

STOCK PRICE PREDICTION USING MACHINE LEARNING

A REPORT

submitted by

Pragna Pulipati (17MIS1044)

in partial fulfilment for the award

of

M. Tech. Software Engineering (Integrated)

School of Computer Science and Engineering



VIT[®]

Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

MAY 2021



VIT[®]
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

School of Computer Science and Engineering

DECLARATION

I hereby declare that the project entitled “**Stock Price Prediction using Machine Learning**” submitted by me to the School of Computer Science and Engineering, Vellore Institute of Technology, Chennai Campus, Chennai 600127 in partial fulfilment of the requirements for the award of the degree of **Master of Technology - Software Engineering (Integrated)** is a record of bona-fide work carried out by me. I further declare that the work reported in this report has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma of this institute or of any other institute or university.

Signature

Pragna Pulipati (17MIS1044)



VIT[®]
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

School of Computer Science and Engineering

CERTIFICATE

The project report entitled “**Stock Price Prediction using Machine Learning**” is prepared and submitted by **Pragna Pulipati (Register No: 17MIS1044)**. It has been found satisfactory in terms of scope, quality and presentation as partial fulfilment of the requirements for the award of the degree of **Master of Technology – Software Engineering (Integrated)** in Vellore Institute of Technology, Chennai, India.

Examined by:

Examiner I

Examiner II



TechCiti Software Consulting Private Limited.

CIN: U72900KA2018PTC117376

D-U-N-S No. : 86 14 54180

No. 22 23 24 25/101, BNR Complex, J.P. Nagar, Bengaluru, Karnataka 560078.

Landline: 080 4162 8482 Email: info@techcitisoftware.in Website: www.techcitisoftware.in

Ref.No.TSCPL/2020-2021/HRD/INT2674

Date: 30th November, 2020

TO WHOMSOEVER IT MAY CONCERN

This is to certify that Ms. **Pragna Pulipati**(17MIS1044) from Vellore Institute of Technology, worked with us as **Software Developer -Intern** from **05th October, 2020** to **28th November, 2020** and has successfully completed her Academic Internship .

Title of Project: **Stock Price Prediction using Machine Learning**

Mode of Internship: Virtual / online

We wish her all the best for her future endeavors.

Sincerely,



Manager

Human Resources Department

TechCiti Software Consulting Private Limited

Registered office: No. 22 23 24 25/101, BNR Complex, J.P. Nagar 7th Phase, Bengaluru, Karnataka 560078.

Landline: 080 4162 8482 Email: info@techcitisoftware.in Web: www.techcitisoftware.in

ACKNOWLEDGEMENT

Throughout the writing of this dissertation, I have received a great deal of support and assistance. I would first like to thank my First Guide, **Ms. Kasturi**, whose expertise was invaluable in the completion of the project.

I would like to acknowledge my colleagues from my internship at **TechCiti Software Consulting Private Limited** for their wonderful collaboration. You supported me greatly and were always willing to help me. I would particularly like to thank all the members for your excellent cooperation and for all of the opportunities I was given to complete the project.

I would also like to thank my tutors, faculties of the university, **Dr. Asnath Vicky Phamila Y**, Head of the Department (HoD), MTech Software Engineering (5 year integrated), SCSE, VIT Chennai; **Dr. Jagadeesh Kannan R**, Dean of the School of Computer Science & Engineering, VIT Chennai; **Dr. Geetha S**, Associate Dean of the School of Computer Science & Engineering, VIT Chennai for their valuable guidance. You provided me with the tools that I needed to choose the right direction and successfully complete my dissertation.

In addition, I would like to thank my parents for their wise counsel and sympathetic ear. You are always there for me. Finally, there are my friends, who were of great support in deliberating over our problems and findings, as well as providing a happy distraction to rest my mind outside of my research.

CONTENTS

Chapter	Title	Page
	Title Page	i
	Declaration	ii
	Certificate	iii
	Industry certificate	iv
	Acknowledgement	v
	Table of contents	vi
	List of Figures	vii
	List of Abbreviation	viii
	Abstract	ix
1	Introduction	01
2	Literature Survey	03
3	Methodology	06
4	System Design and Architecture	08
5	Implementation	10
6	System Study	12
7	Conclusion	13
	References	14
	Appendix – I (Sample source code)	15
	Appendix – II (Working Model, Database and Dataset)	22

LIST OF FIGURES

Fig No.	Title	Page
1	System Architecture	08
2	Sequence Diagram	08
3	Entity Relationship Diagram	09
4	Use Case Diagram	09
5	Regression Model	10
6	Django Framework	10
7	Pair-plot	11
8	Bar graph between Trade Volume and Turnover	11

LIST OF ABBREVIATIONS

Abbreviation	Expansion
HTML	Hypertext Markup Language
CSS	Cascading Style Sheets
SVM	Support Vector Machine
LR	Linear Regression

ABSTRACT

In the financial world, stock trading is one among the foremost important activities. Stock market prediction is an act of trying to work out the end of the day value of stock and other financial instruments traded on a financial exchange. This project explains the prediction of a stock using Machine Learning. The technical and fundamental or the statistical analysis is employed by the foremost of the stockbrokers while making the stock predictions. The programming language is employed to predict the stock exchange using Machine Learning is Python. In this project, we propose a Machine Learning (ML) approach which can be trained from the historical stock data and gain intelligence that uses the acquired knowledge for an accurate prediction. In this context, the Machine Learning (ML) technique called Linear Regression is used to predict stock prices for the large and small capitalizations employing prices with both daily and up-to-the-minute frequencies.

In the existing system, we tend to propose that a company's performance, in terms of its stock worth movement, is foreseen by internal communication patterns. To get early warning signals, we tend to believe that it's vital for patterns in company communication networks to be detected earlier for the prediction of serious stock worth movement to avoid attainable adversities that an organization could face within the securities market in order that stakeholders' interests is protected the maximum amount as attainable. Despite the potential importance of such data regarding corporate communication, little or no work has been tired this vital direction.

Accuracy plays a crucial role in evaluating and predicting the available market. Although many algorithms are available for this purpose, selecting the foremost accurate one continues to be the elemental task in getting the simplest results. Hence, we have chosen the Linear Regression model.

Advantages:

- The successful prediction will maximize the advantage of the customer.
- The dataset used in this project is the stock data of State Bank of India from the BSE stock prediction history equity between the period 01/01/2020 and 31/12/2020.

Disadvantage:

- The results indicate that the stock price is unpredictable when traditional classifier is used.

INTRODUCTION

Why is Machine Learning used?

Machine Learning represents the synergy of statistics and computer programming. It enables model-building based on copious amounts of data without explicit commands. Prediction refers to the output of an algorithm after it has been trained on a historical dataset and applied to new data when forecasting the likelihood of particular outcome. Machine Learning model predictions allow businesses to make highly accurate guesses as to the likely outcomes of a question based on the historical data. These provides the business with insights that result in tangible business value.

What tools are used for developing this project?

Anaconda Distribution: It is a desktop GUI included in Anaconda distribution that allows us to launch applications and easily manage conda packages, environments, and channels without using command-line commands.

1. Jupyter Notebook: Jupyter is a free, open source, interactive web tool known as computational notebook, which researchers can use to combine software code, computational output, explanatory text and multimedia resources in a single document.
2. Spyder: It is the scientific python development environment, is a free integrated development environment (IDE) that is included with Anaconda. It includes editing, interactive testing, debugging, and introspection features.

Why Python?

Python works on different platforms like Windows, Mac, Linux, Raspberry, Pi, etc. It has a simple syntax similar to the regular English language. It has syntax that allows developers to write programs with fewer lines than some other programming languages. Python runs on an interpreter system, meaning that code can be executed as soon as it is written. This means that prototyping can be very quick. Python can be treated in a procedural way, an object-oriented way or a functional way.

Stock market plays a very important role in the fast economic growth of the developing country like India. So, our country and other developing nation's growth may depend on performance of

stock market. If stock market rises, then countries economic growth would be high. If the stock market falls, then the country's economic growth would be down. In other words, we can say that stock market and country's growth is tightly bounded with the performance of stock market. The prediction techniques in the stock market can play a crucial role in bringing more people and existing investors at one place. Machine Learning techniques play important role in stock market which can search uncover and hidden patterns and increasing the certain level of accuracy, where traditional and statistical methods are lacking. There is a huge amount of data generated by stock markets which forces the researchers to apply machine learning to make investment decisions.

To implement this project, we have used the following technologies:

1. User Interface: HTML and CSS

- 1.1. HTML: HTML is the standard markup language for creating web pages and web applications. Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages.

- 1.2. CSS: CSS is used to control the style of a web document in a simple easy way. CSS is a simple design language intended to simplify the process of making web pages presentable. We can reuse the same CSS code for multiple HTML pages.

2. Django: Django is an open source, web development framework present in python which is developed and maintained by the Django Software Foundation. Django is widely used because of its vast functionalities. It supports templates and static files that means we can easily render the HTML pages by putting all the HTML files in the directory called "templates" and similarly you can place all the files related to styles inside the directory called "static". In this project, Django is used for the front-end development. Further, Django provides more features such as built-in local host server and built-in administration facility as compared to other frameworks.
3. Python: It is general-purpose coding language which means that unlike HTML, CSS and JavaScript, it can be used for other types of programming and software development besides web development. That includes back-end development, software development, data science and writing system scripts among other things.

- 3.1. Pandas: Pandas is the library for the python language which is used for data manipulation and analysis of data. It provides some data structures and operations for the manipulation and analysis.

LITERATURE SURVEY

Stock market prediction using an improvised training algorithm of neural networks:

Predicting closing stock price accurately is a challenging task. Computer aided systems have been proved to be helpful for stock prediction such as Artificial Neural Networks, Adaptive Neuro Fuzzy Interface System etc. Latest research works prove that Adaptive Neuro Fuzzy Interface System shows better results than Neural Networks for stock prediction. In this paper, an improvised Levenberg Marquardt training algorithm of artificial neural network has been proposed. Improved Levenberg Marquardt algorithm of neural network can predict the possible day-end closing stock price with less memory and time needed, provided previous historical stock market data of Dhaka Stock Exchange such as opening price, highest price, lowest price, total share traded. Moreover, improvised Levenberg Marquardt algorithm can predict day-end stock price with 53% less error than Adaptive Neuro Fuzzy Interface System and traditional Levenberg Marquardt algorithm. It also requires 30% less time, 54% less memory than traditional Levenberg Marquardt and 47% less time, 59% less memory than Adaptive Neuro Fuzzy Interface System.

Research on Stock Price Prediction Method based on Convolutional Neural Network:

In order to meet the needs of the financial industry and the financial market, effectively improve the rate of return in funds and avoid market risks, this paper proposes a stock price prediction model based on Convolutional Neural Network, which has obvious self-adaptability and self-learning ability. Combining the characteristics of Convolutional Neural Network and Thai stock market, the dataset is trained and tested after pre-treatment. On this basis, three stocks (Bangkok Bank Public Company Limited, Charoen Pokphand All Public Company Limited and Petroleum Authority of Thailand Public Company Limited) listed on the Thai Stock Exchange are tested and compared with the actual stock price. The results show that the model based on Convolutional Neural Network can effectively identify the changing trend of stock price and predict it which can provide valuable reference for stock price forecast. The prediction accuracy is high, and it is worth further promotion in the financial field.

Price Trend Prediction of Stock Market using Outlier Data Mining algorithm

In this paper, they presented a novel data mining approach to predict long term behaviour of stock trend. Traditional techniques on stock trend prediction have shown their limitations when using time series algorithms or volatility modelling on price sequence. In this research, a novel outlier mining algorithm is proposed to detect anomalies on the basis of volume sequence of high frequency tick-by-tick data of stock market. Such anomaly trades always inference with the stock price in the stock market. By using the cluster information of such anomalies, their approach predicts the stock trend effectively in the really world market. Experiment results show that our proposed approach makes profit on the Chinese stock market, especially in a long-term usage.

Exploring Cluster stocks based on deep learning for Stock Prediction

Stock prediction is an important yet challenging problem in finance field. As the stock increases, cluster stocks have a certain on target stock, because the same type of stocks have a linkage effect. However, a large number of approaches only utilize their own information, and not use extra information from cluster stocks. In this paper, to improve the stock prediction, we exploit cluster stocks, and develop a deep learning model for stock prediction. A Long-Short Term Memory model is first introduced, then we introduce attention model. The combination Long-Short Term Memory and attention model are proposed for stock prediction by using cluster stocks. To validate the proposed approach, they collected the data from Wind Financial Terminal. Extensive experiments are carried out on this data and results show that exploring cluster stocks based on deep learning is a promising development that improves the performance of stock prediction, and the cluster stocks information can offer promising enhancement.

An Ensemble Stock Predictor and Recommender System

Stock market analysis and prediction tools have been prevalent for several years now with various techniques and models to predict stock market efficiency. This paper presents the design and implementation of a novel technique to predict stock market trends. The approach is an ensemble model which takes into account historical stock data, tweets and news affecting the stock prices of various companies and provides recommendations on which stocks to invest

in for a particular duration. The model is built on real historical stock dataset obtained from Indian National Stock Exchange (www.nseindia.com) over a 20-year period. The model uses Recurrent Neural Networks and Long-Short Term Memory to learn and predict the future stock trend for a basket of 50 companies with Root Mean Square Error of 1.43 stock units with respect to Indian currency. Finally, the predicted stock price within a duration is converted into a graphical image, which is passed to a Convolutional Neural Network classifier. The classifier is trained on finer features like peaks and troughs within the stock trend image, to provide recommendations on when to invest in a particular company stock. When tested with real stock prices over a week, it was found that the model was able to achieve extremely high accuracy in predicting the stock trends.

METHODOLOGY

A machine learning project may not always be a linear, but it has a number of well-known steps: Define Problem, prepare data, evaluate algorithms, Improve algorithms, Results

Type of Machine Learning Model Chosen:

Regression model: In the machine, learning regression is a set of problems where the output variable can take continuous values. For example, predicting the airline price can be considered as a standard regression task.

- LR – simplest baseline model for regression task, works well only when data is linearly separable and very less or no multi-collinearity is present.
- Lasso Regression – linear regression with L2 regularization
- Ridge Regression – linear regression with L1 regularization
- SVM Regression
- Decision Tree Regression etc.

Which model is the best among them?

It is always preferable to start with the simplest model applicable to the problem and increase the complexity gradually by proper parameter tuning and cross-validation. Cross-validation is more trustworthy than domain knowledge.

Modules:

Registration Module: The user can be able to register by giving their details and setting up the account user name and password.

Login Module: The user can login to the application using their credentials.

Input Module: The user can enter the input feature values in this module. This takes the input for the prediction required.

Stock Prediction Module: The stock prediction module displays all the input data which has been given for prediction along with the predicted Stock Turn Over value obtained by applying the trained model.

Functional Requirements:

1. Login
 - 1.1. Description: The user has to enter username and password
 - 1.2. Input: Username and password
 - 1.3. Processing: Validating the user
 - 1.4. Output: Home Screen
 - 1.5. Post Condition: A new session would be generated
2. Manage User Details
 - 2.1. Input: Details of the user
 - 2.2. Processing: Viewing details about user
 - 2.3. Output: Add the details to database
3. Admin Manage Admin
 - 3.1. Input: Details of the Admin
 - 3.2. Processing: Viewing details about Admin
 - 3.3. Output: Add the details to database
4. Admin Manage Prediction
 - 4.1. Input: Details of the Stock Prediction
 - 4.2. Processing: Adding details and query about prediction
 - 4.3. Output: Add the details to database
5. Manage Module
 - 5.1. Input: Details of the module
 - 5.2. Processing: Adding details about module
 - 5.3. Output: Add the details to database

Non-functional Requirements:

1. Adaptability: Web documents are changeable according to different condition like screen size difference, resolution difference, network speed difference, OS, etc.
2. Availability: Database connectivity is well defined so the available resources are provided and updated in recursive manner.
3. Maintainability: One can update the present project without affecting the ongoing or the deployed project.

SYSTEM DESIGN AND ARCHITECTURE

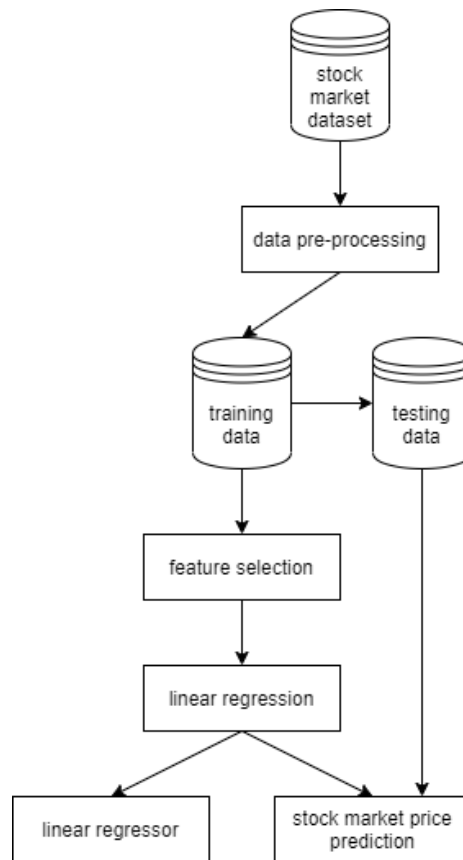


Fig 1. System Architecture

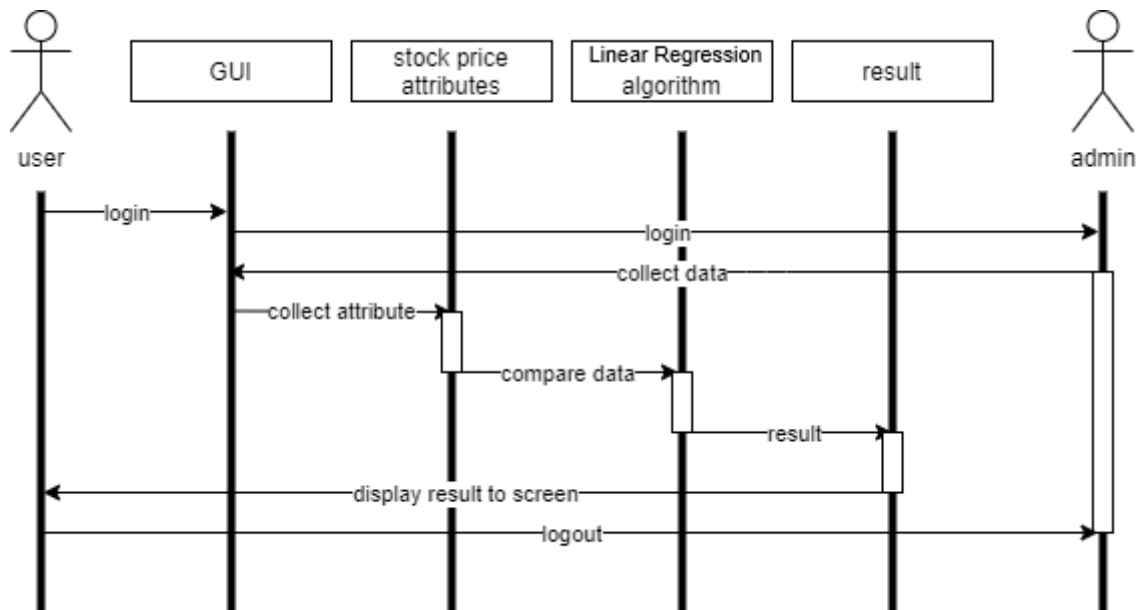


Fig 2. Sequence Diagram

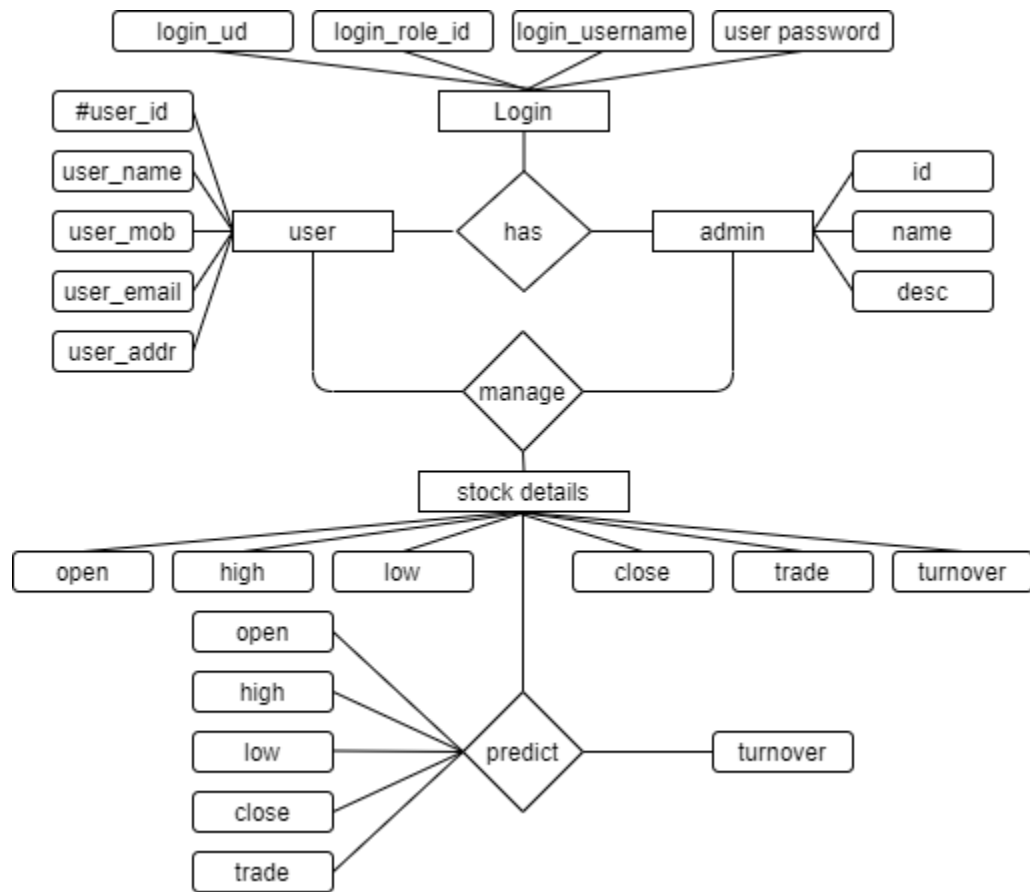


Fig 3. Entity Relationship Diagram

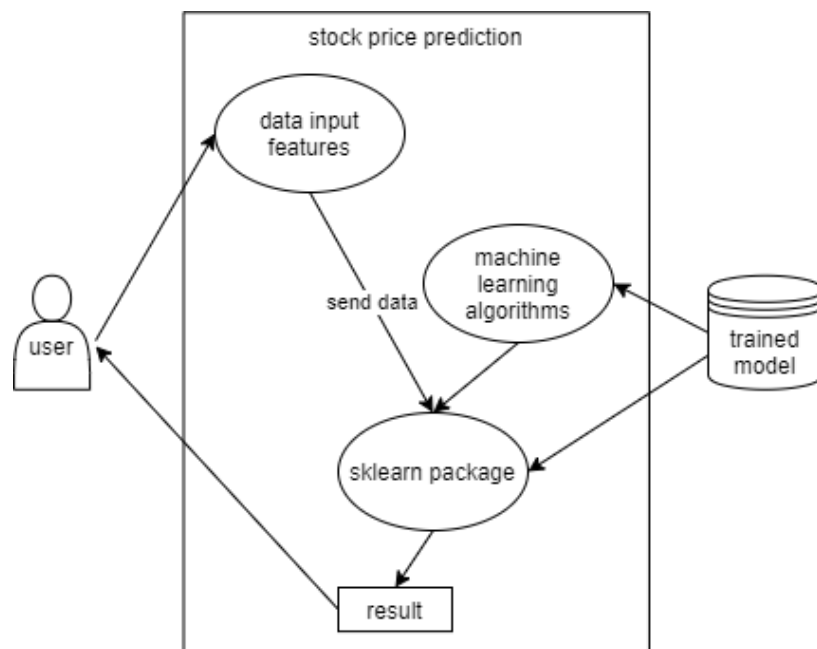


Fig 4. Use case Diagram

IMPLEMENTATION

Stock market prediction seems a complex problem because there are many factors that have yet to be addressed and it doesn't seem statistical at first. But by proper use of machine learning techniques, one can relate previous data to the current data and train the machine to learn from it and make appropriate assumptions. Machine learning as such has many models but this focuses on two most important of them and made the predictions using them.

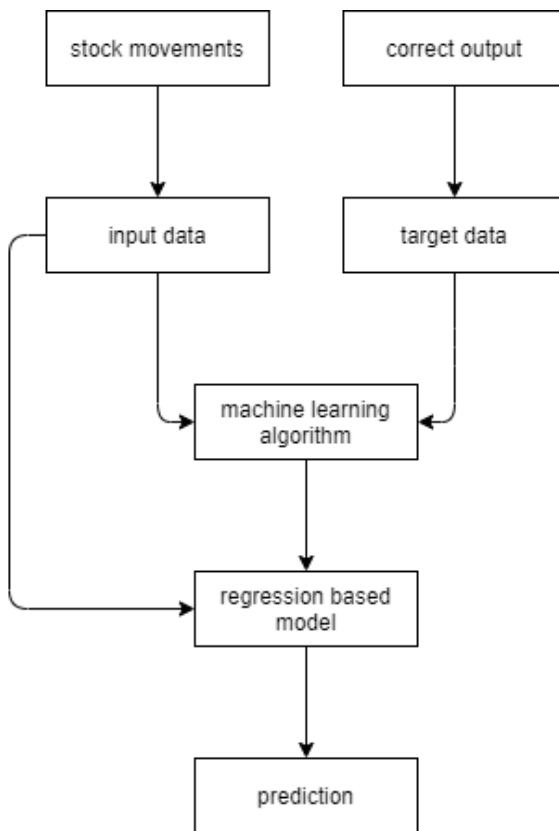


Fig 5. Regression Model

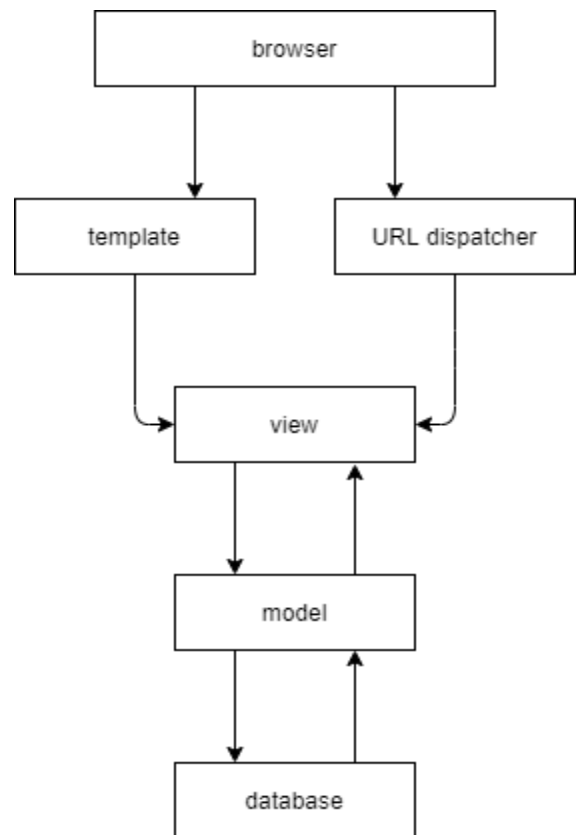


Fig 6. Django framework

Work was carried out on CSV format of data through pandas library and calculated the parameter which is to be predicted, the turnover price of the stocks. The data is divided into different training sets for cross-validation to avoid over fitting. The test set is kept as 40% of the whole dataset. Linear regression is performed on the data and then predictions are made, which are plotted to show the results of the stock market prices w.r.t total trade volume.

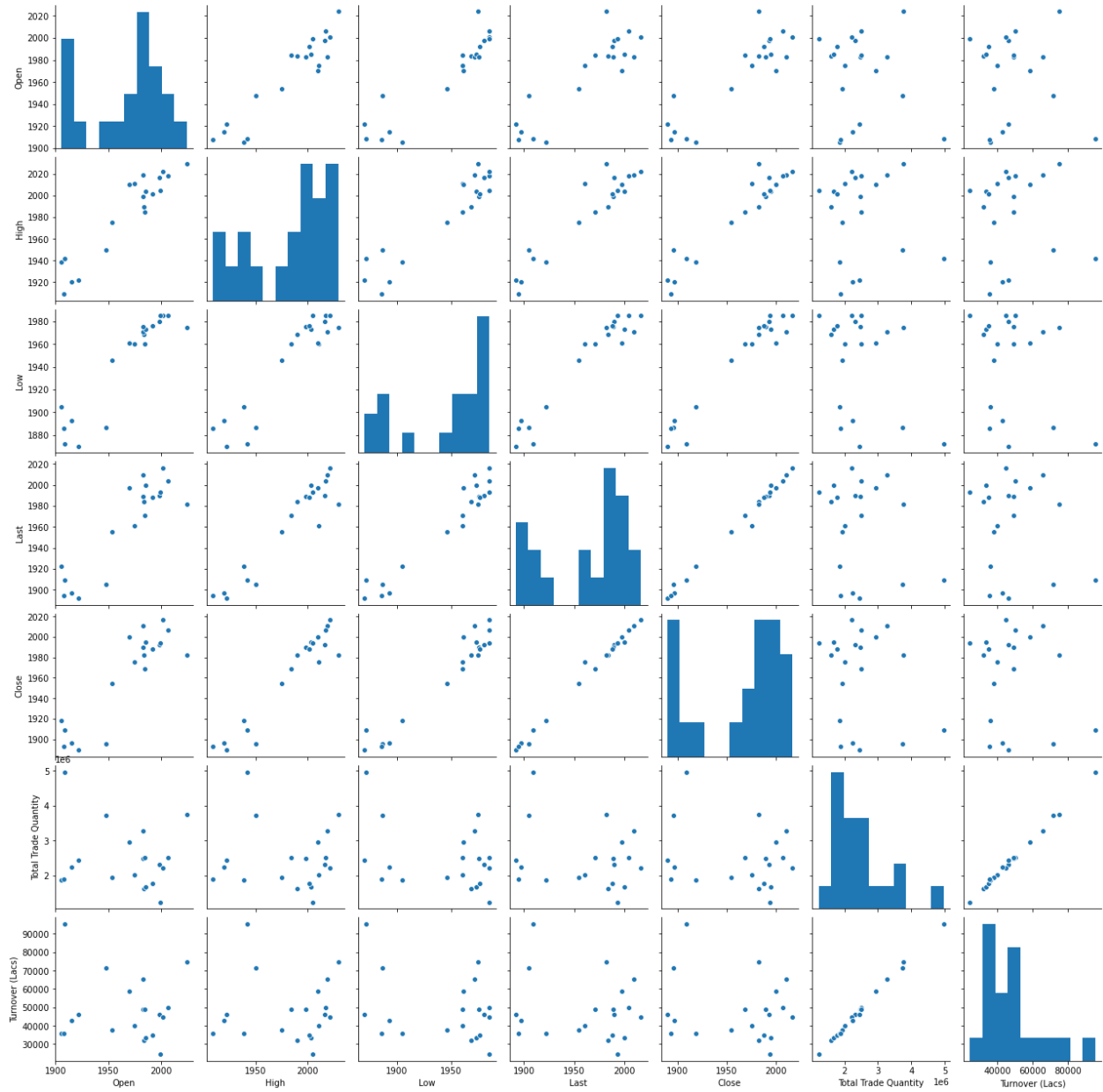


Fig 7. Pair plot

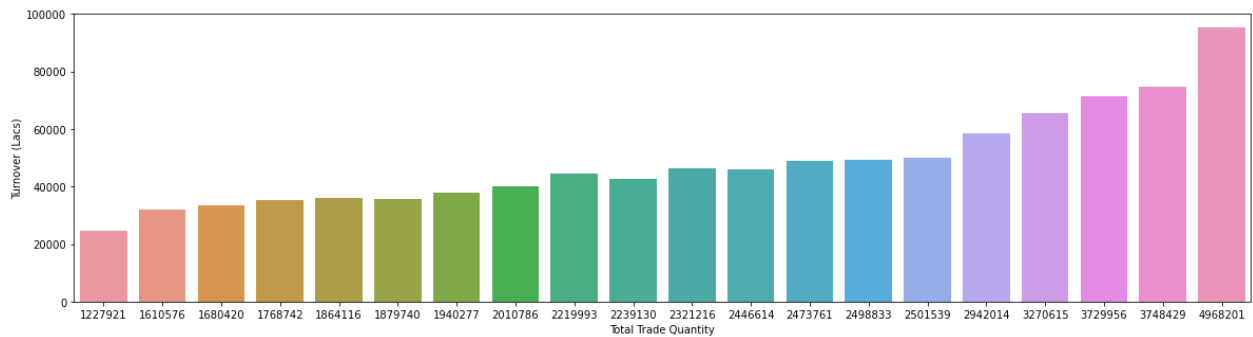


Fig 8. Bar graph (Trade Volume VS Turnover)

SYSTEM STUDY

Feasibility Study: The feasibility of the project is analysed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis, the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential. The considerations involved in the feasibility analysis are:

1. **Economic Feasibility:** This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into research and development of the system is limited. The expenditures must be justified. Thus, the developed system as well as within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.
2. **Technical Feasibility:** This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.
3. **Social Feasibility:** The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. Their level of confidence must be raised so that they are also able to make some constructive criticism, which is welcomed, as they are the final users of the system.

CONCLUSION

The aim of this project is to help the stock brokers and investors for investing money in the stock market. The prediction plays a very important role in stock market business which is very complicated and challenging process due to the dynamic nature of the stock market. In this project, I have developed a linear regression model which has shown an improvement in the accuracy of predictions, thereby yielding positive results. Use of recently introduced Machine Learning techniques in the stock price prediction have yielded promising results and thereby marked the use of them in profitable exchange schemes. It has led to the conclusion that it is possible to predict stock market with more accuracy and efficiency using machine learning techniques. In the future, the stock price prediction system can be further improved by utilizing a much bigger data set than the one being utilized currently. This would help to increase the accuracy of our prediction models. Furthermore, other models of Machine Learning could also be studied to check for the accuracy rate resulted by them.

REFERENCES

- [1] Abdulsalam, S. O., Kayode S. Adewole, and R. G. Jimoh. "Stock trend prediction using regression analysis—a data mining approach." (2011)
- [2] Chen, Shyi-Ming, and Yu-Chuan Chang. "Multi-variable fuzzy forecasting based on fuzzy clustering and fuzzy rule interpolation techniques." *Information sciences* 180, no. 24 (2010): 4772-4783.
- [3] Dash, Ambika Prasad. *Security analysis and portfolio Management*. IK International Pvt Ltd, 2009.
- [4] Dunham, Margaret H. *Data mining: Introductory and advanced topics*. Pearson Education India, 2006.
- [5] Fama, Eugene F. "The behavior of stock-market prices." *The journal of Business* 38, no. 1 (1965): 34-105.
- [6] Lai, Robert K., Chin-Yuan Fan, Wei-Hsiu Huang, and Pei-Chann Chang. "Evolving and clustering fuzzy decision tree for financial time series data forecasting." *Expert Systems with Applications* 36, no. 2 (2009): 3761-3773.
- [7] Larose, Daniel T., and Chantal D. Larose. *Discovering knowledge in data: an introduction to data mining*. Vol. 4. John Wiley & Sons, 2014.
- [8] Sharma, Ashish, Dinesh Bhuriya, and Upendra Singh. "Survey of stock market prediction using machine learning approach." In *2017 International conference of electronics, communication and aerospace technology (ICECA)*, vol. 2, pp. 506-509. IEEE, 2017.
- [9] Sheta, Alaa. "Software effort estimation and stock market prediction using takagi-sugeno fuzzy models." In *2006 IEEE International Conference on Fuzzy Systems*, pp. 171-178. IEEE, 2006.
- [10] Zarandi, MH Fazel, Babak Rezaee, I. B. Turksen, and Elahe Neshat. "A type-2 fuzzy rule-based expert system model for stock price analysis." *Expert Systems with Applications* 36, no. 1 (2009): 139-154.

APPENDIX I

urls.py

```
from django.urls import path
from . import views
urlpatterns=[
    path("",views.index,name='index'),
    path('register',views.register,name='register'),
    path('login',views.login,name='login'),
    path('data',views.data,name='data'),
    path('predict',views.predict,name='predict'),
    path('logout',views.logout,name='logout')
]
```

views.py

```
from django.shortcuts import render, redirect
from django.contrib import messages
from django.contrib.auth.models import User, auth
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeRegressor
def index(request):
    return render(request, 'index.html')
def register(request):
    if request.method == 'POST':
        first_name = request.POST['first_name']
        last_name = request.POST['last_name']
        username = request.POST['username']
        password1 = request.POST['password1']
        password2 = request.POST['password2']
        email = request.POST['email']
```



```

if password1 == password2:
    if User.objects.filter(username=username).exists():
        messages.info(request, 'Username Taken')
        return redirect('register')
    elif User.objects.filter(email=email).exists():
        messages.info(request, 'Email Taken')
        return redirect('register')
    else:
        user=User.objects.create_user(username=username,password=password1,
email=email, first_name=first_name, last_name=last_name)
        user.save();
        print('user created')
        return redirect('login')
    else:
        messages.info(request, 'password not matching')
        return redirect('register')
    return redirect('/')
else:
    return render(request, 'register.html')
def login(request):
    if request.method == 'POST':
        username = request.POST['username']
        password = request.POST['password']
        user = auth.authenticate(username=username, password=password)
        if user is not None:
            auth.login(request, user)
            return render(request, 'data.html')
        else:
            messages.info(request, 'invalid credentials')
            return redirect('login')
    else:

```

```

        return render(request, 'login.html')
def data(request):
    return render(request,"data.html")
def predict(request):
    if (request.method == 'POST'):
        open = request.POST['open']
        high = request.POST['high']
        low= request.POST['low']
        close = request.POST['close']
        trade=request.POST['trade']
        df = pd.read_csv(r"static/datasets/Stock.csv")
        df.dropna(inplace=True)
        df.isnull().sum()
        X_train = df[['Open','High','Low','Close','Total Trade Quantity']]
        Y_train = df[['Turnover (Cr)']]
        tree = DecisionTreeRegressor()
        tree.fit(X_train, Y_train)
        prediction = tree.predict([[open,high,low,close,trade]])
        return render(request, 'predict.html',
                        {"data":prediction,'open':open,'high':high,'close': close,"low":low,'trade':trade})
    else:
        return render(request, 'predict.html')
def logout(request):
    return render(request,"logout.html")

```

models.py

```

from django.db import models
class StockData(models.Model):
    company_name=models.CharField(max_length=90)
    Open_Year=models.IntegerField()
    Last_Year=models.IntegerField()

```

```
High_Profit=models.IntegerField()
Low_Profit=models.IntegerField()
Trade_Amt=models.IntegerField()
```

apps.py

```
from django.apps import AppConfig
class UserConfig(AppConfig):
    name = 'User'
```

admin.py

```
from django.contrib import admin
from .models import StockData
admin.site.register(StockData)
```

asgi.py

```
import os
from django.core.asgi import get_asgi_application
os.environ.setdefault('DJANGO_SETTINGS_MODULE', 'Stockpred.settings')
application = get_asgi_application()
```

settings.py

```
import os
BASE_DIR = os.path.dirname(os.path.dirname(os.path.abspath(__file__)))
SECRET_KEY = '4g4hfk=r&)3ti(4i-kug_bwyu*++bajhqh8gdi1vmb*38(!1^('
DEBUG = True
ALLOWED_HOSTS = []
INSTALLED_APPS = [
    'User.apps.UserConfig',
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
```

```

'django.contrib.sessions',
'django.contrib.messages',
'django.contrib.staticfiles',
]
MIDDLEWARE = [
'django.middleware.security.SecurityMiddleware',
'django.contrib.sessions.middleware.SessionMiddleware',
'django.middleware.common.CommonMiddleware',
'django.middleware.csrf.CsrfViewMiddleware',
'django.contrib.auth.middleware.AuthenticationMiddleware',
'django.contrib.messages.middleware.MessageMiddleware',
'django.middleware.clickjacking.XFrameOptionsMiddleware',
]
ROOT_URLCONF = 'Stockpred.urls'
TEMPLATES = [
{
'BACKEND': 'django.template.backends.django.DjangoTemplates',
'DIRS': [os.path.join(BASE_DIR, 'templates')],
'APP_DIRS': True,
'OPTIONS': {
'context_processors': [
'django.template.context_processors.debug',
'django.template.context_processors.request',
'django.contrib.auth.context_processors.auth',
'django.contrib.messages.context_processors.messages',
],
},
},
]
WSGI_APPLICATION = 'Stockpred.wsgi.application'
DATABASES = {

```

```

'default': {
    'ENGINE': 'django.db.backends.sqlite3',
    'NAME': os.path.join(BASE_DIR, 'db.sqlite3'),
}
}
AUTH_PASSWORD_VALIDATORS = [
    {'NAME': 'django.contrib.auth.password_validation.UserAttributeSimilarityValidator'},
    {'NAME': 'django.contrib.auth.password_validation.MinimumLengthValidator'},
    {'NAME': 'django.contrib.auth.password_validation.CommonPasswordValidator'},
    {'NAME': 'django.contrib.auth.password_validation.NumericPasswordValidator'},
]
LANGUAGE_CODE = 'en-us'
TIME_ZONE = 'UTC'
USE_I18N = True
USE_L10N = True
USE_TZ = True
STATIC_URL = '/static/'
STATICFILES_DIRS=[
    os.path.join(BASE_DIR, 'static')
]

```

urls.py

```

from django.contrib import admin
from django.urls import path,include
urlpatterns = [
    path("",include("User.urls")),
    path('admin/', admin.site.urls),
]

```

wsgi.py

```

import os

```

```
from django.core.wsgi import get_wsgi_application
os.environ.setdefault('DJANGO_SETTINGS_MODULE', 'Stockpred.settings')
application = get_wsgi_application()
```

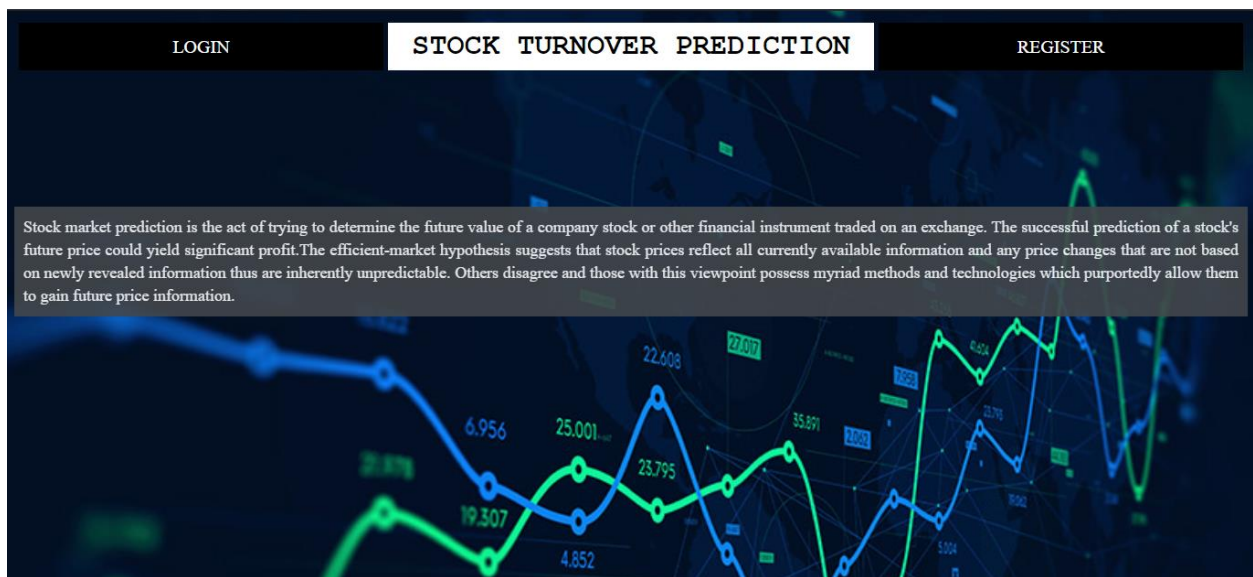
manage.py

```
import os
import sys
def main():
    os.environ.setdefault('DJANGO_SETTINGS_MODULE', 'Stockpred.settings')
    try:
        from django.core.management import execute_from_command_line
    except ImportError as exc:
        raise ImportError(
            "Couldn't import Django. Are you sure it's installed and "
            "available on your PYTHONPATH environment variable? Did you "
            "forget to activate a virtual environment?"
        ) from exc
    execute_from_command_line(sys.argv)
if __name__ == '__main__':
    main()
```

APPENDIX II

	A	B	C	D	E	F	G
1	Open	High	Low	Close	Total Trad	Turnover (Cr)	
2	275	277.25	272.55	274.75	1025892	28.23523	
3	277.85	278.5	272.6	276.85	1535034	42.24901	
4	276.2	279.75	274.7	277.9	2116003	58.75146	
5	268.25	275.85	268.25	275.15	1468835	39.97151	
6	264.6	268.8	264.6	266.75	2008059	53.60711	
7	257.6	263.85	255.5	263.05	1610796	42.0628	
8	254	258.55	248.15	257.55	1799859	45.70934	
9	267.3	269	251.1	254.6	2450581	63.87782	
10	267.7	271.9	264.9	271.4	1044431	27.99396	
11	270.4	272.1	266.8	267.6	5563140	149.6557	
12	272.2	273.55	268.55	269.45	1152799	31.1856	
13	274.25	274.25	268.3	270.6	1833421	49.56846	
14	273.4	276.3	272.45	274.25	1377776	37.8497	
15	270	276.15	270	272.45	3745846	102.3199	
16	269.45	270.8	265.55	269.6	1757355	47.0827	
17	272	274.85	269.25	270.35	3102170	84.38987	
18	270.35	273.65	265.2	271.95	4336216	117.1361	
19	263.45	271.35	263.1	269.7	3212445	85.60153	
20	256.5	264.2	253.75	263.45	4913756	127.3447	
21	249	259	248.05	256.35	4348434	110.5249	

Screenshot 1: Dataset (It contains the historic stock price values of State Bank of India the time period 01/01/2020 and 31/12/2020)



Screenshot 2: Home Page (This is the first page that the user gets to see when the project is executed, it contains information about Stock Price Prediction & the Register and Login buttons)

A registration form titled "REGISTRATION" is centered over a background of Indian 500 Rupee banknotes. The form includes input fields for Firstname, Lastname, Username, Email, Password, and Confirm Password, followed by a Submit button.

REGISTRATION

Firstname _____

Lastname _____

Username _____

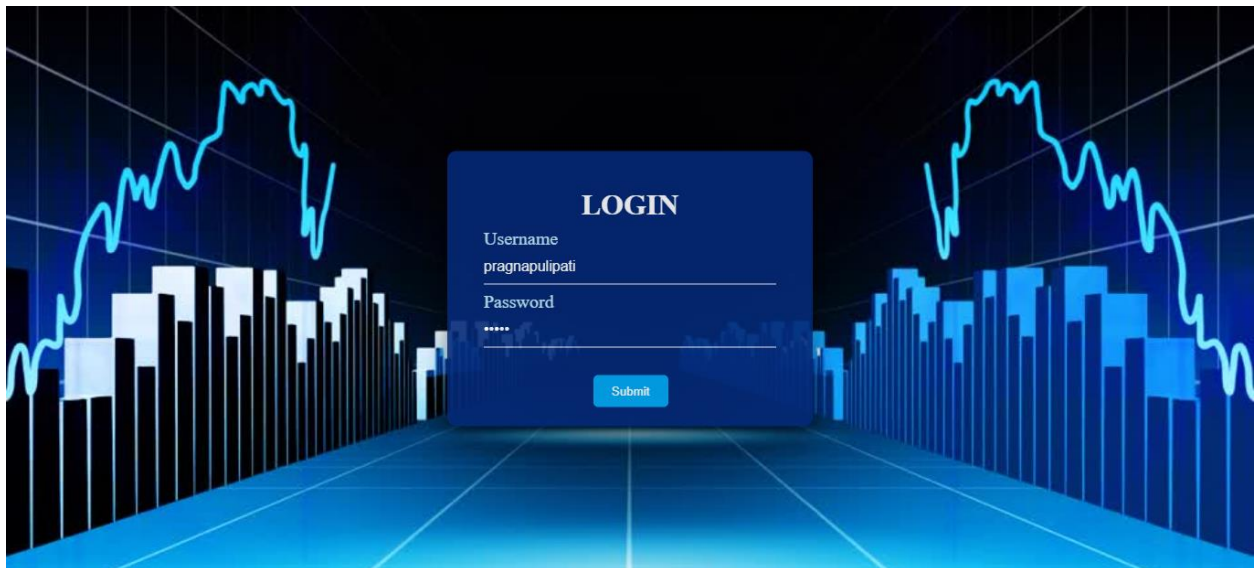
Email _____

Password _____

Confirm Password _____

Submit

Screenshot 3: Registration Page (The user can enter their details for creating an account in the Stock Price Prediction portal by giving their name, desired username, email ID and password)

A login form titled "LOGIN" is centered over a background of a stock market chart with a blue line graph and a bar chart. The form includes input fields for Username (pre-filled with "pragnapulipati") and Password (masked with dots), followed by a Submit button.

LOGIN

Username
pragnapulipati _____

Password
..... _____

Submit

Screenshot 4: Login Page (This allows the user to safely log in to the portal and proceed with the prediction process with appropriate username and password)

PREDICTION FORM

Open:

High:

Low:

Close:

Trade:

Screenshot 5: Prediction Form (The user can enter the various values such as Open, High, Low, Close and Trade Volume which are required for performing the prediction of Turn Over)

Stock Market Prediction

Hello Pragna!

Open 2022.9

High 2004.9

Close 1994.3

Low 2007.3

Trade 37895

Predicted TurnOver: [71.47]

[Log Out](#)

Screenshot 6: Result (This page contains the input data of Open, High, Low, Close and Trade and the result predicted in terms of lakhs)

