Data Visualization

J Component Project Report

Real-Time Stock Analyzer

Submitted by

Henil Jayesh Thakor (18BCE0934)

Mrinmay Date (18BIS0147)

Harshith Chukka (18BCE0941)

Under the guidance of

Prof. Nalini N

VIT®

Vellore Institute of Technology

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

Wnter Semester 2020 -2021

Abstract

It is essential to conduct research prior to making an investment. You can only make assumptions about an investment's worth and future performance after conducting thorough study. Even if you're following stock trading recommendations, it's a good idea to do some research to be sure you're making an investment that will pay off handsomely. Market participants use stock analysis to make purchasing and selling choices. Investors and traders try to obtain an insight in the markets by making rational choices by analysing and examining historic and current data. Fundamental and technical analysis are the two most common methods of stock analysis. Bank documents, economic statistics, firm assets, and market share are among the data sources used in fundamental research. Technical analysis is the second type of stock analysis. Technical analysis examines previous and current market action in order to forecast the likelihood of future price fluctuations. So, we will be making a real-time stock analyser which predicts stock prices of certain companies and also represent them graphically on a website.

Keywords: Stock Market, Long-Short Term Memory Networks, Closing Price, Time Series Dataset.

Introduction

i. Motivation

Investors can use share market research to determine a safety's intrinsic value prior dealing in it. Experts conduct rigorous investigation before forming any share market recommendations. Analysts and investors endeavour to predict future activities of a certain item, industry, or trade.

It is essential to conduct detailed research prior to making an investment. One may only assume things about an investment's worth and potential progress after conducting comprehensive study. Even if one is following share market recommendations, it's a good idea to conduct some analysis to be sure one is making an investment that will pay off handsomely.

When people deal in stock, they buy a few of the organization's shares with the expectations of making profit from the organization's increased worth. People perform qualitative investigation and quality of everything they buy, whether it's a house or a computer. It's the same with an investment. Investors are willing to invest their hard-earned income, therefore they must have a good understanding of what they are investing in.

ii. Significance

When it comes to data science, stock market analysis and prediction is usually one of the most popular topics. It is essential to conduct research prior to investing money. You can only make assumptions about an investment's worth and future results after conducting thorough study. Even if you're following stock trading recommendations, it's a good idea to do some research to be sure you're making an investment that will pay off handsomely. When you invest in equity, you buy some of a company's stock in the hopes of profiting from the company's increased worth. You do some study on the performance and quality of everything you buy, whether it's a car or a phone. It's the same with an investment. You are ready to invest your hard-earned money, therefore you must have a good understanding of what you are doing. By looking at the financial reports for a stock, you can determine whether the firm is solid, growing, and has a bright future. There are just too many people who invest in struggling businesses in the hopes of a turnaround. Stocks of firms that are already performing well and have a good foundation for ongoing growth are frequently the greatest investments.

iii. Scope and Application

Prior to embarking on a business venture, it is imperative that you conduct study. It is only after a thorough investigation that you may form some suspicions about a venture's worth and potential for future success. Regardless of whether you're following stock trading recommendations, it's a good idea to do some research to ensure you're making a decision that will yield the best results.

Companies seeking to raise funds for expansion use the stock market as their primary source of capital. It can also aid in the launch of new products and the repayment of debts.

Any changes in the currency exchange rate have an impact on the funds of a partnership, as well as the global economy. In the United States, the securities exchange serves as a financial benchmark. When a speculator observes that a company is doing well financially, they are more likely to buy more stocks, capital investments, or stock common assets.

That is why conducting a financial exchange evaluation before to investing money is critical. Depending on corporate security recommendations and articles isn't enough to ensure that you'll make the best purchasing and selling decisions.

Reasons to perform Stock Market Analysis are as follows:

- Get to know the business.
- Learn about the company's financial statements.
- Find out how much money the company owes.
- Examine the company's rivals.

iv. Drive to the proposed work

The below mentioned the few of the significant reasons for developing an easy-to-use stock market price analyser.

- Patterns in the market tend to replicate themselves.
  The underlying reason for technical analysis is that the market reflects all of the facts that needs to be known. As a result, there's little use in attempting to discern between undervalued and overvalued equities. Stock prices are assumed to reflect all known and unknown data, therefore prices are merely arbitrary movements. As a result, the best method to evaluate markets is to concentrate on previous instances and use these instances to infer what will happen in the future.

- The right entrance and departure can make a big difference.
  The accurate time of the investor to invest in a particular stock, seeing its potential growth value and the perfect time to sell those bought stocks does matter a lot in the field of share market to maximize the profit.

- Technical are the best at capturing trend and momentum.

  There are a variety of patterns to be on the lookout for. For example, there is a rising pattern, a downward pattern, a sideways pattern, a pattern that does not show, and so on. Technical indicators assist you in recognising patterns and trading accordingly. If the market is trending upward, for example, you should use each dunk to become entangled with the market or the stock. If, on the other hand, the market is trending downward, you should take advantage of each upward movement to sell the market. However, how would you determine whether the pattern is favourable or negative? That is where energy can be put to good use.

- Guides to choosing the right derivative strategy.

  Above all, the technological blueprints enable us to fine-tune our subordinates' systems. When the energy is good, long prospects and calls can be used to play a more decisive pattern. When the force is negative, however, financial specialists must ensure that their long positions in the money market are adequately sustained. When the overall market energy is focused on an unpredictably volatile market with no discernible direction, the dealer can use risky tactics such as rides and chokes on the long side. When the market is likely to be range-bound, an oppositional technique can be used. Market vitality, as well as backings and resistance levels, are important factors in fine-tuning the destinies and alternative strategies that dealers can use to get the most out of stock trade patterns.

v. Reason for choosing LSTM and it Working

LSTM (Long Short-Term Memory) is a type of artificial Recurrent Neural Network (RNN) architecture from the subject of Deep Learning.

Because there might be gaps of uncertain duration between critical occurrences in a time series data set, Long Short-Term Memory was chosen above other algorithms. When contrasted to a basic recurrent neural network, an LSTM cell has a cell memory unit. The cell vector can incorporate the idea of losing part of its pre - stored memory while also adding part of fresh data.

A neuron, an input gate, an output gate, and a forget gate make up a typical LSTM unit. The three gates pass the information into and out of the neuron, and the neuron remembers values across variable time periods. The details of the three gates are given below.

- Input Gate: Controls whether the memory cell gets updated.

$$i^{(t)} = \sigma\left(W^i\left[h^{(t-1)}, x^{(t)}\right] + b^i\right)$$

- Forget Gate: Controls whether the memory cells get reset.

$$f^{(t)} = \sigma\left(W^f\left[h^{(t-1)}, x^{(t)}\right] + b^f\right)$$

- Output Gate: Control whether the info of the current cell state is visible.

$$o^{(t)} = \sigma\left(W^o\left[h^{(t-1)}, x^{(t)}\right] + b\right.$$

2. Literature Survey and Existing Works

Many analysts and scholars have long been interested in stock market forecasting. According to popular belief, stock markets are fundamentally random walks, and attempting to anticipate them is a fool's game. The market operates like a ballot machine in the near term, but like a weighing machine in the longer term, allowing for the prediction of market volatility over a longer timescale. The use of machine learning methods and other algorithms to analyse and predict share prices is a field with a lot of potential. The first section of this study [1] presented a brief overview of share prices and a taxonomy of share price prediction systems. They next go over some of the research accomplishments in market analysis and forecasting. Then we spoke about how to analyse stocks using technical, fundamental, short-term, and long-term techniques. Finally, several problems and research prospects in this sector were addressed.

An understanding of share price patterns has long been a source of fascination for individuals who benefit financially from buying stocks. The share market is often associated with a greater risk and massive profits. Despite the fact that there is a large pool of potential traders, just a small percentage of them would have invested in the share market. The fundamental cause for this is the incapacity of inexperienced risk takers to take calculated risks. Investors desire to save their investment despite its low rewards. One of the major causes of this issue is that they lack suitable direction in creating their strategy. The paper [2] focused on the real-world situation in this study, and have used three indices: CNX Realty, BANK NIFTY, and MIDCAP 50. The study is solely based on information gathered over the previous three years. Time series perception is an information retrieval approach that is used in predictive analytics to depict the ups and downs of a certain indicator. Correlations and Beta are two methods that provide information about a share price risk. The correlations tool is designed to determine the specific link between some of the indices and the firm. This Beta is used to determine the share price risk level.

Analysing share price changes will be of interest to a share market enthusiast. The share market's rates vary owing to the market's constant trading. When it comes to interpreting stock value changes, there have been fundamentally 2 techniques. The two types of approaches as mentioned in paper [3] are: fundamental and technical approach. These techniques have same goal of obtaining a strong return on investment by purchasing at a less price and selling at a higher cost. It is possible to say that the final objective of these two strategies is the same. The underlying principles of these two techniques, on the other hand, are vastly different. The underlying aspects would be of interest to the researcher in behavioural finance. He/she would want to know how to calculate a stock's real value or inherent value depending on its present and future earnings potential. When the stock's market value falls below its inherent worth, they will acquire it.

The word "technical analysis" refers to a wide range of investment strategies. The examination of market data and a several other associated summarized facts regarding stock exchanges is used in technical analysis to predict possible values. The pattern of price fluctuations is often a worry for a strategic investor. Predictive analytics is the academic discipline of using graphs to plot share prices data such as market volatility, trade volume, and market scenarios in order to anticipate potential price patterns. It can venture capitalists in predicting what is ‘feasible' to occur to values in the near future. By analysing business turning moments, it is also possible to determine the underlying millions of stocks or even if the shares are cheap or overrated. This research paper [4] aims to apply statistical analytical tools to a few routines in order to help investors make more accurate financial decisions in the Local share market. The research is entirely dependent from various sources gathered from the National Stock Exchange's (NSE) website, periodicals, and publications. The Moving Average Convergence

Divergence (MACD) analysis was conducted to assess whether or not a software is detail oriented. This allows the investors to see the recent situation and hazards connected with the stock at the very same time as the rest of the economy. This study intends to conduct technical analysis of chosen businesses' assets in order to aid investment choices in the share market. In a prejudiced environment, technological research offers impartial answers.

This has never been profitable to cast in a group of investments; the abnormalities of the stock sector prevent modelling techniques from accurately predicting potential stock prices. Machine learning, which entails teaching computers to execute activities that would ordinarily need human understanding, is the contemporary science study popular issue as seen in [5].

To anticipate stock price, a number of data mining methods have been developed. The employment of neural networks, genetic algorithms, associations, decision trees, and fuzzy systems is common. Pattern identification is also useful for stock market forecasting, because public mood is linked to stock price forecasting. There is some sort of link between them. Over the last two decades, a review of prior studies on stock price forecasting has revealed the widespread usage of technical indicators combined with artificial neural networks (ANNs) for stock market prediction. One study used a backpropagation neural network with technical indicators to estimate fuzzy time series, and the results revealed that the ANN outperformed the time series model. Association Rule Mining (ARM), Clustering, and Decision Tree for financial forecasting are three of the most well-researched and important algorithms in the data mining field. The creation of strong rules is a crucial aspect of data mining. The goal of association rule mining is to find connections between item sets in data repositories.

Now, the Apriori method is used to find common item sets and generate rule sets from them. Decision trees are ideal for making financial decisions that need the consideration of a large amount of complex data. They give a useful framework for laying out and evaluating alternative alternatives as well as the consequences of those actions. They also create an accurate and balanced picture of the risks and benefits of a given decision. The challenge of identifying association rules was initially discussed in this study in 1993, and an algorithm for mining association rules termed AIS was proposed. Many algorithms for rule mining have been suggested in the last fifteen years, as well as one novel methodology for processing stock data called Granule mining, which minimises the width of transaction data and generates association rules.

The [2] paper proposed fragment-based mining, which focuses on lowering the time and space complexity of data processing in association rule mining techniques. The fragment-based technique, like granule mining, divides data sets into fragments for processing, lowering the size of data sets provided to the algorithm. In contrary to granule mining, the condition and decision attributes are summed in fragment-based mining to provide generalised association rules. With the use of a Genetic Algorithm technique, [4],[5] study has offered a novel strategy to produce association rules(E-Rules), namely providing faster production of frequent item sets to offer interesting and useful rules in an effective and optimised way.

According to the preceding literature study, technical indicators have been widely used using various data mining approaches, but fundamental indicators have been studied in only a few cases. The significance of basic analytical variables such as the price-earnings ratio, moving average, and rumours has been completely overlooked. However, according to another article, using hybridised parameters produces better and more accurate results than using simply one type of input variable. [11] proposes a predictive model that has the potential to improve the quality of stock market decision-making by providing more accurate stock prediction using hybrid parameters. They employed ANN for this, although their results aren't always reliable. A forecasting model that combines a data clustering methodology, a fuzzy decision tree (FDT), and a genetic algorithm (GA) to design a decision-making system based on historical data and technical indices has been proposed in paper [6],[9].

News, blogs, Twitter mood, social networking sites, and stock articles all have the potential to influence stock market trends. Numerous studies presented stock price prediction approaches based on assessing web feelings using text mining; many publications advocated stock price predicting approaches based on analysing web feelings using text mining. Data mining and text mining approaches were used to predict stock trend changes, and a new statistically based piecewise segmentation was presented. Another study evaluated several textual representations of news stories to linear regression using SVM in order to forecast future stock price. Mood tracking technologies were used to analyse daily text content on Twitter, and it was discovered that these public mood time series might be used to anticipate changes in DJIA closing values.

Problem identified in existing works are:

The LSTM algorithm is used to anticipate the stock market, but because to the Covid 19 period, it is unable to provide a better outcome. A proposed predictive model has the ability to enhancing the effectiveness of stock market decision-making by making better stock prediction using hybrid parameters, according to one article. They employed ANN for this, although their results aren't always reliable. Furthermore, the RFE method is unaffected by term lengths other than two days, weekly, and bimonthly. A suggested future research direction would be to do more in-depth investigation into what technical indices determine irregular term durations. The assumption of constant variance is one of the most essential properties of integrated models; most financial data reflect fluctuations in volatility, and this property of the data cannot be satisfied under this premise. The symmetric joint distribution of the stationary models does not fit data with strong asymmetry (as we can observe a strong negative asymmetry for daily EUR/RON exchange rate and positive asymmetry for returns). The stationary models' symmetric linear combination does not match data with substantial asymmetry (as evidenced by the high negative asymmetry of the daily EUR/RON exchange rate and positive asymmetry of returns).

Because of the assumption of normalcy, these models are better suited to data with a low chance of abrupt bursts of very large amplitude at arbitrary time epochs. These limits of integrated models lead us to models in which we can keep the general framework but include non-Gaussian White Noise (innovations) or forego the assumptions of normality.

Proposed Methodology

i. Algorithm

Step 1: Start

Step 2: Get Dataset from yahoo finance API for 5 different stocks.

Step 3: Normalize data using Min-Max Scaler.

Step 4: Split data set into train set, validation set and test set.

Step 5: Train the LSTM model on the training data set.

Step 6: Validate the trained LSTM model using validation data set.

Step 7: Test the LSTM model using test data set.

Step 8: Save the built model

Step 9: Load the saved model in the dash file.

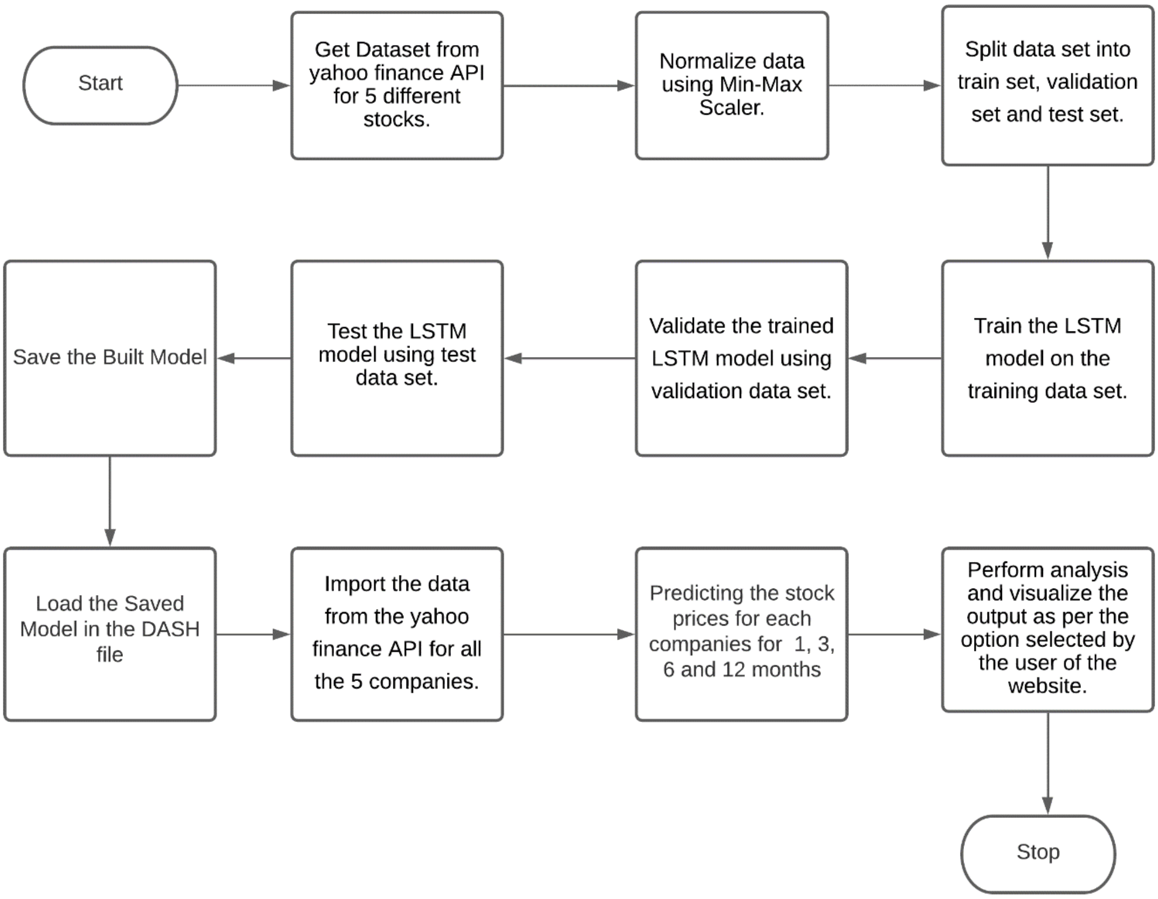
Step 10: Import the data from the yahoo finance API for all the 5 companies.

Step 11: Predict the output for each company after 1 month, 3 months, 6 months and 12 months respectively.

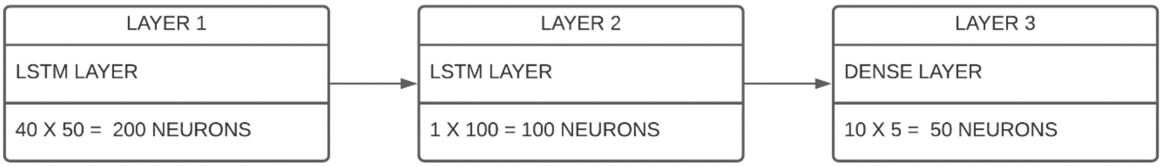
Step 12: Perform analysis and visualize the output as per the option selected by the user of the website.

Step 13: Stop

ii. System Flow Chart


Start
Get Dataset from yahoo finance API for 5 different stocks.
Normalize data using Min-Max Scaler.
Split data set into train set, validation set and test set.
Train the LSTM model on the training data set.
Validate the trained LSTM model using validation data set.
Test the LSTM model using test data set.
Save the Built Model
Load the Saved Model in the DASH file
Import the data from the yahoo finance API for all the 5 companies.
Predicting the stock prices for each companies for 1, 3, 6 and 12 months
Perform analysis and visualize the output as per the option selected by the user of the website.
Stop

iii. LSTM Architecture



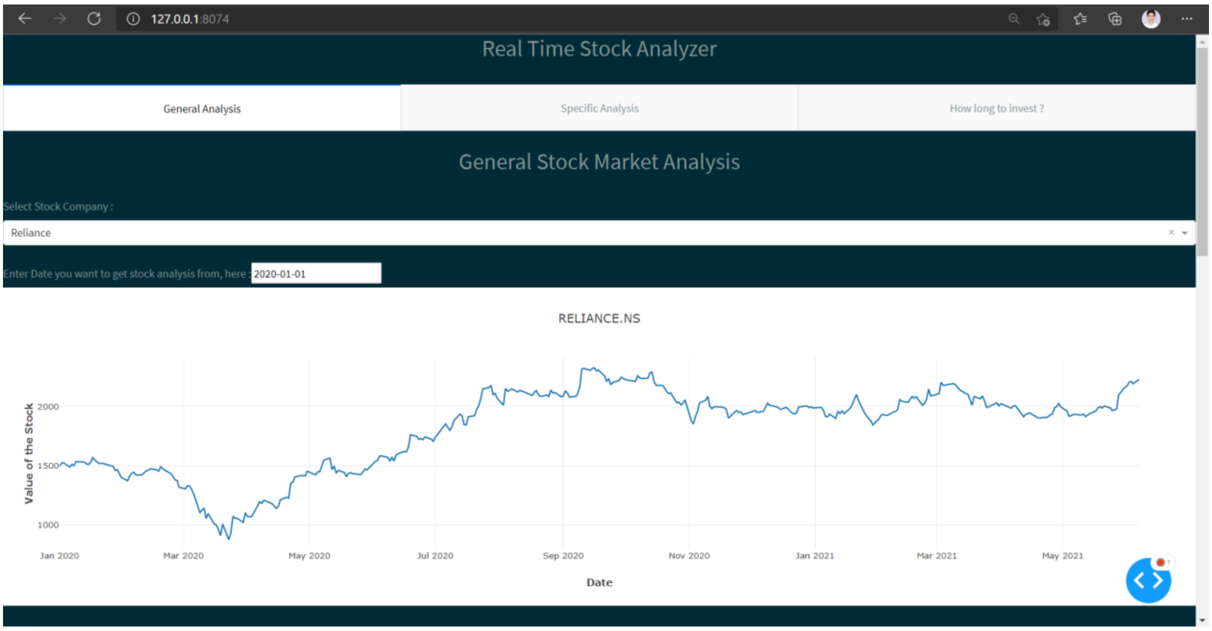
iv. Complexity Analysis

A single hidden LSTM layer is followed by a typical feedforward output layer in the original LSTM model. The Stacked LSTM is an expansion of this model that has multiple hidden LSTM layers with numerous memory blocks in each layer. Stacking LSTM hidden layers deepens the model, earning it the label of "deep learning" more correctly. The depth of deep learning is often credited with the approach's performance on a wide range of difficult prediction challenges.

For difficult sequence prediction issues, stacked LSTMs have become a reliable approach. An LSTM model with numerous LSTM layers is known as a stacked LSTM model. An LSTM layer above sends a sequence of values to the LSTM layer below, instead of a single value. Specifically, instead of one output time step for all input time steps, one output time step per input time step is used.

Results and Outputs

- DASH Home Page with real-time stock prices of the company



127.0.0.1:8074

Real Time Stock Analyzer

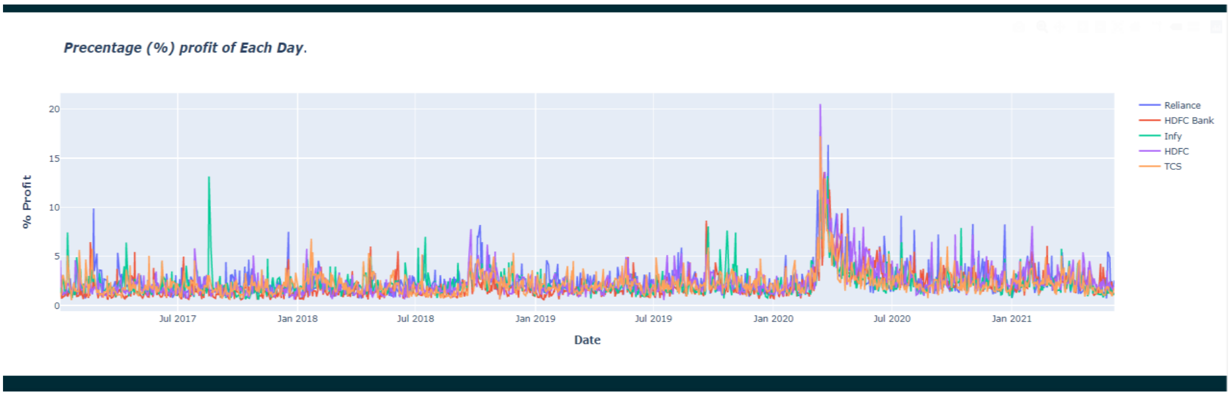
General Analysis | Specific Analysis | How long to invest ?

General Stock Market Analysis

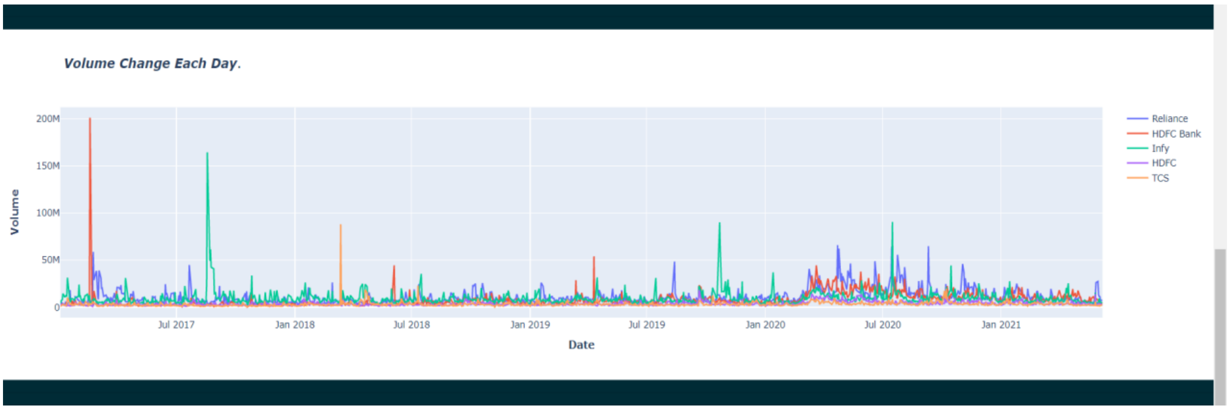
Select Stock Company : Reliance

Enter Date you want to get stock analysis from, here : 2020-01-01

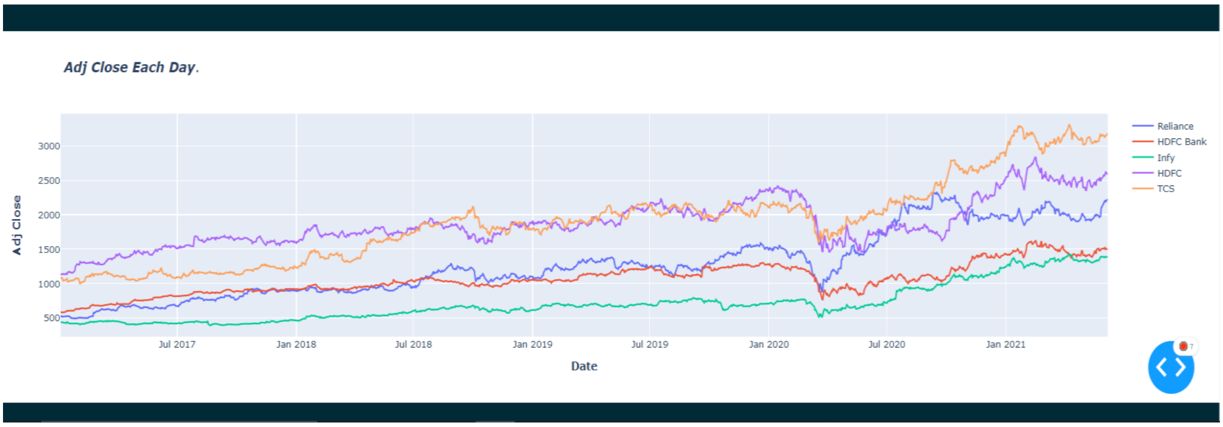
- Profit percentage of each day for all the companies.



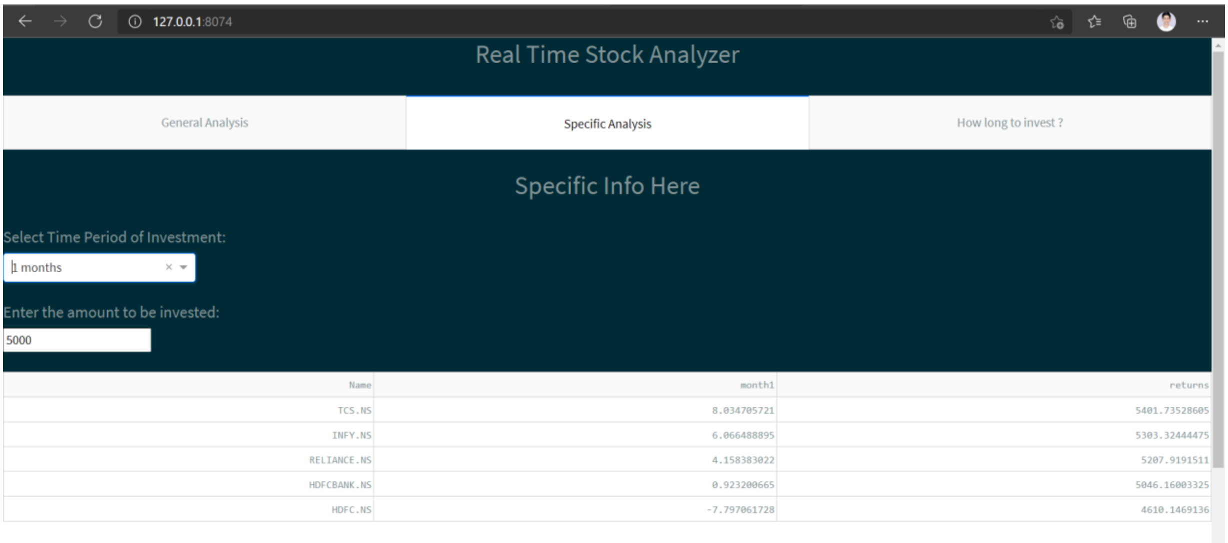
- Volume change per day for stock prices of all the companies



- Adjacent Closing Prices of stocks of the all the companies



- Stock prices of each company after the selected time period and total value of investment after the same time period.

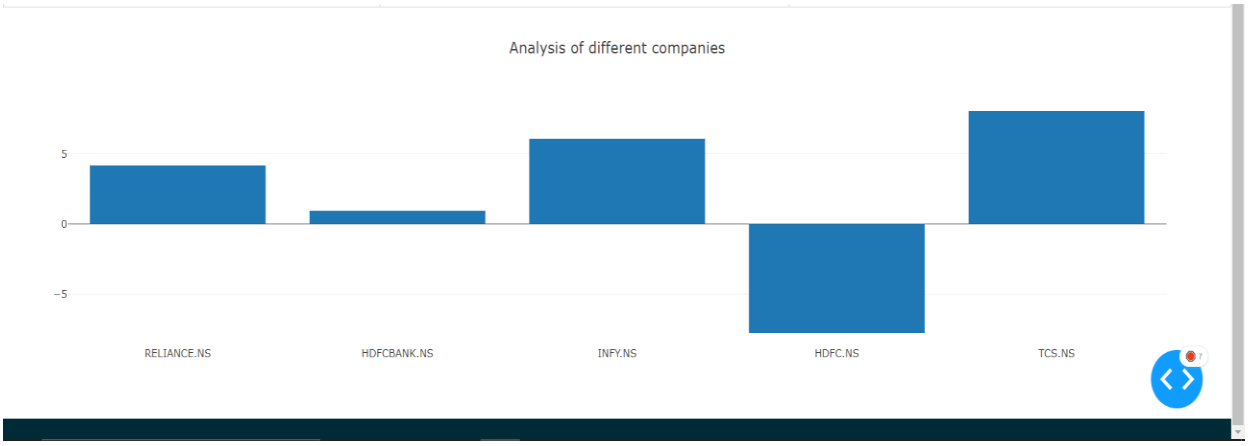


Select Time Period of Investment: 1 months

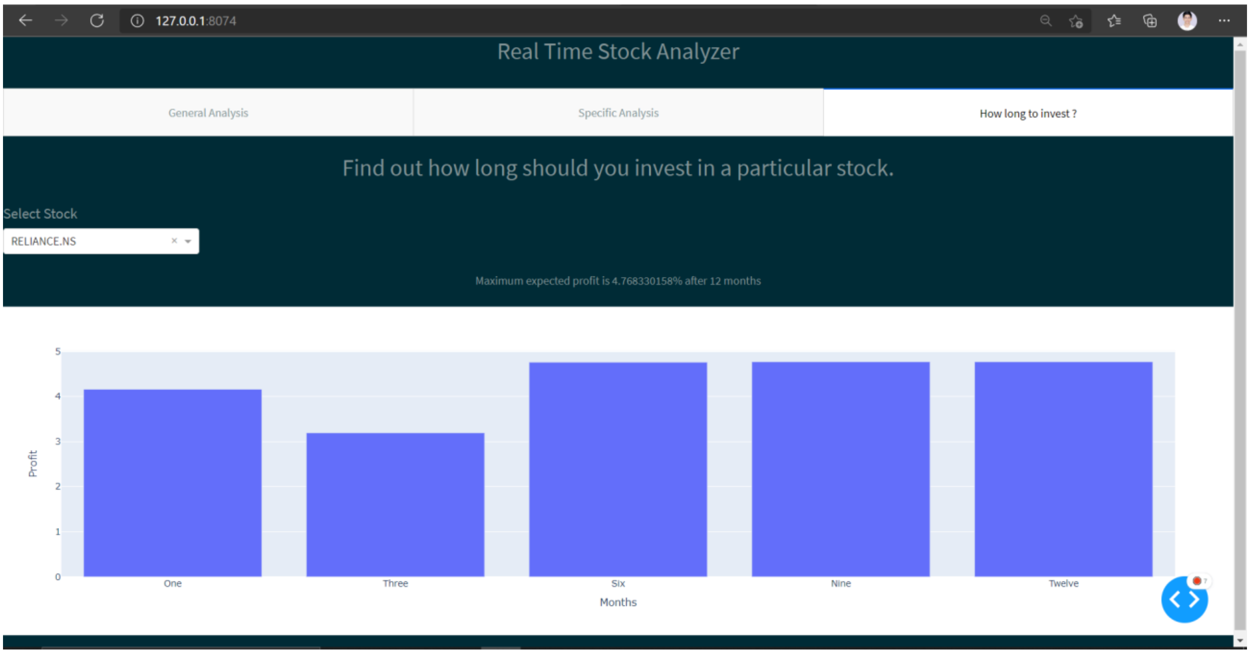
Enter the amount to be invested: 5000

Name	month1	returns
TCS.NS	8.034705721	5401.73528605
INFY.NS	6.066488895	5303.32444475
RELIANCE.NS	4.158383022	5207.9191511
HDFCBANK.NS	0.923200665	5046.16003325
HDFC.NS	-7.797061728	4610.1469136

- Visualizing the above shown net profit percentage after the investment period.



- Investment guide for stocks of each company guiding the investors in selecting the time period of investment.



Real Time Stock Analyzer

Find out how long should you invest in a particular stock.

Maximum expected profit is 4.768330158% after 12 months

Conclusion

We developed a real time stock analyser which predicts the stock prices of certain companies in real time and also visually represent them on a website. We used LSTM model for training and predicting the stock prices. The website consists of three sections: General Analysis, Specific Analysis and duration of investment. The user can choose any of the above options, and the website will display the output accordingly. In General analysis, the user can find the real-time stock prices of the companies and other features related to the companies. In specific analysis, the user can find out the stock prices after specific intervals of time. In the last section, the user can find the amount the time he/she should invest in-order to get the maximum profit. We were able to achieve an approximate accuracy of 70% and we hope to increase its future.

Future Works

The share market is one domain that has dozens of companies with complicated data, and performance is generally measured using chart diagrams that display price variation over time, as well as tree maps to indicate volume and relationships. We present a method that uses a force-directed algorithm to provide stakeholders with a novel perspective on stock connection analytics, which includes an overview and comprehensive graphs based on sheer math calculations. When compared to existing methodologies, it has the potential to assist consumers in discovering potential stock linkages. As a result, stakeholders will be able to obtain a better understanding of the potential connections between stocks in the market, and will be able to alter their investments accordingly depending on relevant stock patterns. Only a few stocks/factors were investigated in our tests, and some data was lost during the data cleansing process, which could compromise the final correctness of experiment results. New stock data factors will be addressed in future development, and more data visualisation approaches will be changed and customised to meet specific needs in stock data analysis fields. Furthermore, due to Covid 19, there is a need to update the stock predictor because there are not many excellent results in terms of forecast due to the rapid downfall during the Covid period. So, we'll aim to strengthen our prediction model so that it can produce more accurate results, and we'll also focus on predicting stocks for longer periods of time.

References

1. Yang, W. (2009). *Granule-based knowledge representation for intra and inter transaction association mining* (Doctoral dissertation, Queensland University of Technology).

2. Argiddi, R. V., & Apte, S. S. (2012). A Study of Association Rule Mining in Fragmented Item-Sets for Prediction of Transactions Outcome in Stock Trading Systems. *International Journal of Computer Engineering & Technology (IJCET)*, *3*(2), 478-486.

3. Huarng, K., & Yu, T. H. K. (2006). The application of neural networks to forecast fuzzy time series. *Physica A: Statistical mechanics and its applications*, *363*(2), 481-491.

4. Ramaraj, E. (2013). E-Rules: An Enhanced Approach to Derive Disjunctive and useful Rules from Association Rule Mining without Candidate Item Generation. *International Journal of Computer Applications*, *72*(19).

5. Agrawal, R., Imieliński, T., & Swami, A. (1993, June). Mining association rules between sets of items in large databases. In *Proceedings of the 1993 ACM SIGMOD international conference on Management of data* (pp. 207-216).

6. Chang, P. C., Fan, C. Y., & Lin, J. L. (2011). Trend discovery in financial time series data using a case based fuzzy decision tree. *Expert Systems with Applications*, *38*(5), 6070-6080.

7. Argiddi, R. V., & Apte, S. S. (2012). Future trend prediction of Indian IT stock market using association rule mining of transaction data. *International Journal of Computer Applications*, *39*(10), 30-34.

8. Anandhavalli, M., Sudhanshu, S. K., Kumar, A., & Ghose, M. K. (2009). Optimized association rule mining using genetic algorithm. *Advances in Information Mining, ISSN*, *9753265*.

9. Lai, R. K., Fan, C. Y., Huang, W. H., & Chang, P. C. (2009). Evolving and clustering fuzzy decision tree for financial time series data forecasting. *Expert Systems with Applications*, *36*(2), 3761-3773.

10. Chavan, P. S., & Patil, S. T. (2013). Parameters for stock market prediction. *International Journal of Computer Technology and Applications*, *4*(2), 337.

11. Adebiyi, A. A., Ayo, C. K., Adebiyi, M., & Otokiti, S. O. (2012). An improved stock price prediction using hybrid market indicators. *African Journal of Computing & ICT*, *5*(5), 124-135.

12. Schumaker, R. P., & Chen, H. (2009). Textual analysis of stock market prediction using breaking financial news: The AZFin text system. *ACM Transactions on Information Systems (TOIS)*, *27*(2), 1-19.

13. Bollen, J., Mao, H., & Zeng, X. (2011). Twitter mood predicts the stock market. *Journal of computational science*, *2*(1), 1-8.

Appendix

- Dash.py file

```
import pandas as pd

import pycountry

import dash

from dash.dependencies import Input, Output

import plotly.express as px

import dash_core_components as dcc

import dash_html_components as html

import plotly

import plotly.graph_objs as go

import numpy as np

import datetime

import dash_bootstrap_components as dbc

import pandas_datareader.data as web

from pandas import DataFrame

import dash_table


comp = df.Name.unique()

opt = []

for i in comp:

    opt.append(i)
```

```
st = datetime.datetime(2017,1,1)
en = datetime.datetime.now()
df_rel = web.DataReader('RELIANCE.NS','yahoo',st,en)
df_hdfc_bank = web.DataReader('HDFCBANK.NS','yahoo',st,en)
df_infy = web.DataReader('INFY.NS','yahoo',st,en)
df_hdfc = web.DataReader('HDFC.NS','yahoo',st,en)
df_tcs = web.DataReader('TCS.NS','yahoo',st,en)

fig = go.Figure()

fig.add_trace(go.Scatter(
    x=df_rel.index,
    y = ((df_rel.High - df_rel.Low)/(df_rel.High))*100,
    name = 'Reliance',
))
fig.add_trace(go.Scatter(
    x = df_hdfc_bank.index,
    y = ((df_hdfc_bank.High - df_hdfc_bank.Low)/(df_hdfc_bank.High))*100,
    name='HDFC Bank',
))
fig.add_trace(go.Scatter(
    x=df_infy.index,
    y = ((df_infy.High - df_infy.Low)/(df_infy.High))*100,
    name='Infy',
))
fig.add_trace(go.Scatter(
    x=df_hdfc.index,
    y = ((df_hdfc.High - df_hdfc.Low)/(df_hdfc.High))*100,
    name='HDFC',
))
fig.add_trace(go.Scatter(
    x=df_infy.index,
    y = ((df_tcs.High - df_tcs.Low)/(df_tcs.High))*100,
    name='TCS',
))

fig.update_layout(
    title_text="<b><i>Precentage (%) profit of Each Day</i></b>."
)
fig.update_xaxes(title_text="<b>Date</b>")
fig.update_yaxes(title_text="<b>% Profit</b>")
```

```
fig1 = go.Figure()


fig1.add_trace(go.Scatter(
    x=df_rel.index,
    y = df_rel.Volume,
    name = 'Reliance',
))
fig1.add_trace(go.Scatter(
    x = df_hdfc_bank.index,
    y = df_hdfc_bank.Volume,
    name='HDFC Bank',
))
fig1.add_trace(go.Scatter(
    x=df_infy.index,
    y = df_infy.Volume,
    name='Infy',
))
fig1.add_trace(go.Scatter(
    x=df_hdfc.index,
    y = df_hdfc.Volume,
    name='HDFC',
))
fig1.add_trace(go.Scatter(
    x=df_tcs.index,
    y = df_tcs.Volume,
    name='TCS',
))
fig1.update_layout(
    title_text="<b><i>Volume Change Each Day</i></b>."
)
fig1.update_xaxes(title_text="<b>Date</b>")
fig1.update_yaxes(title_text="<b>Volume</b>")

fig2 = go.Figure()

fig2.add_trace(go.Scatter(
    x=df_rel.index,
    y = df_rel["Adj Close"],
    name = 'Reliance',
))
fig2.add_trace(go.Scatter(
    x = df_hdfc_bank.index,
    y = df_hdfc_bank["Adj Close"],
    name='HDFC Bank',
))
fig2.add_trace(go.Scatter(
    x=df_infy.index,
    y = df_infy["Adj Close"],
    name='Infy',
))
fig2.add_trace(go.Scatter(
    x=df_hdfc.index,
    y = df_hdfc["Adj Close"],
    name='HDFC',
))
fig2.add_trace(go.Scatter(
    x=df_tcs.index,
    y = df_tcs["Adj Close"],
    name='TCS',
))
fig2.update_layout(
    title_text="<b><i>Adj Close Each Day</i></b>."
)
fig2.update_xaxes(title_text="<b>Date</b>")
fig2.update_yaxes(title_text="<b>Adj Close</b>")
```

```
#app = dash.Dash(__name__,external_stylesheets=[dbc.themes.CYBORG])

tab1 = html.Div([

    html.Br(),
    html.Div([
        html.H2("General Stock Market Analysis", style={"textAlign": "center"}),
    ], className="cls"),

    html.Br(),

    html.Label("Select Stock Company : "),
    html.Br(),

    dcc.Dropdown(
        id='dropdown',
        options=[
            {'label': 'Reliance', 'value': 'RELIANCE.NS'},
            {'label': 'HDFC Bank', 'value': 'HDFCBANK.NS'},
            {'label': 'INFY', 'value': 'INFY.NS'},
            {'label': 'HDFC', 'value': 'HDFC.NS'},
            {'label': 'TCS', 'value': 'TCS.NS'}
        ],
        value='RELIANCE.NS'
    ),
    html.Br(),
    html.Label("Enter Date you want to get stock analysis from, here :   "),
    dcc.Input(id='input',value='2017-01-01',type='text'),


    html.Div(id='output-graph'),
    html.Hr(className="abc"),


   html.Div(


        dcc.Graph(
        id='example-graph-2',
        figure=fig
        )
   ),
    html.Hr(className="abc"),


    html.Div(
           dcc.Graph(

           id='example-graph-3',

           figure=fig1

           )

       ),

        html.Hr(className="abc"),

        html.Div(


           dcc.Graph(

           id='example-graph-4',

           figure=fig2

           )

       )


   ])






   tab2 = html.Div([


        html.Br(),
   html.H2("Specific Info Here", style={'text-align': 'center'}),
   html.Br(),
   html.H5("Select Time Period of Investment: "),
   dcc.Dropdown(id = 'menu_select11',
   options =[
       {'label': '1 months', 'value': '1 month'},
       {'label': '3 months', 'value': '3 months'},
       {'label': '6 months', 'value': '6 months'},
       {'label': '9 months', 'value': '9 months'},
       {'label': '12 months', 'value': '12 months'},


   ],
       multi=False,
       value="",
       style={'width': "40%"}
   ),
   html.Br(),
   html.H5("Enter the amount to be invested:"),
   dcc.Input(id='input11', value='5000', type='number'),


   html.Div(id='output_container',children=[]),
   html.Br(),
    html.Div(id='output-graph1'),



    #dcc.Graph(id='my_bee_map',figure={}),
    ])



tab3= html.Div(children=[


        html.Br(),
        html.H2("Find out how long should you invest in a particular stock.", style={'text-align': 'center'}),


        html.Br(),
        html.H5("Select Stock"),
        dcc.Dropdown(id = 'slct_st2',
            options = [{'label':name, 'value':name} for name in opt],
            multi=False,
            value="RELIANCE.NS",
            style={'width': "40%"}
        ),
        html.Br(),
        html.Div(id='tab2_container',children=[],style={'text-align': 'center'}),
        html.Br(),

         html.Div(
               id='cx1'
               )
        ])



    app = dash.Dash(__name__,external_stylesheets=[dbc.themes.SOLAR])
    theme =  {
      'dark': True,
      'detail': '#007439',
      'primary': '#00EA64',
      'secondary': '#6E6E6E',
    }

    app.layout = html.Div([
      html.H2('Real Time Stock Analyzer',style={'text-align': 'center'}),
      html.Br(),
      dcc.Tabs(id="tabs-example", value='tab-1-example', children=[
        dcc.Tab(id="tab-1", label='General Analysis', value='tab-1-example'),

        dcc.Tab(id="tab-2", label='Specific Analysis', value='tab-2-example'),

        dcc.Tab(id="tab-3", label='How long to invest ?', value='tab-3-example'),

    ]),

    html.Div(id='tabs-content-example',

             children = tab1)

])



@app.callback(Output('tabs-content-example', 'children'),

             [Input('tabs-example', 'value')])

def render_content(tab):

    if tab == 'tab-1-example':

        return tab1

    elif tab == 'tab-2-example':

        return tab2

    elif tab == 'tab-3-example':

        return tab3


@app.callback(

    Output(component_id='output-graph',component_property='children'),

    [Input('dropdown', 'value')],
    [Input(component_id='input',component_property='value')])

def update_output1(value,date):

    l = date.split('-')
    print(l)
    year = int(l[0])
    month = int(l[1])
    day = int(l[2])
    start = datetime.datetime(year,month,day)
    end = datetime.datetime.now()
    df = web.DataReader(value,'yahoo',start,end)

    return dcc.Graph(
            id='example-graph',
            figure={
                'data': [
                    {'x': df.index,'y':df.Close,'type':'line','name':value},
                ],
                'layout':{
                    'title':value,
                    'xaxis':{
```

```
                                'title':'<b>Date</b>'

                                },

                                'yaxis':{

                                    'title':'<b>Value of the Stock</b>'

                                }

                        }

                }

        )


@app.callback(

    Output(component_id='output-graph1', component_property='children'),

    [Input(component_id='menu_select11',component_property='value'),

    Input(component_id='input11', component_property='value')]

)

def update_output2(option_slctd,input1):



    df=pd.read_csv("file_name1.csv")

    dfff=df.copy()

    i1=float(input1)



    if(option_slctd=='1 month'):
```

```
    dfff =df[['Name','month1']].copy()

    dfff=dfff.sort_values(by=['month1'], ascending=False)

    dfff['returns'] =  df['month1'].map(lambda a: i1+(a*int(i1)/100))

    print(dfff)

    return [dash_table.DataTable(

    id='table',

    columns=[{"name": i, "id": i} for i in dfff.columns],

    data=dfff.to_dict('records'),

    ),

    dcc.Graph(

    id='example-graph',

    figure={

        'data': [

            {'x': df.Name, 'y': df.month1, 'type': 'bar', 'name': 'input_data'},

        ],

        'layout': {

            'title': "Analysis of different companies"

        }

    }

)


]
```

```
if(option_slctd=='3 months'):
    dfff =df[['Name','months3']].copy()
    dfff=dfff.sort_values(by=['months3'], ascending=False)
    dfff['returns'] =  df['months3'].map(lambda a: i1+(a*int(i1)/100))
    print(dfff)
    return [dash_table.DataTable(
    id='table',
    columns=[{"name": i, "id": i} for i in dfff.columns],
    data=dfff.to_dict('records'),
    ),
    dcc.Graph(
    id='example-graph',
    figure={
        'data': [
            {'x': df.Name, 'y': df.months3, 'type': 'bar', 'name': 'input_data'},
        ],
        'layout': {
            'title': "Analysis of different companies"
        }
    }
)
```

```
]

if(option_slctd=='6 months'):
    dfff =df[['Name','months6']].copy()

    dfff=dfff.sort_values(by=['months6'],ascending=False)
    dfff['returns'] =  df['months6'].map(lambda a: i1+(a*int(i1)/100))
    print(dfff)
    return [dash_table.DataTable(
    id='table',
    columns=[{"name": i, "id": i} for i in dfff.columns],
    data=dfff.to_dict('records'),
    ),
    dcc.Graph(
    id='example-graph',
    figure={
        'data': [
            {'x': df.Name, 'y': df.months6, 'type': 'bar', 'name': 'input_data'},
        ],
        'layout': {
            'title': "Analysis of different companies"
        }
    }
)

]

if(option_slctd=='9 months'):
    dfff =df[['Name','months9']].copy()

    dfff=dfff.sort_values(by=['months9'], ascending=False)
    dfff['returns'] =  df['months9'].map(lambda a: i1+(a*(i1)/100))
    print(dfff)
    return [dash_table.DataTable(
    id='table',
    columns=[{"name": i, "id": i} for i in dfff.columns],
    data=dfff.to_dict('records'),
    ),
    dcc.Graph(
    id='example-graph',
    figure={
        'data': [
            {'x': df.Name, 'y': df.months9, 'type': 'bar', 'name': 'input_data'},
        ],
        'layout': {
            'title': "Analysis of different companies"
        }
    }
)

]

if(option_slctd=='12 months'):
    dfff =df[['Name','months12']].copy()


    dfff=dfff.sort_values(by=['months12'], ascending=False)
    dfff['returns'] =  df['months12'].map(lambda a: i1+(a*int(i1)/100))
    print(dfff)
    return [dash_table.DataTable(
    id='table',
    columns=[{"name": i, "id": i} for i in dfff.columns],
    data=dfff.to_dict('records'),
    ),
    dcc.Graph(
    id='example-graph',
        figure={
            'data': [
                {'x': df.Name, 'y': df.months12, 'type': 'bar', 'name': 'input_data'},
            ],
            'layout': {
                'title': "Analysis of different companies"
            }
        }
    )

    ]

@app.callback(
    [Output(component_id = 'tab2_container',component_property='children'),
    Output(component_id='cx1',component_property = 'children')],
    [Input(component_id='slct_st2',component_property='value')]
)

def update_output3(value):

    df=pd.read_csv("file_name1.csv")
    df1 = df[df["Name"]==value]

    ind = list(np.where(df["Name"]==value))

    ind = ind[0][0]

    x1 = ['One','Three','Six','Nine','Twelve']

    y1                                                                    =
[df1['month1'][ind],df1['months3'][ind],df1['months6'][ind],df1['months9'][ind],
df1['months12'][ind]]

    #print(df1['month1'])



    x1 = DataFrame(x1,columns=['Months'])

    y1 = DataFrame(y1,columns=['Profit'])


    df_new = pd.concat([x1,y1],axis=1)




    df_new = pd.concat([x1,y1],axis=1)


    temp3 = px.bar(df_new, x='Months', y='Profit')


    fig3 = html.Div(

        dcc.Graph(
            id='bar chart',
            figure=temp3,
        )
)

maxi = max(y1.values)
cnt=1
for i in y1:
    if maxi[0]==i:
        break
    else:
        if cnt==1:
            cnt=3
        else:
            cnt = cnt+3
pos = -1
month = 0
ind = 0
for i in y1.values:
    print("working")
    if i == maxi:
        pos = ind
```

```
            break
        ind = ind+1
    if(pos==0):
        month = 1
    elif(pos==1):
        month = 3
    elif(pos==2):
        month = 6
    else:
        month = 12
    print(month)
    container1 = "Maximum expected profit is {}% after {} months".format(maxi[0],month)
    print(container1)
   # container1 = "Suggested invest duration is {} & esmtimated profit is {} %".format(10,value)
    return container1, fig3



if __name__ == '__main__':
    app.run_server(debug=True,port=8074)
```