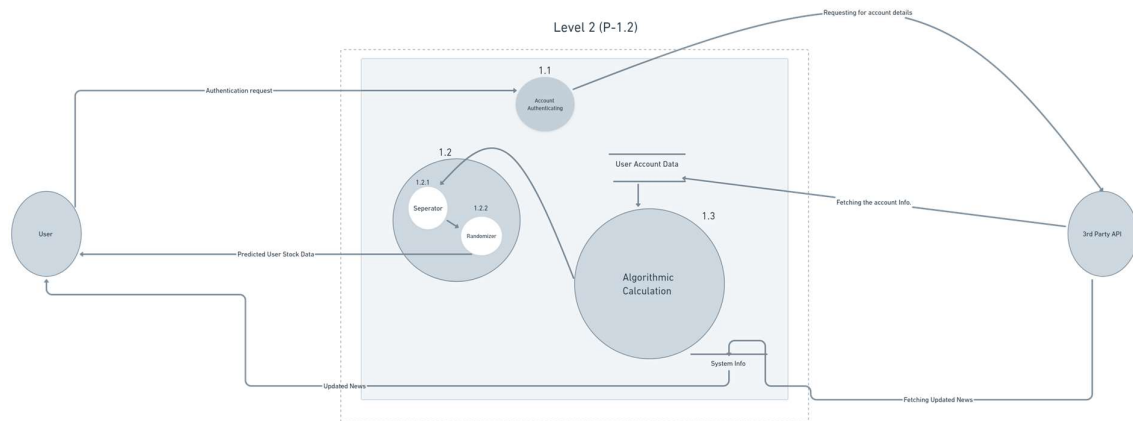


Fig.6. Level 2(P-1.2) Data Flow Diagram

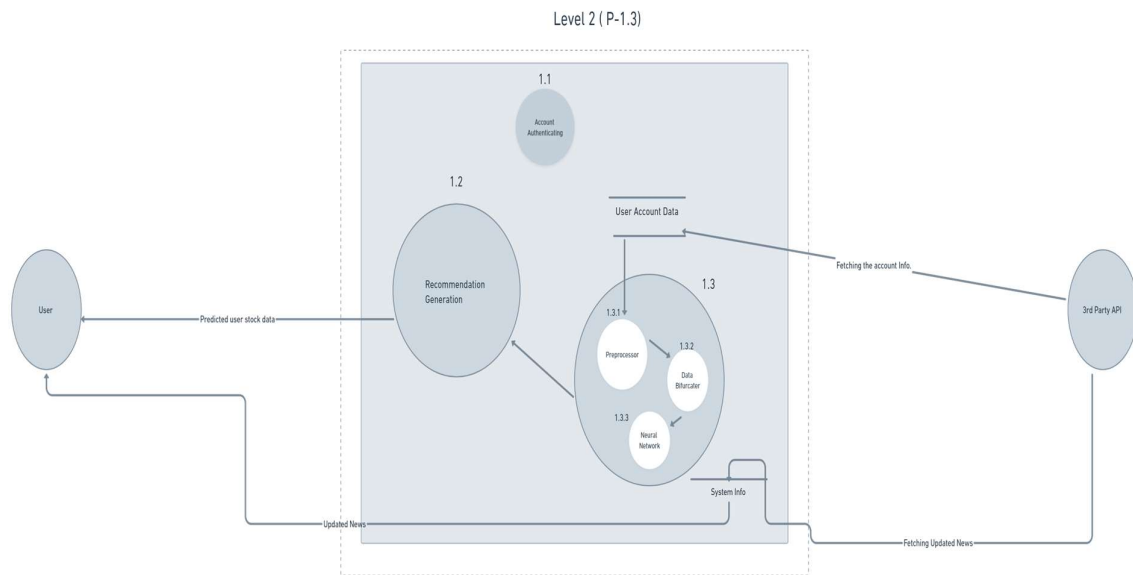


Fig.7. Level 2(P-1.3) Data Flow Diagram

3.2.5 Sequence Diagrams

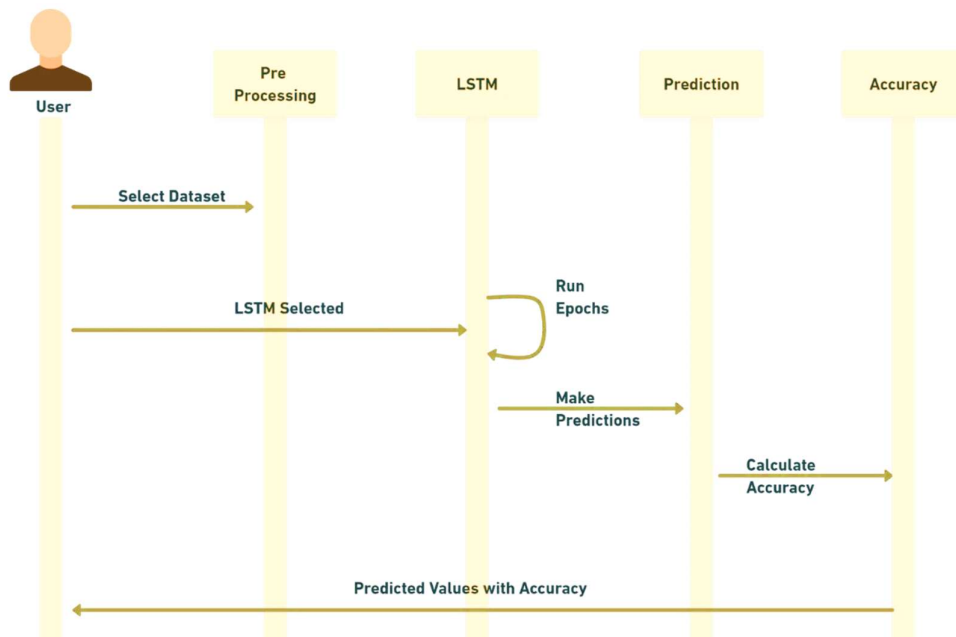


Fig.8.LSTM Sequence Diagram

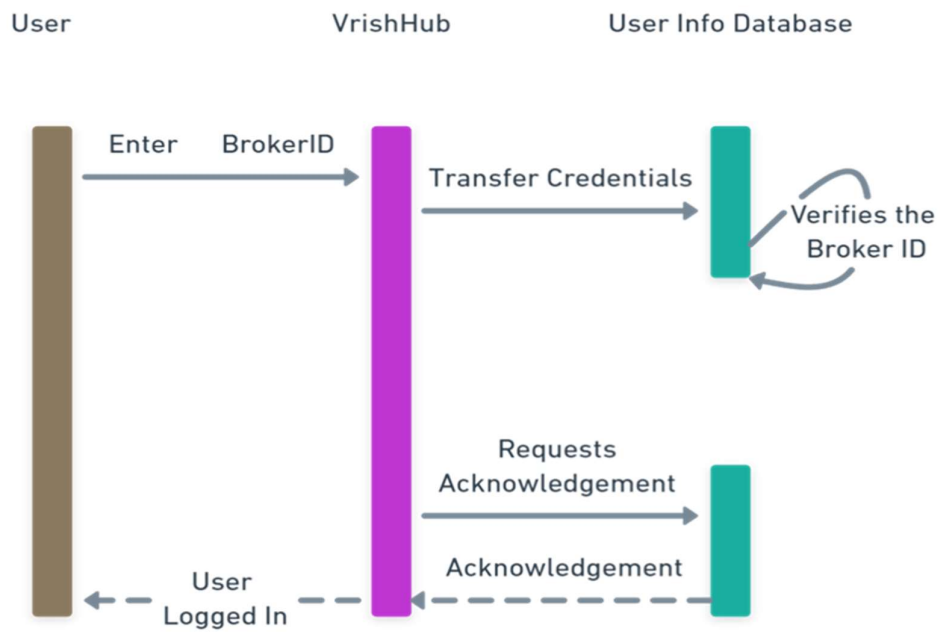


Fig.9. Log in Sequence Diagram

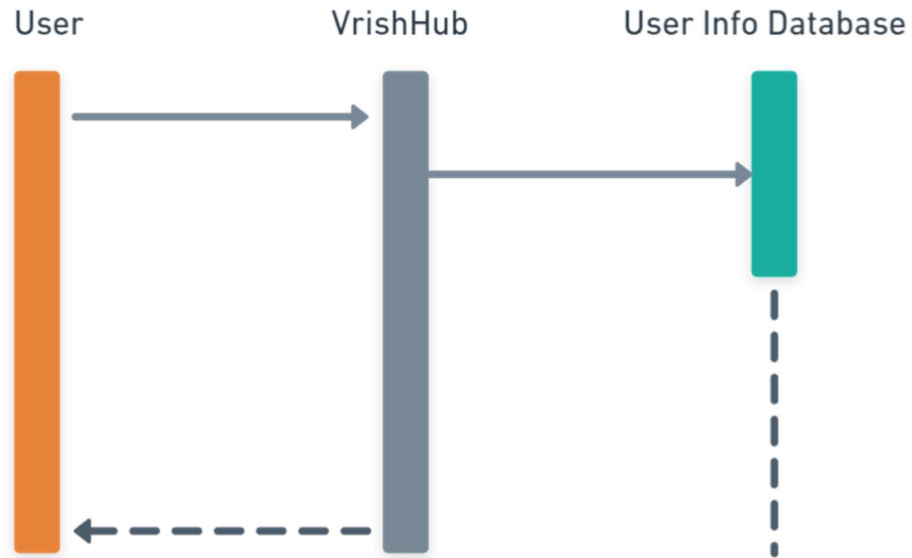


Fig.10 DashBoard View Sequence Diagram

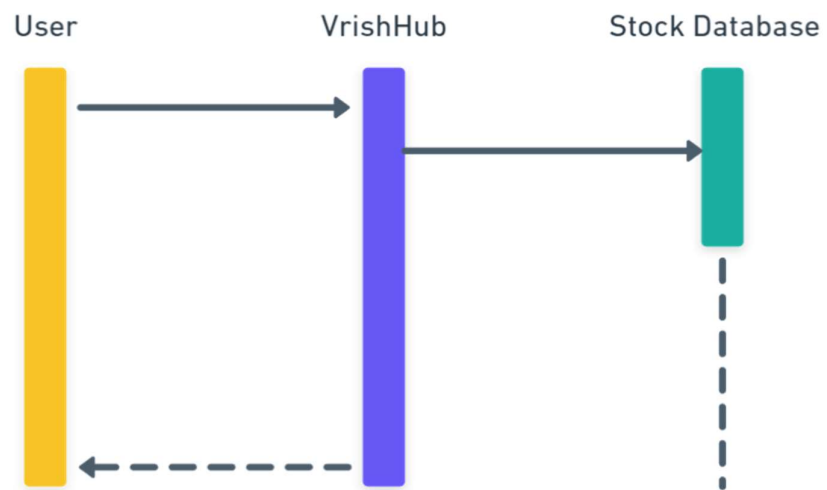


Fig.11 Browse Stocks Sequence Diagram

3.2.6 State Diagram

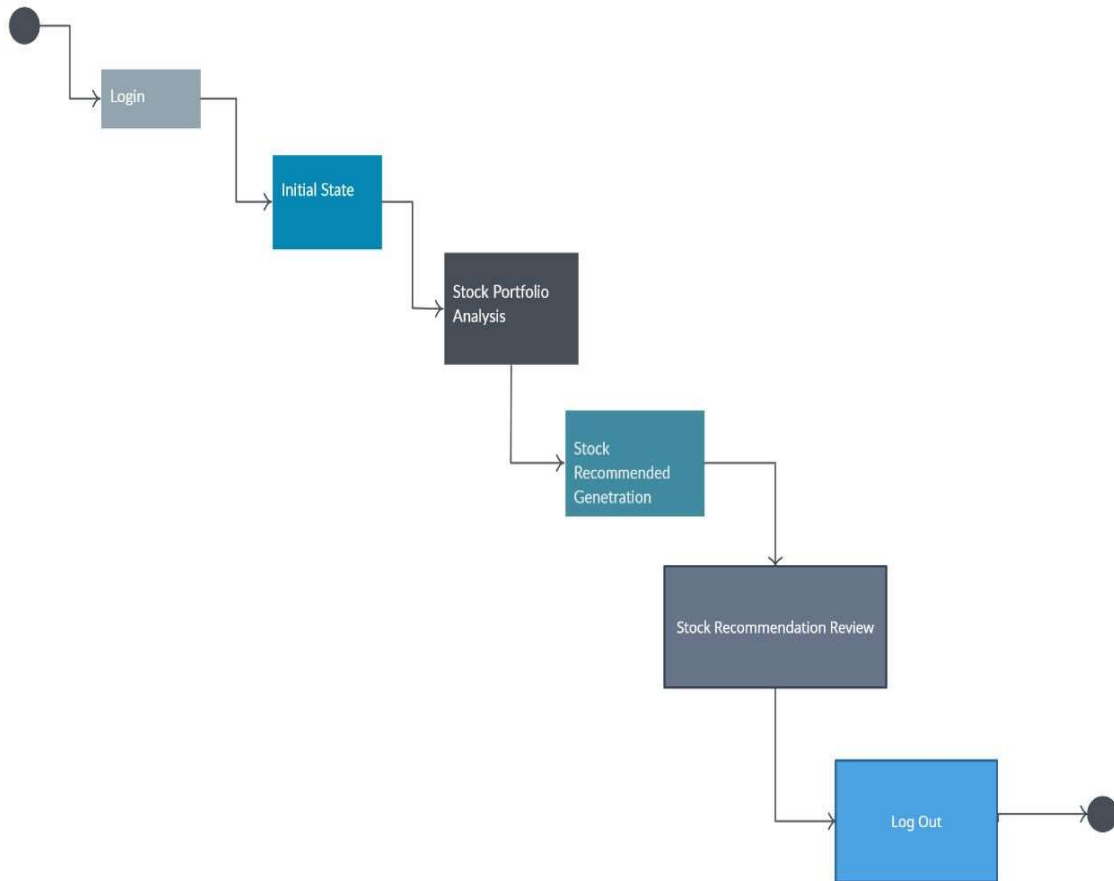


Fig.12.State Diagram

3.2.7 Class Diagram

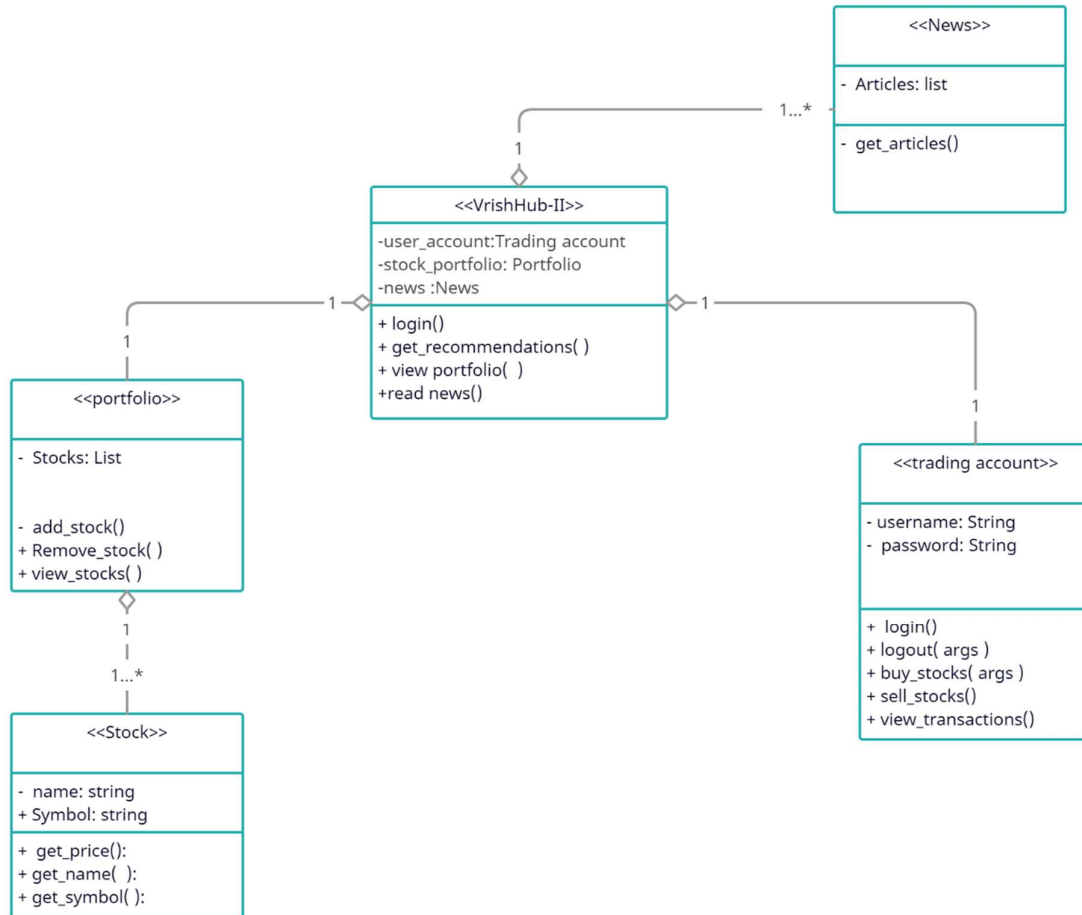


Fig.13.Class Diagram

3.2.8 Neural Network

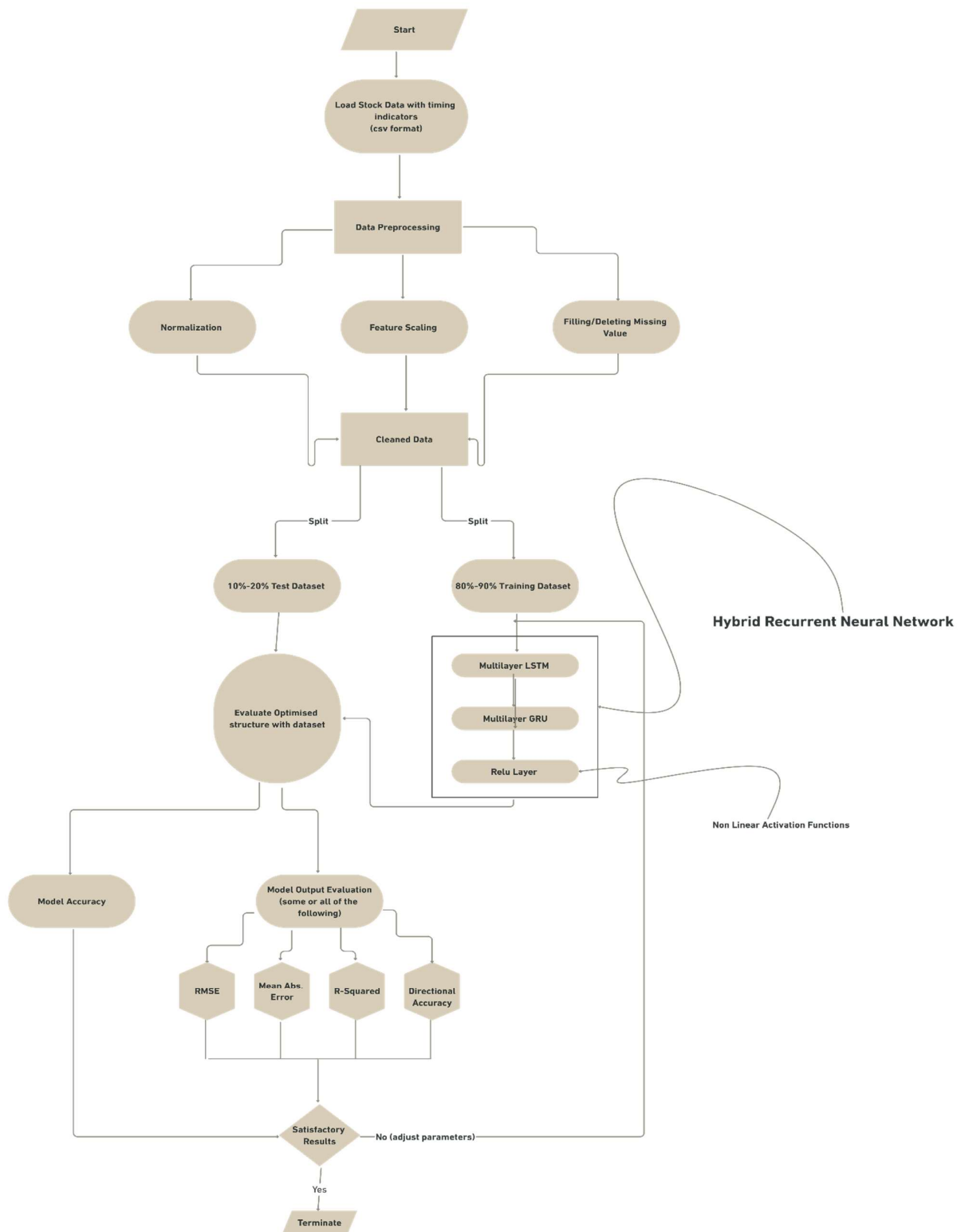


Fig.14.Neural Network

OUTPUTS

By the aid of our website's recommendation, the investors will be able to better judge and make effective decisions on whether to buy, hold or sell stocks. We have also provided an interactive dashboard so that all the investor's trading information is available at a single spot and they can manage their portfolio hassle freely.

Further to make our product more interactive and useful, we have also added the functionality of reading blogs etc. This will increase the time sent by each user to be investing.

We also worked upon the security of the login page where we added the feature of Two factor Authentication and also restricted multiple logins.

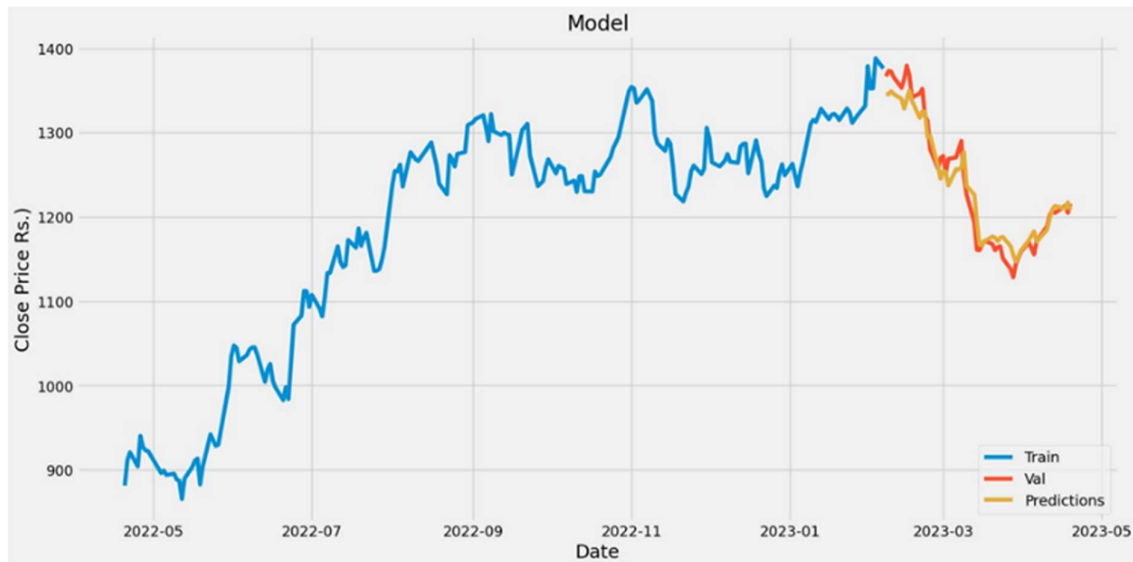


Fig.15 Output graph of the RNN showing the actual and predicted prices for the “M&M” (Mahindra and Mahindra stock) from 20th Apr. 2022 to 20th Apr 2023

CONCLUSIONS / RECOMMENDATIONS

We have described a real-time personalized stock trading and recommendation system that can tailor an advice for each user who has a wide variety of interests and backgrounds. To obtain a good recommendation output, all the information, rules, constraints, and techniques must be applied and integrated. We built a prototype and conducted experiments based on both real human subjects and synthetic subjects. The results support our objectives that recommendation system could successfully provide better suggestions to users. However, our objective of building such tool is to help an individual user be able to make his/her own decision effectively rather than making decision for them.

REFERENCES

1. J. Yoo et al., "An Adaptive Stock Tracker for Personalized Trading Advice," International Conference on Intelligent User Interfaces, pp. 123-145, 2003.
2. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, 2nd Edition
3. Panda et al (2011) "A study on the problems faced by investors towards online share trading", International Journal of Management Reviews 13 (2), 199-217, 2011.
4. Kasisomayajula (2012) "A study on the problems faced by investors towards online share trading," Kasisomayajula (2012), study was to understand the working of ... January 2006 Volume 5, Number 1 55.
5. Abdul Rahim (2013)"A study on the problems faced by investors towards online share trading," International Journal of Marketing, Financial Services& Management Research, ISSN-2277-3622, Volume-2, No.4 (April 2013) by Dr.A.Abdul Rahim.
6. Prabakaran (2017) "A study on the problems faced by investors towards online share trading," Prabakaran, V., (2017). A Study on Factors Influencing the Intention to Online Share Trading, International Journal of Recent Research and Applied Studies, Vol. 4, No.4 (29), April, pp. 120 - 126.
7. G. Webber, "A Learning System in the WWW," 17th International Conference on User Modelling, Banff, Canada, pp. 371-377, 1999.
8. C. Tasso and P. Giangrandi, "Managing Temporal Knowledge in Student Modelling," 6th International Conference on User Modelling, pp. 415-426, 1997.
9. P. Brusilovsky, "Adaptive Hypermedia," Modelling and User-Adapted Interaction, Vol. 11, nr. 1-2, pp. 87-110, 2001.
10. Shepherd et al., "Adaptive User Modelling for Filtering Electronic News," Proceeding of the 35th Hawaii International Conference on System Sciences IEEE, pp. 123-145, 2002.
11. J. Kay, "The User Model Toolkit for Cooperative User Modelling," User Modelling and User-Adapted Interaction, pp. 149-196, 1995.
12. Trade trek Enterprise and Trade trek Online, Available online:
<http://www.tradetrek.com>
13. T. Achelis, "Parameter Tuning in Trading Algorithms Using ASTA," Department of Computer Science, Umea, Sweden, 1995.

14. R.J. Kuo, "A Decision Support System for the Stock Market Through Integration of Fuzzy Neural Network and Fuzzy Delphi," Applied Artificial Intelligence, Vol. 12, pp. 501-520, 1998.
15. C.C. Tseng and P.J. Gmytrasiewicz, "Real Time Decision Support System for Portfolio Management," Operations Research, pp. 123-145, 1999.
16. J. Yoo et al., "An Adaptive Stock Tracker for Personalized Trading Advice," International Conference on Intelligent User Interfaces, pp. 123-145, 2003.
17. <https://venturebeat.com/data-infrastructure/what-is-firebase/>
18. <https://www.geeksforgeeks.org/?newui>