APPLYING PREDICTIVE ANALYSIS ON STOCK MARKETS

GONDI RAJEEV (19Z213)
KOUSHIK BALAJI P (19Z223)
LOHITH SOWMIYAN P S (19Z224)
M.MANOJKUMAR (19Z226)
RAKESH. M (19Z235)
SAIRAM VAIDYA M (19Z238)

Dissertation submitted in partial fulfillment of the requirements for the degree of

BACHELOR OF ENGINEERING
BRANCH : COMPUTER SCIENCE AND ENGINERING
PSG College of Technology

May 2022

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

PSG COLLEGE OF TECHNOLOGY

(Autonomous Institution)

COIMBATORE – 641 004

PSG COLLEGE OF TECHNOLOGY

(Autonomous Institution)

Coimbatore – 641 004

APPLYING PREDICTIVE ANALYSIS ON STOCK MARKETS

Work done by

GONDI RAJEEV (19Z213)
KOUSHIK BALAJI P(19Z223)
LOHITH SOWMIYAN P S(19Z224)
M.MANOJKUMAR (19Z226)
RAKESH. M (19Z235)
SAIRAM VAIDYA M. (19Z238)

Dissertation submitted in partial fulfillment of the requirements for the degree of

BACHELOR OF ENGINEERING
BRANCH : COMPUTER SCIENCE AND ENGINERING

MAY 9 , 2022

…….
Mrs. Swathi . J
Faculty guide

…….
Dr. Sudha Sadasivam
Head of the Department

Certified that the candidate was examined in the viva-voce examination held on ……

…….
(Internal Examiner)

…….
(External Examiner)

CONTENTS

CHAPTER **Page No.**

ACKNOWLEDGEMENT

The Innovation Practices Laboratory We had, gave us an irreplaceable platform to enhance our skills and knowledge. We are really grateful to our Principal Dr.K.Prakasan for offering us this opportunity which has made us gain more knowledge in this field. We feel glad to extend our gratitude to our Head-In-Charge Dr.Sudha Sadasivam who has supported us till the end. We sincerely thank our Tutor and our Project Mentor Ms.J.Swathi who has shown her support in all the ways for completing our internship with ease and spent her valuable time to aid us for coming out with good project.

We would like to place our sincere thanks to PredictRAM (Params Data Provider Pvt Ltd.) and Industry Mentor Mr.Subir Singh for offering us this internship with their organization and They have showered us with their utmost support and made us aware of unique things which we had came across without any knowledge and gave us the vision to see where our future industries are heading towards.

Finally We would like to thank our teammates whose involvement and curiosity has always been a driving force to learn new and unique things.

SYNOPSIS

Stocks represent the performance of a company with respect to parameters like supply and demand exchange rates of the stocks and overall performance of the company. The stock prices give us the standing of a company in their respective sectors. These factors are highly unpredictable which in turn makes the prices of the stocks highly volatile.

This basic idea of the stock market is, investors try to buy atomic shares of a company to claim their stake which is proportional to the amount of shares they have. As the prices of the stocks may increase or decrease at a particular after a certain time, investors hope to make profits by selling the stocks once their price rises. While it may sound simple, making profits through stocks is a highly difficult task.

Veteran investors might have great experiences in this process and they might know the caveats involved in stock market, but new investors always find it difficult to cope up with market. As a result, they have high probabilities of making losses. The main aim of the project is to aid these new investors to leverage their chances of making profits by providing them with various aspects of both the market and the particular stock they are interested in.

Conventionally people use popular models like auto regression and moving averages which makes use of the past data to give predictions or insights about the future prices but as we recently witnessed the onslaught of the pandemic driving every industry to an unprecedented crisis which in turn had an adverse impact on their stock prices. In order to overcome these situations, it would be better to analyze various parameters like customer sentiments which could accurately depict the current situation of the market, how changes made by companies are responded by people etc.

Moreover, analyzing the overall economy and industry might go hand in hand in determining the future potential of a stock and even trends in their revenues also might be influential. The main idea is to use these various analyses to help use decide whether it would be safe to invest in a particular stock.

CHAPTER 1
INTRODUCTION

1.1 MOTIVATION

Besides the life skills that a person should possess, recently there has been an increasing interest in having two skills that are in fact considered to be essential in the current world – being legally aware, and financially sound. Finance is something that is involved in our everyday lives, including the grocery purchases, buying commodities online (generally during sales or with high discounts!), applying for a loan, and so many more!

One of the major aspects of Finance is to increase the value of one's assets – this could be seen as purchasing more gold, investing in Mutual Funds, etc. and the essential idea remains to increase the "money" value that each of these assets possess. The classical model observed in Indian households includes the following theories – investing in more gold, buying more real estate, and that GDP growth drives the stock market: so, one can time the economic cycle and thereby time the stock market.

However, these theories are considered as myths by many economists because they possess several demerits in today's world, and generally the best approach to increase one's corpus is via stock markets.

1.2 ABSTRACT

Investing in gold might seem attractive given the historical and traditional value gold possesses, but if one were to disregard the sentimental attachment towards gold and only see it from a "money" perspective it has a primary disadvantage – over the last decades, gold has performed poorly when compared to the BSE Sensex index by over 3% on average, and is also more volatile which suggests that on a volatility-adjusted basis, there is not a compelling reason to own gold. From the below table, we see how gold has become more volatile in recent years when compared to the BSE Sensex index while offering lower returns.

	Cumulative periods			Decadal periods		
Asset	2010-2020 (10-years) (1)	2000-2020 (20-years) (2)	1990-2020 (30-years) (3)	2010-2020 (4)	2000-2010 (5)	1990-2000 (6)
Annualized returns						
– Gold	9.2%	12.7%	9.3%	9.2%	16.3%	2.8%
– BSE Sensex	10.4%	15.0%	14.8%	10.4%	19.9%	14.4%
Standard deviation						
– Gold	15.8%	15.2%	14.9%	15.8%	14.1%	11.9%
– BSE Sensex	14.1%	27.9%	36.3%	14.1%	35.7%	48.5%

Fig 1.1 Comparative returns and volatility of gold and Indian equities

Real estate is widely regarded to be something the “big guns” do, since it requires a lot of resources and initial investment. In fact, India has one of the highest house prices against GDP per capita as indicated below, making real estate a poor choice. So, there are many other reasons to why real estate is not suggested for someone’s portfolio – needs a lot of resources, procedures, legal papers, very high financial investment, and an inability to liquidate the assets when required. It is also very volatile, and very hard to maintain, hence making it a poor choice for a person’s portfolio.

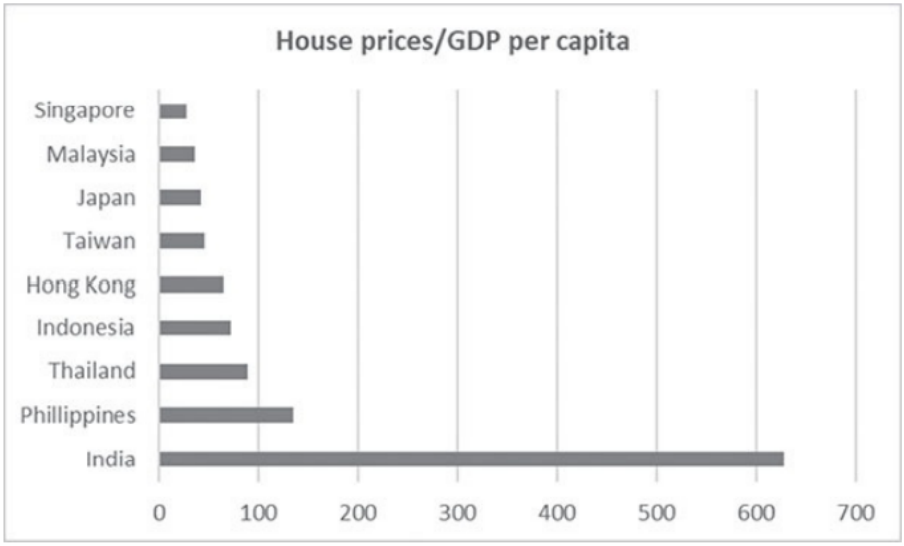


Fig 1.2 Residential house prices/GDP per capita for multiple countries

The final myth is the theory that you can possibly time the market if you have a good grasp on the GDP – but this is in fact a terrible approach. In the past, there might have been some relation between them but over the last 5 years there has been a growing gap between the GDP and the Stock Index – in fact their growth has been in the opposite direction.

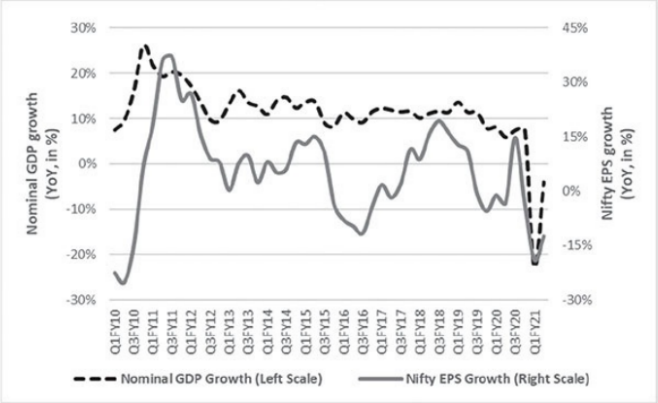


Fig 1.3 Nifty50 EPS growth vs the nominal GDP growth

Let's consider a case study to further enunciate the differences between the different investment theories described above,

Assume that a person earns Rs.50,000/- per month and spends Rs.30,000/- towards cost of living. The balance of Rs.20,000/- becomes the monthly surplus. For this case study, we do not consider personal income tax. We make a few more assumptions to make the case more intricate:

1. The employer gives the person a 10% salary hike every year
2. The cost of living is likely to go up by 8% year on year
3. The person is 30 years old and plans to retire at the age of 50. So, this leaves him with 20 more years to earn
4. The person doesn't intend to work after retirement
5. The expenses are fixed, and we don't foresee and other expense
6. The balance case of Rs.20,000/- per month is retained in the form of hard cash.

Going by these assumptions, in 20 years the cash balance will look like:

Table 1.1 Total cash balance in twenty years

Years	**Yearly Income**	**Yearly Expense**	**Cash Retained**
1	6,00,000	3,60,000	2,40,000
2	6,60,000	3,88,800	2,71,200
…	…	…	…
19	33,35,950	14,38,567	18,97,383
20	36,69,545	15,53,652	21,15,893
		Total Income	1,78,90,693

From the above table we can observe that,

1. After 20 years of work the person would have accumulated Rs.1.7 Crs.
2. Since the expenses are fixed, the person's lifestyle also wouldn't have changed – i.e., he wouldn't have bought a car, house, vacations, etc.

3. After the person retires and if the expenses grow with the same rate of 8%, Rs.1.7 Crs is good enough for about 8 years. After the 8th year, the person is in a very tight spot with no savings left to back himself up.

Consolidating the models mentioned above, a person generally has few choices available to invest in and hence increase their corpus. The popular asset classes are – Fixed income instruments (fixed deposits offered by banks, bonds issued by the Government/Corporates), Real estate, Bullion (precious metals like gold, silver, etc.) and Equity. Without going into the detail, if the person mentioned in the above case were to invest in each of the above asset classes, it would look like:

1. By investing in fixed income (average rate is 9% per annum), the corpus would have grown by Rs.3.3 Crs
2. By investing in bullion (average rate is 8% per annum), the corpus would have grown by Rs.3.09 Crs
3. By investing in equities (average rate is 15% per annum), the corpus would have grown by Rs 5.4 Crs
4. It is very difficult to predict the returns real estate would provide given the intricacies and inherent complexity of it, so the corpus's growth cannot be estimated.

Hence, we notice that investments should have their primary focus towards equity to maximize one's returns; they can be diversified to include all the asset classes, but the principal focus should remain on the equities, i.e., the stock market.

1.3 PROBLEM STATEMENT

However, it has been proved time and again that stock markets can be very difficult to predict by humans due to a huge array of factors and the time frames. And during unlikely events, like the COVID-19 pandemic, we have seen the tables turn drastically in the stock market.

The problem statement deals with apply predictive analysis on stock markets on a hierarchical basis – the Economy, then the industry and finally Companies using various techniques such as data analysis, data mining, machine learning, deep learning, etc. Then, there is a need to capture the market mood and combine it with the analysis performed. Finally, the results need to be aggregated and displayed to the client to offer various insights using which the client can improve his portfolio.

Our dashboard takes factors like GDP, industry specific incomes and multiple other relevant factors to predict the trends of the market. We also apply our tests on differing time constraints, industry types and ranges to obtain reliable results.

1.4 PROPOSED SOLUTION

The proposed solution uses the same ideas mentioned in the problem statement, by performing hierarchical analysis and combining it with sentimental analysis performed dynamically using Google News API. Then, the results obtained are aggregated and

displayed through a dashboard where only the essential metrics are shown, and the client has a choice to select any other metric required from the options provided. The system is built to work dynamically, so that it captures the market mood at every instant rather than a default mood stored in some database. The timeline of the proposed solution is given below, which consists of an input phase being the problem statement, output phase being the dashboard creating and six intermediate phases to convert the input to the required final product.

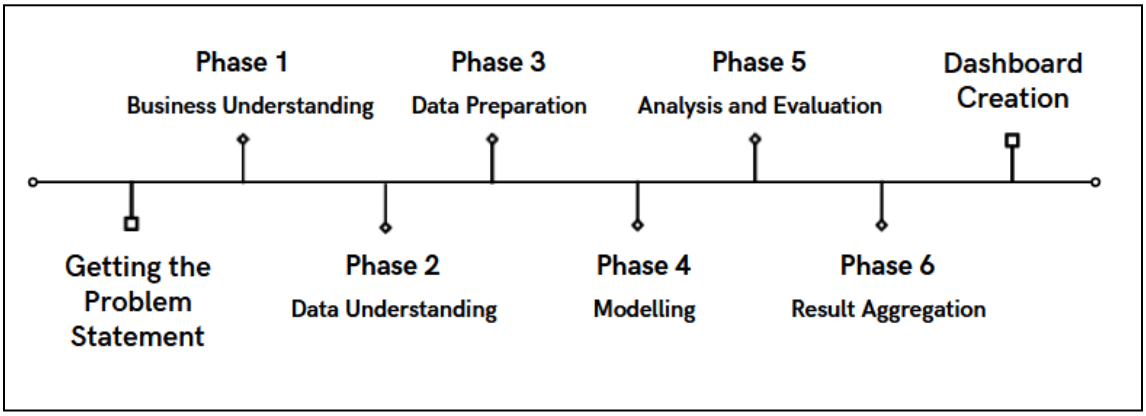


Fig 1.4 Timeline of the proposed solution

To achieve this, the phases were further divided, and a Gantt chart was used to plan the schedule for product development. The Gantt chart is given below, and it consists of 8 modules spread across 9 steps over a 18 week period from initial stage to product completion. This was the primary plan followed throughout the course of the project, to develop the final product.

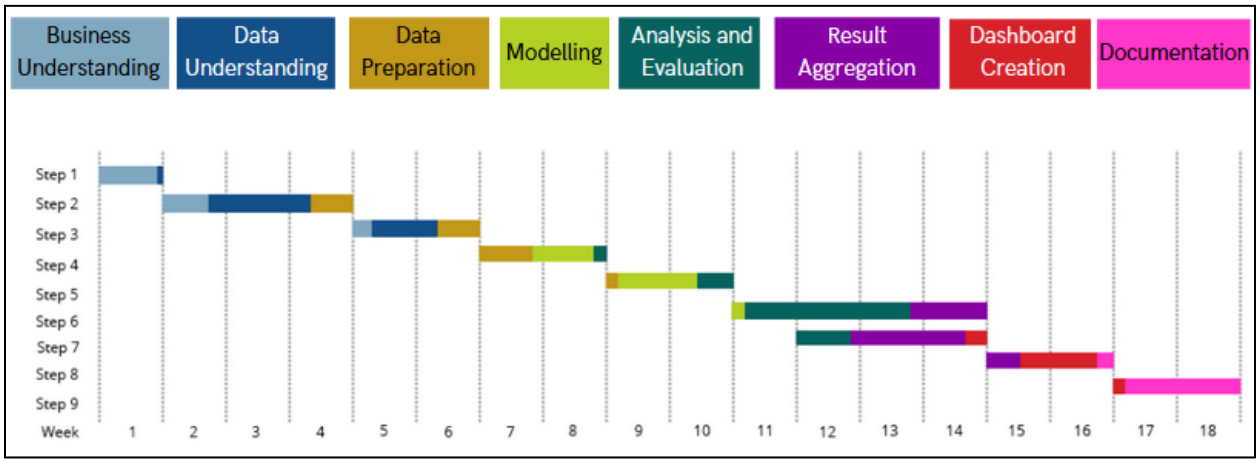


Fig 1.5 Gannt chart showing planned steps for proposed solution

CHAPTER 2

LITERATURE SURVEY

S.No.	Purpose	References	Advantages	Disadvantages
1.	Prediction and analysis of stock market data using various algorithms (linear/non-linear)	Financial Markets Prediction with Deep Learning - Jia Wang, et al.	Probability of high returns for short term, ownership stake in the company	Volatility, unsystematic risk is unique to a specific industry or a company and can be diversified
2.	Prediction and analysis of stock market data using LSTM, Linear regression, random forest and k-nearest neighbours neural network	Real Time Stock Market Analysis -Naman Adlakha; Ridhima; Avita Katal	Effective forecast tools assist traders indirectly by supplying supportive knowledge such as price position in the future.	Effective forecast tools assist traders indirectly by supplying supportive knowledge such as price position in the future.
3.	Data Science relies heavily on forecasting future outcomes and data modeling using Time-Series modelling	The prediction system for data analysis of stock market - Ching-Te Wang, et al.	Stock market plays a pivotal role in the financial aspect of the nation's growth.	It is highly volatile and complex in nature. It is affected by significant political issues and uncertainty in the future of a company.
4.	It provides consistency in measures and measurement procedures across department and business units.	Pauwels, et al. (2009). Dashboards & Marketing: Why, what, How and Which Research is Needed? Journal of Service Research. 12. 175-189.	Dashboards can communicate what an organization values and can be used to disclose crucial marketing information to investors.	The academic research is needed fully to exploit their potential in viewing the analysis for the users.

CHAPTER 3

SYSTEM REQUIREMENTS

System requirements defines the software and hardware specifications that has to be met for the optimal functioning of the system.

3.0 HARDWARE REQUIREMENTS:

- A System with 8 GB Ram and 64 GB storage

3.1 SOFTWARE REQUIREMENTS:

3.2.1 IDE / Text Editor

Visual Studio Code

Visual Studio Code (a.k.a. VS Code) is a free, open-source text editor developed by Microsoft. VS Code is available for various operating systems like Windows, Linux, and macOS. Though the editor is quite lightweight, it provides some powerful features that have made VS Code one of the most popularly used development environment tools in current times.

3.2.2 Libraries and Frameworks

3.2.2.1 Tableau and Power BI

Power BI is a business analytics service provided by Microsoft that can analyze and visualize data, extract insights, and share it across various departments within your organization. While Tableau is a powerful Business Intelligence tool that manages the data flow and turns data into actionable information.

3.2.2.2 Tensorflow

TensorFlow is a free and open-source software library for machine learning and artificial intelligence. It can be used across a range of tasks but has a particular focus on training and inference of deep neural networks.

3.2.2.3 Streamlit

Streamlit is an open-source app framework in Python language. It helps us create web apps for data science and machine learning in a short time. It is compatible with major Python libraries such as scikit-learn, Keras, PyTorch, NumPy, pandas, Matplotlib etc.

3.2.2.4 Beautiful Soup

Beautiful Soup is a Python package for parsing HTML and XML documents. It creates a parse tree for parsed pages that can be used to extract data from HTML, which is useful for web scraping.

3.2.2.5 Matplotlib

Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, wxPython, Qt, or GTK

3.2.2APIs

3.2.2.1 Yahoo Finance API

It provides data of stocks, financial news, press releases and financial reports of companies. The free tier of Yahoo Finance API gives 500 calls/month.

3.2.2.2 Google News API

The API allows you to integrate Google News search results into your application or web pages. You can use it to display topics, headlines, trending stories, URLs, and other news items from Google searches.

3.2.2.3 NLTK

NLTK, Natural Language Toolkit, is a toolkit that was built for working with natural language processing in Python. NLTK provides us with several important text processing libraries along with a large number of test datasets. It also enables the user to perform a variety of tasks in the NLP process like tokenization, lemmatization, visualization of parse trees, etc.

CHAPTER 4

SYSTEM DESIGN

4.1 IIP ANALYSIS

The index of industrial production is a measure of India's short-term increase in industrial output. It represents the overall level of industrial activity. This indicator is based on data from three major industries: manufacturing, mining, and electricity. It is also quantified separately for each of the six use-base sectors: primary products, capital goods, intermediate goods, infrastructure/construction goods, consumer durables, and consumer non-durables. Initially, economic analysis on IIP dataset was performed to get a good understanding on the various terms that are related to the stock market. The IIP shows the growth rate in different industry groups of the economy in a stipulated period, and it could potentially make a large impact on the stock market such as a weak IIP leading to a sudden fall in stock prices. The IIP values also give good investment opportunities, since a continuous fall in IIP may lead to many strong stocks being undervalued. As a result, an individual gets a good opportunity to invest in strong companies at a lower price. For the project, we used the two different quantifications of IIP mentioned above and analyzed them as given below.

4.1.1 NIC ANALYSIS

For this, the considered dataset involved various sectors and their IIP values over a period, after which basic pre-processing and various visualization techniques were applied. The visualization techniques included line plots, graph decomposition to identify the trend, seasonality, cyclic patters. These plots were also used to identify stationarity among the sectors. However, to go for a more quantifiable approach than a usual one given the inaccuracy and complexity involved in identifying patters visually, for handling stationarity the Augmented Dickey Fuller (ADF) test was used.

From this, it was understood that most of the sectors were not stationary, so the dataset needed some changes to reach stationarity. For this, there are two general approaches – differencing or transformation. Given the volatility in the dataset, differencing techniques were preferred over transformation, and hence by taking different orders of differentiation the dataset was able to attain stationarity.

Then, the correlation matrix was plotted using Pearson correlation to identify the relationship between the sectors. Next, an ARIMA model was created to understand the growth/fall of the IIP, and since time is a major factor in stock markets, time series analysis was performed using ARIMA. ARIMA is basically an integration of Auto Regression and Moving Average Models.

The first phase is to display the ACF/PACF graphs, which determine the parameters for Auto Regression and Moving Average. Next, based on the stationarity of

the data the Integrating factor (I in ARIMA) is selected. Finally, by adjusting the other parameters of the ARIMA model, the final version is built to make predictions.

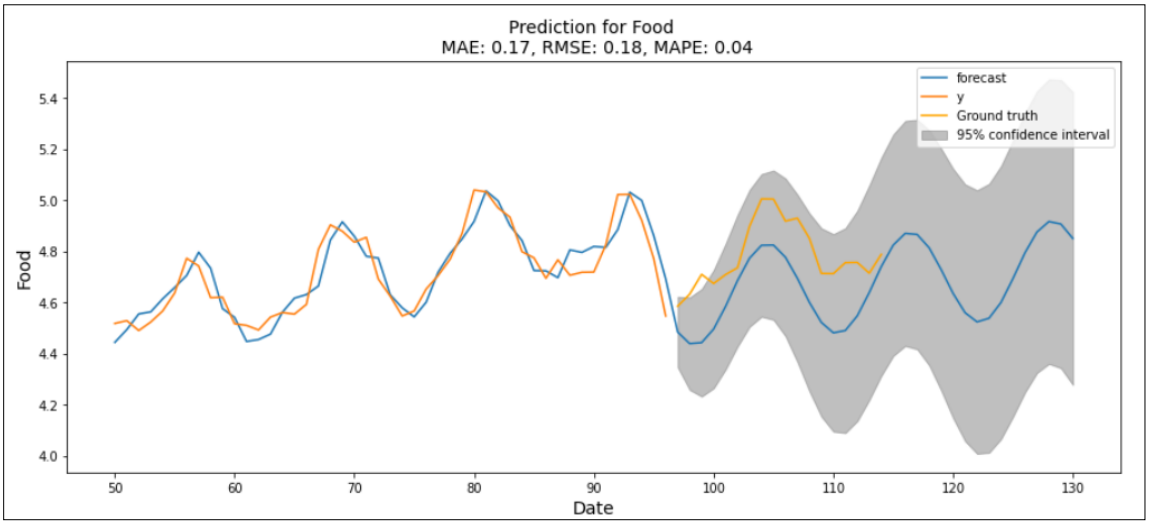


Fig 2.1 Prediction for Food Sector using ARIMA model

4.1.2 UBC ANALYSIS

The UBC analysis differs from the NIC in the way that they have different fundamental attributes. The NIC speaks about the different industrial sectors involved whereas the UBC speaks about the use-based sectors. So, for analyzing the various sectors, the first step is to understand the data set and perform basic pre-processing, visualization techniques to get some more insights such as line charts, violin charts, pyramid plots, etc., for the various use-based sectors.

Next, the stationarity of the dataset was checked visually using rolling mean and variance. Given the intricacy involved with visual attestation, the ADF test was performed to confirm the presence of stationarity of values. The ADF test is a statistical method also known as unit root test. Similarly, to the NIC analysis to handle data that were not stationary, differencing was the better approach over transformation, so different orders of differencing were performed on the different use-based sectors until stationarity was attained.

Since most time series models, including the ARIMA model have stationarity of the dataset as a prerequisite for proper functioning of the model, after differencing the dataset is ready to be modelled on. Following a similar procedure to the one mentioned above, the modelling phases begins with displaying the ACF/PACF plots and identifying the Auto Regression and Moving Average. Next, the Integrating Factor is selected along with additional parameters required. With this, the ARIMA is modelled for the dataset, and prediction is performed.

To ensure that the ARIMA model and the selected parameters are optimal, Auto-ARIMA is used on the dataset after preprocessing. Auto-ARIMA learns the parameters on its own by undergoing various iterations given a range of values and determines the best possible parameters for the dataset. Using this, validation can be done on the selected parameters and further fine-tuning of the parameters can be done.

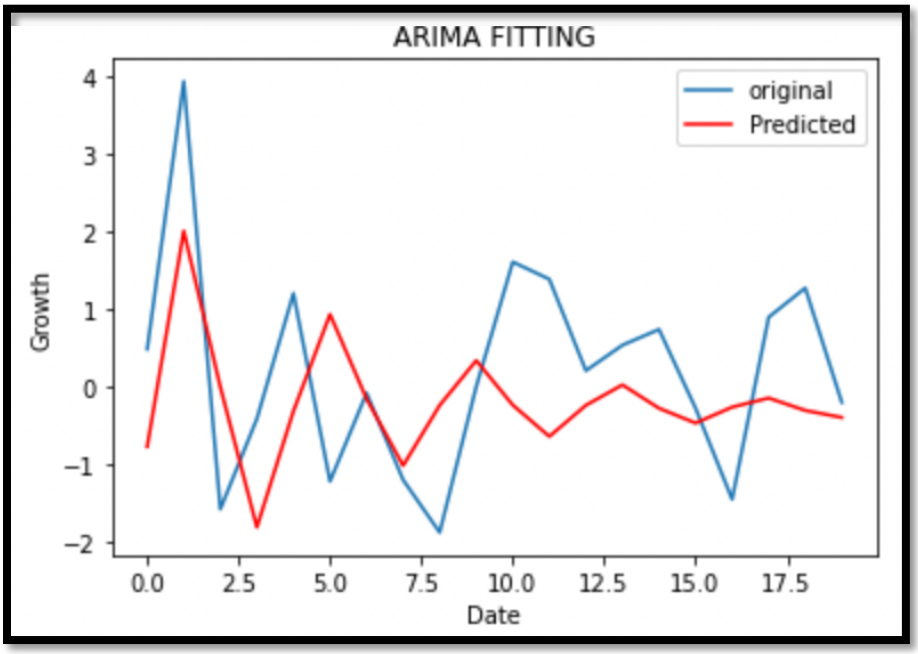


Fig 2.2 Growth vs Date for use-based sectors using ARIMA fitting

4.2 SECTOR ANALYSIS

The previous chapter was concluded with a good understanding of the basics of the stock market. This chapter deals with performing generic predictive analysis on a selected sector. For simplicity, the sector analysis was performed for top performing companies from a selected industry – Finance in this case. The companies selected from this sector included Axis Bank, HDFC Bank, ICICI Bank, etc. For this sector analysis, the dataset was called from Yahoo Finance, and predictive analysis was performed on it.

Techniques that were used in the analysis included:

- Firstly, the performance of the companies needs to be compared (especially since they belong to the same sector) by using contrasting visualization techniques such as line charts.
- Next, for understanding the volatility involved within a company, different Moving Averages need to be plotted (ideally three moving averages are appreciated – short-term, mid-term and long-term) for the various companies simultaneously.
- Finally, the extent of influence of one stock with respect to another stock is checked through the help of plots such as Kernel Density Estimate (KDE) plot, scatter plot, etc.

Through these relative comparisons the patterns among the competitive nature of the companies can be estimated. For example, we should not invest in stock X while stock Y in growing.

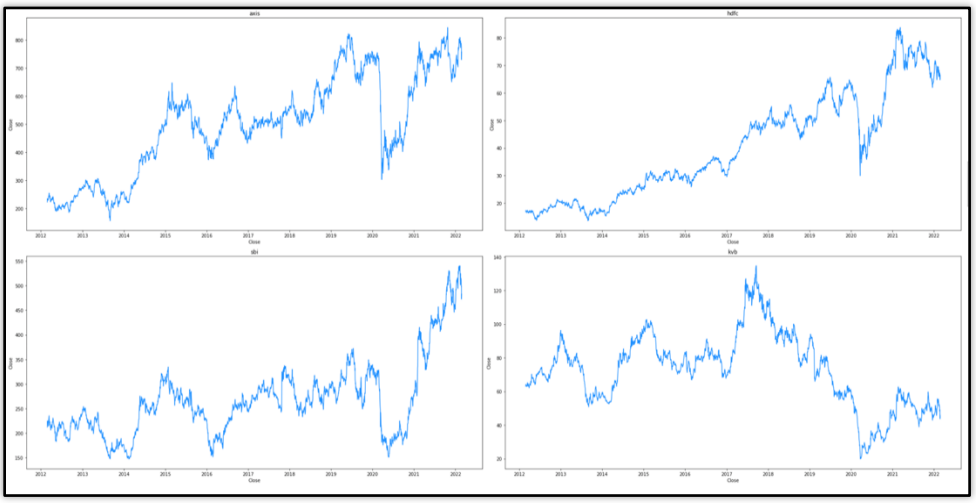


Fig 3.1 Line charts for comparing the stock prices of four different banks

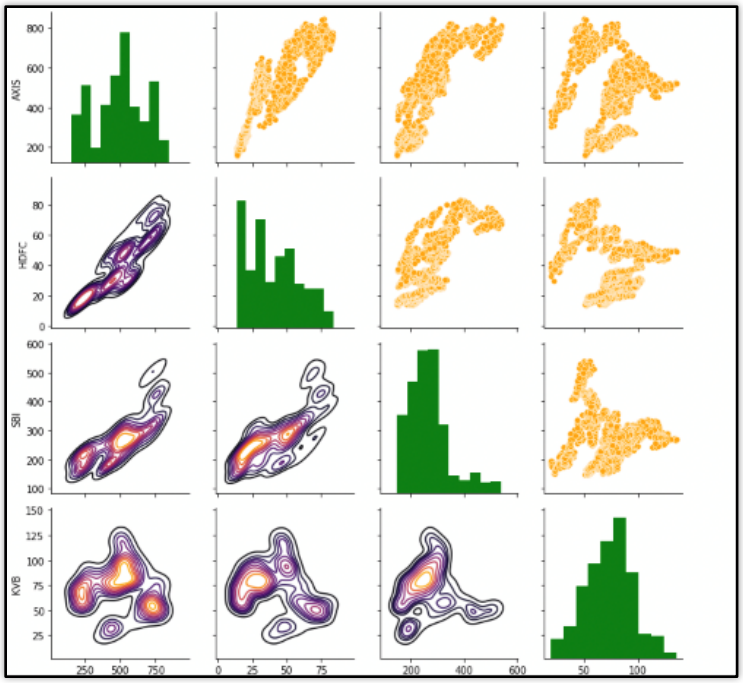


Fig 3.2 KDE charts, bar charts, and scatter plots for understanding the correlation between four banks

4.3. COMPANY ANALYSIS

4.3.1 INTRODUCTION

The main aim of the project is to figure out the overall performance of a company belonging to a particular sector and also determine the best performing companies in that sector. As there are various parameters and factors that determine the standing of a company in a sector, The analysis is an ensemble of results from various analyses comprising of multiple perspectives.

The three different analyses performed are:

- Customer Sentiment Analysis
- Stock Market Analysis
- Company Growth Analysis

4.3.2 CUSTOMER SENTIMENT ANALYSIS

Sentiment analysis is a widely used Natural language Understanding technique to estimate the currently prevailing sentiments of the customers as a collective entity. On simpler terms it can be used to estimate the context and responses for a particular company's current actions. Analyzing customer sentiments by understanding and processing them can give inferences that can't be provided by strict parameters that are present in conventional datasets.

Sentimental Analysis generally involves these steps:

- Collection of relevant data: This step could be done through various API services provided by various service providers. Recent News articles, feeds and customer responses will be generated to the respective keywords (topic) user provides. Example for these services include (Google News, TwitterAPI).
- Processing of data: The data collected must be preprocessed before performing analysis or training. Natural language processing techniques like Tokenization, stop words Removal, Stemming, Lemmatization are performed, and a bag of words model is created to ensure efficient processing of data.
- Vectorization: Machine learning models cannot process text directly.
  Vectorization is the process of converting texts into binary form based on various parameters, which can be either count of the words in text or frequency of words, etc. The vectorizers popularly used are Count Vectorizer and Tfid Vectorizer.
- Classification: The scraped data could be either with labels or unlabeled. Text sentiment classification models like Vader Sentiment Intensity analyzer or BERT are used to classify unlabeled text from various sources.

  For text with labels (E.g.: customer review data sets), Machine learning models could be trained by using vectorized form of texts and labels. The Models learn from the sentiments and the labels and could be used to classify future sentiments.

The most frequently used models be (Multinomial Naïve Bayes, Random forests, Etc.).

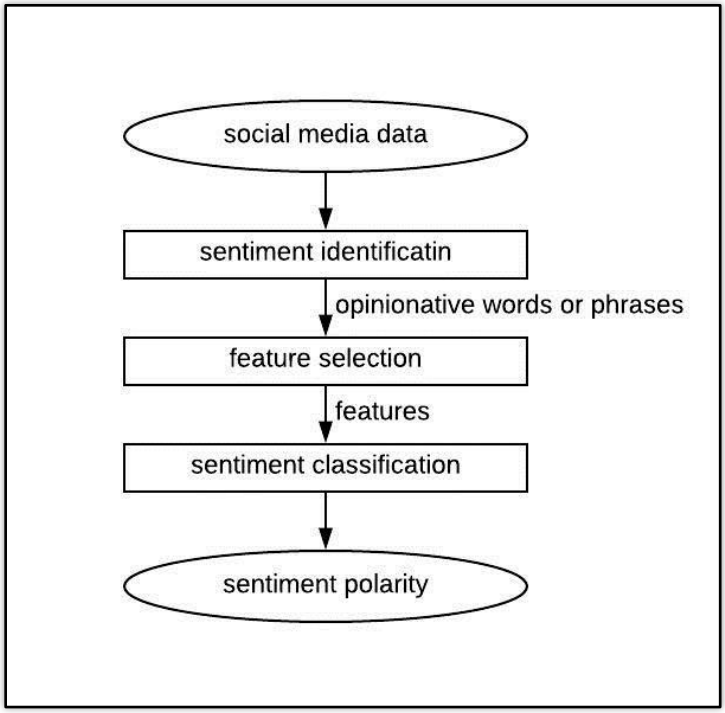


Fig 4.1 Phases involved in Sentiment Analysis

4.3.2.1 Workflow

- Data is generated for the company specified by the user through Google News API and the trending news from the past 30 are extracted from the links of their respective sites.
- Data collected is organized in the form of Data Frames and the 'Summary' column of the table is considered for performing sentiment analysis.
- The text is pre-processed with NLP tools available in the NLTK library.
  And vectorized using Count Vectorizer available in the Scikit learn package.
- After preprocessing the data, unlabeled classification model VADER is applied to get the context of the text.
- Output of the Vader Sentiment Intensity analyzer would provide the probability distribution of the text with respect to their context. Example {pos: 0.46, neut: 0.13, neg: 0.41}. It indicates the probability of the text being a positive sentiment is 46% and it being neutral and negative sentiments are 13% and 41% respectively.
- Based on the results, Pie Graphs are plotted to give sophisticated visualizations to the users.

4.3.3. STOCK MARKET ANALYSIS

The stock market broadly refers to the collection of exchanges and other venues where the buying, selling, and insurance of shares of publicly held companies take place. Stock prices can determine a company's performance with regards to the above-mentioned aspects. Stock market data of a company contains, opening price of a stock, closing price, volume, and adjacent closing price for a period (Time frame) as the features.

4.3.3.1 Overall process

- Stock market data of a company is extracted for a specified time.
- Exploratory data analysis for finding out the daily fluctuations in the prices, finding out moving averages, finding Relative strength index, Correlation among similar stocks is performed.
- Inferences made from the Exploratory Data analysis are used to determine best performing stocks in a particular sector.
- The inferences could be very useful for people who try to invest money in the stocks, and it could be used for building custom models to predict future stock prices.

4.3.3.2 Workflow

- Stock prices of the user specified company is extracted from yfinance API by using symbol references that are provided through http messages.
- Linear visualizations of the prices are plotted in the chronological order to provide users with insights about the recent and current trends.
- Moving average is calculated and plotted along with original values to provide more insights about trends.
- Increase/decrease trends are plotted to understand the fluctuations in prices

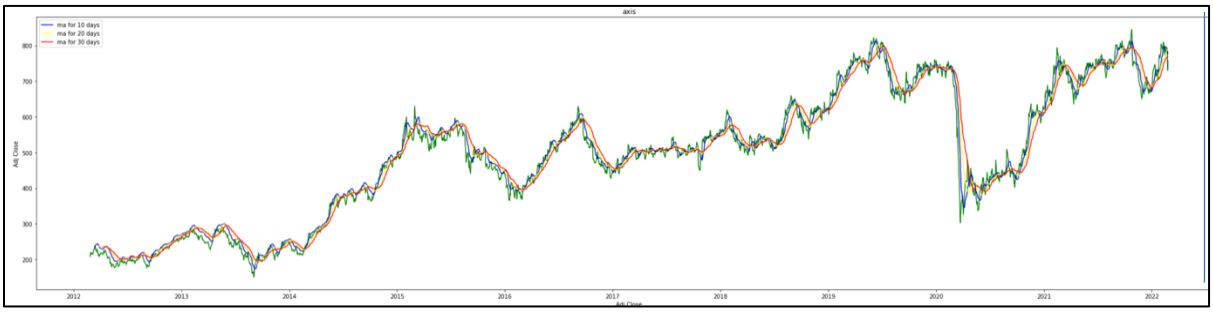


Fig 4.2 Example for Moving Averages of Stocks

- During certain cases performance of one company in the same sector can influence others with a negative correlation so it would be more insightful to plot Pair Grid plot with many companies and graphs like KDE plots and Distplots can be used to compare influences exerted by every company on one another.

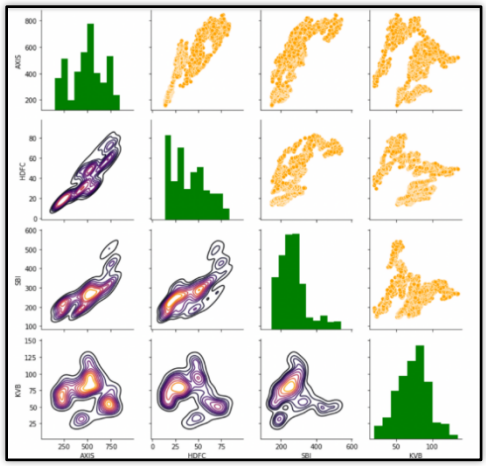


Fig 4.3 Pair grid between stocks

4.3.4. COMPANY GROWTH ANALYSIS

Analyzing company's growth over periods is one of the most conventional and practical methods comparison to various methods available. Since the features used are not as volatile as stock market data and customer sentiments, more Robust insights can be drawn. The features include monthly, quarterly, and annual values of Profit margins of a company, Gross Profits, Net income, etc.

4.3.4.1 Data sets

- Datasets are extracted from yahoofin API.

4.3.4.2 Plotting to be done

- Line graphs with time and date as x-axis and individual parameters as y-axis provides the picture about the current trends in revenue or the overall performance of a company.
- Seasonal trends and changes about the data could be visualized using seasonal models from statsmodel package. It contains plots for seasonal changes, cyclic changes, random values, residual error in the dataset.

4.3.4.3 ARIMA Model

- Arima model is one of the most powerful models in time series estimation. ARIMA stands for (Auto Regressive Integrated Moving Average).
- ARIMA uses a combination of both auto regression and moving averages of the past values to predict the successive outcomes.
- Auto Regression is the correlation between data and its lagged values.

$$y_t = \beta_0 + \beta_1 y_{t-1} + \varepsilon_t$$

Yt -> value at time t

Yt-1 -> value at time t-1
β0 and β1 are learnable parameters
εt is the error (Noise)

- Moving average is the average between values of a particular time (Moving Window). Formula for Simple Moving Average (SMA) is given by,

$$\mathrm{SMA} = \frac{A_1 + A_2 + \ldots + A_n}{N}$$

- I in ARIMA specifies the order of differencing. I = 1 indicates the usage of 1 step lagged values and I = 2 indicates the usage of 2 stepped lagged values. Formula for ARIMA model is given by,

$$r_t = \phi_1 + \phi_1 r_{t-1} + \ldots \phi_p r_{t-p} + \theta_1 \varepsilon_{t-1} + \ldots + \theta_q \varepsilon_{t-q} + \varepsilon_t$$

4.4. DASHBOARD CREATION

4.4.1 NEED FOR THE DASHBOARD

What most of customer needs is the company's data. Company's data along with the dashboard provides the on-demand access of all important metrics. Users can see the detailed overview of the business and hence reduce their time in compiling the reports. Not only reducing the time but also users could take better decisions if they had the unbiased view of overall company's performance. These can be done with the help of dashboard creation. The other usage included are the Solutions that users can get about the ROI and to make sense of the data that they were tracking of. Also, it enables quick access to key performance indicators (KPIs) related to a certain goal or business activity.

4.4.2. STREAMLIT

Streamlit is a free, open-source, all-Python framework that allows easily constructing interactive dashboards and machine learning web apps with no prior knowledge in front-end web development. It has a diverse set of UI components. It includes practically every standard user interface component, such as a checkbox, slider, collapsible sidebar, radio buttons, file upload, progress bar, and so on. Furthermore, these components are quite simple to utilize.

4.4.3. DASHBOARD INTEGRATION USING STREAMLIT

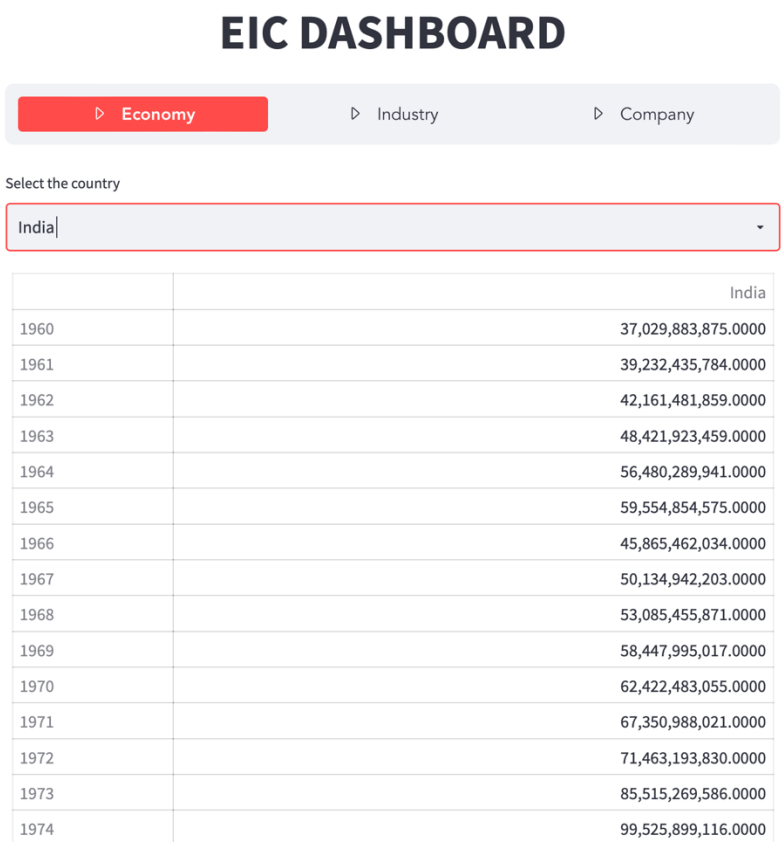
Creating Dashboard with the help of Streamlit can be done by installing the streamlit and then installing some necessary libraries for data reading, manipulation, and visualization. After that Creating a python file for the web application. This web application file will be used by streamlit to host the dashboard on the local server. The application file is deployed in the local server and later the dashboard can be viewed.

CHAPTER 5

RESULTS AND COMPARISON

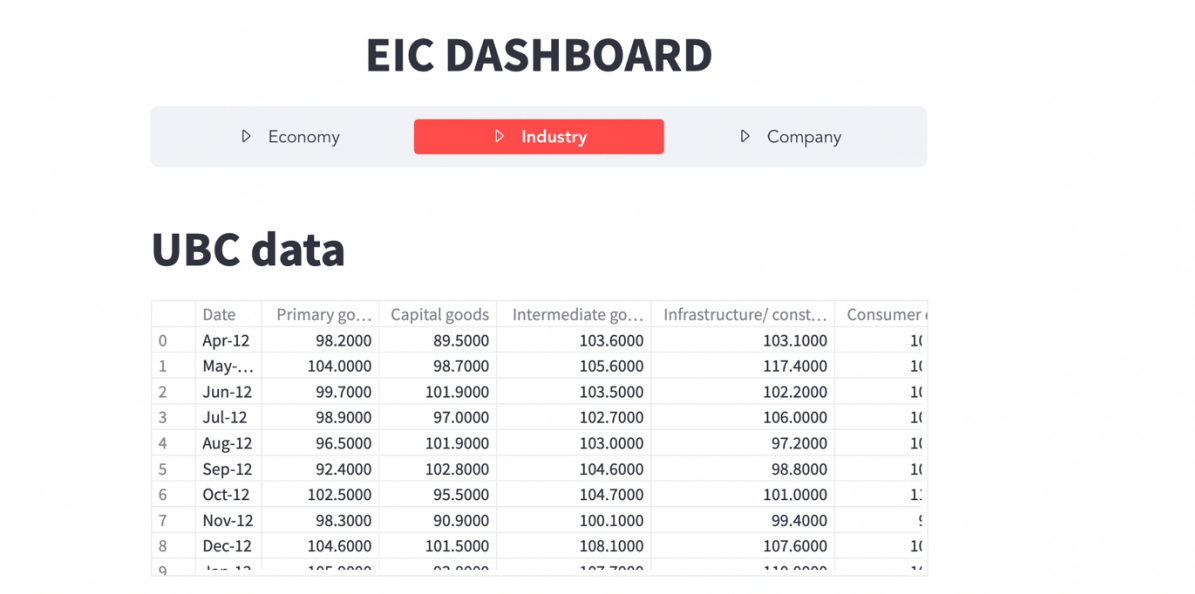
5.1 DASHBOARD DESIGN

The Dashboard includes the aggregation of the most important plots and datasets required for the EIC analysis. The home page of the dashboard contains three sections which are Economy, Sector and Company analysis respectively.



	India
1960	37,029,883,875.0000
1961	39,232,435,784.0000
1962	42,161,481,859.0000
1963	48,421,923,459.0000
1964	56,480,289,941.0000
1965	59,554,854,575.0000
1966	45,865,462,034.0000
1967	50,134,942,203.0000
1968	53,085,455,871.0000
1969	58,447,995,017.0000
1970	62,422,483,055.0000
1971	67,350,988,021.0000
1972	71,463,193,830.0000
1973	85,515,269,586.0000
1974	99,525,899,116.0000

In Economy analysis users can select any country and view their GDP measures between 1960-2020.



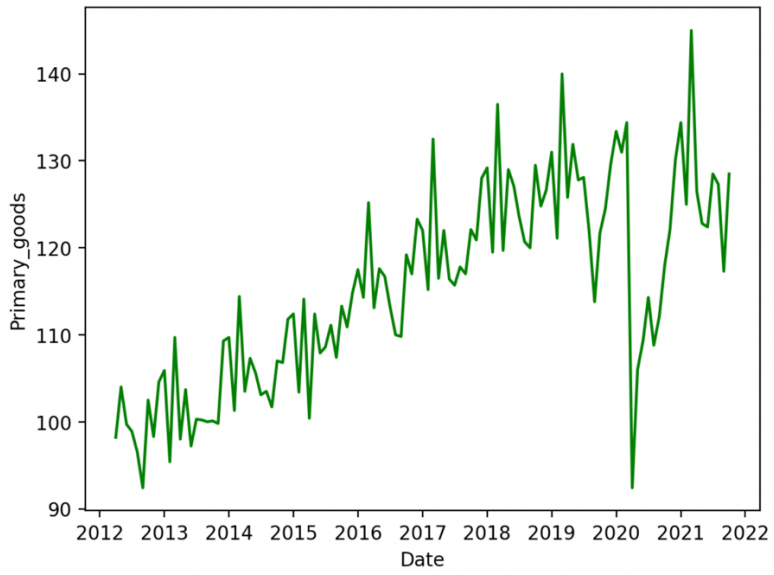
	Date	Primary go…	Capital goods	Intermediate go…	Infrastructure/ const…	Consumer
0	Apr-12	98.2000	89.5000	103.6000	103.1000	1(
1	May-…	104.0000	98.7000	105.6000	117.4000	1(
2	Jun-12	99.7000	101.9000	103.5000	102.2000	1(
3	Jul-12	98.9000	97.0000	102.7000	106.0000	1(
4	Aug-12	96.5000	101.9000	103.0000	97.2000	1(
5	Sep-12	92.4000	102.8000	104.6000	98.8000	1(
6	Oct-12	102.5000	95.5000	104.7000	101.0000	1
7	Nov-12	98.3000	90.9000	100.1000	99.4000	9
8	Dec-12	104.6000	101.5000	108.1000	107.6000	1(

Industry analysis contains detailed analysis of UBC and NIC indices covering various sectors. Users can select a specific sector they are interested in can compare its growth with other sectors using plots like line plots and seasonal decomposition graphs.

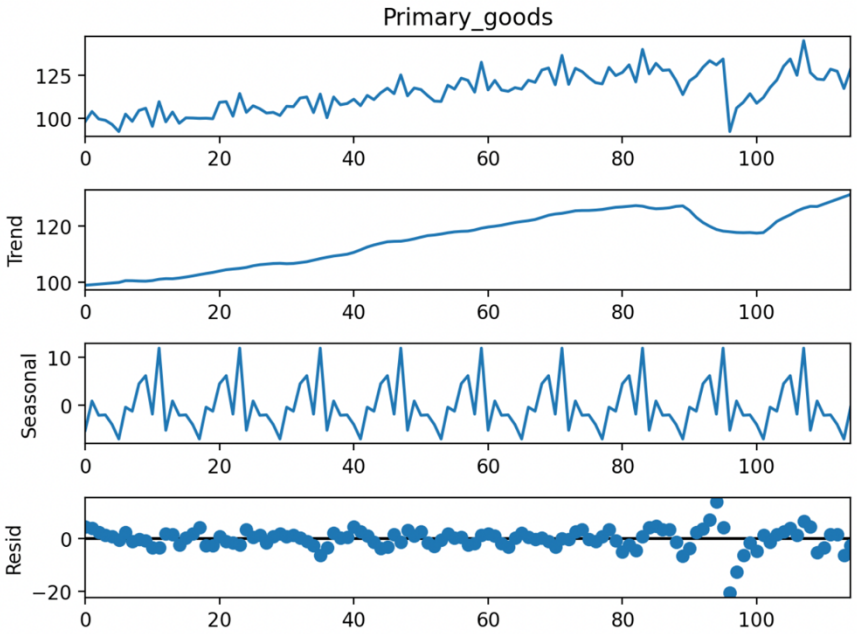
Choose the industry

Primary_goods

Primary_goods



Seasonal Decomposition



NIC indices are similar to UBC indices but the major difference between them is that UBC indices cover only the major sectors like (primary goods, capital goods, etc) whereas NIC covers for almost every sector.

NIC data

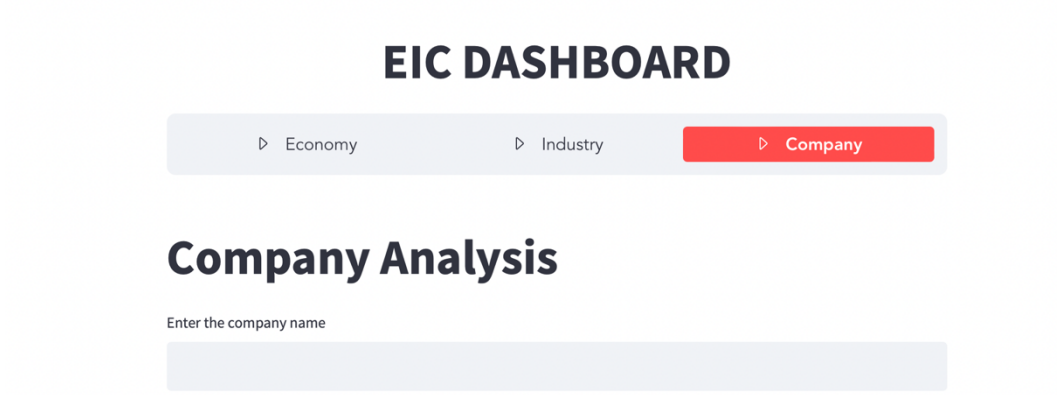
	Date	Manufacture of food …	Manufacture of beve…	Manufacture of toba…	Manufacture o
0	Apr-12	97.3000	134.2000	105.0000	
1	May-…	91.7000	147.1000	105.6000	
2	Jun-12	86.1000	130.5000	96.3000	
3	Jul-12	88.1000	93.1000	96.5000	
4	Aug-12	85.9000	85.1000	86.2000	
5	Sep-12	84.1000	82.3000	97.9000	
6	Oct-12	98.2000	96.0000	102.3000	
7	Nov-12	109.2000	90.2000	115.3000	
8	Dec-12	132.8000	91.9000	117.9000	

Choose sector

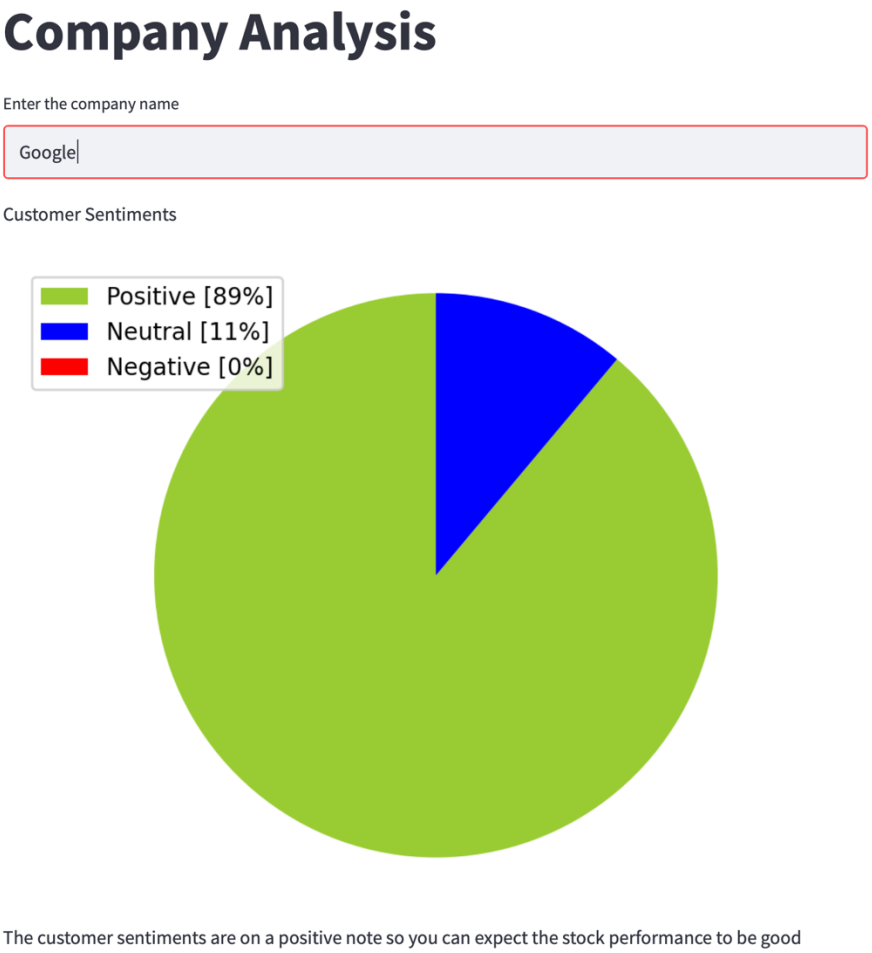
Food

- Food
- Beverages
- Tobacco
- Textiles
- Wearing_Apparel
- Leather
- Wood_and_Cork

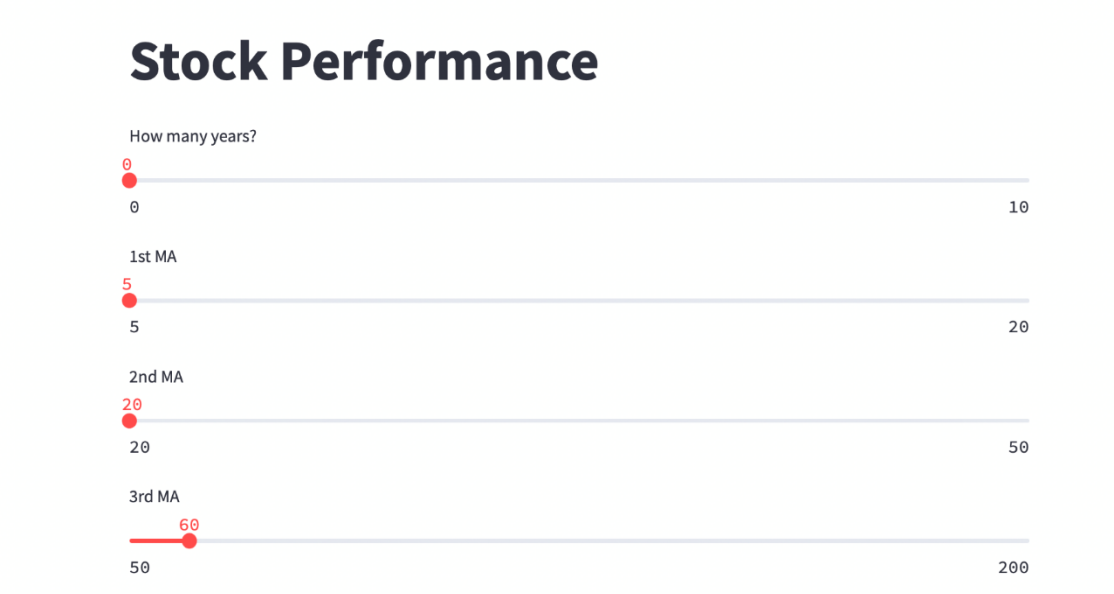
Company analysis



In company analysis users can enter a company name of their interest and the dashboard fetches various details of the company dynamically through different APIs. Sentiment analysis, Stock market analysis and Overall growth analysis of the company would be presented to the user.

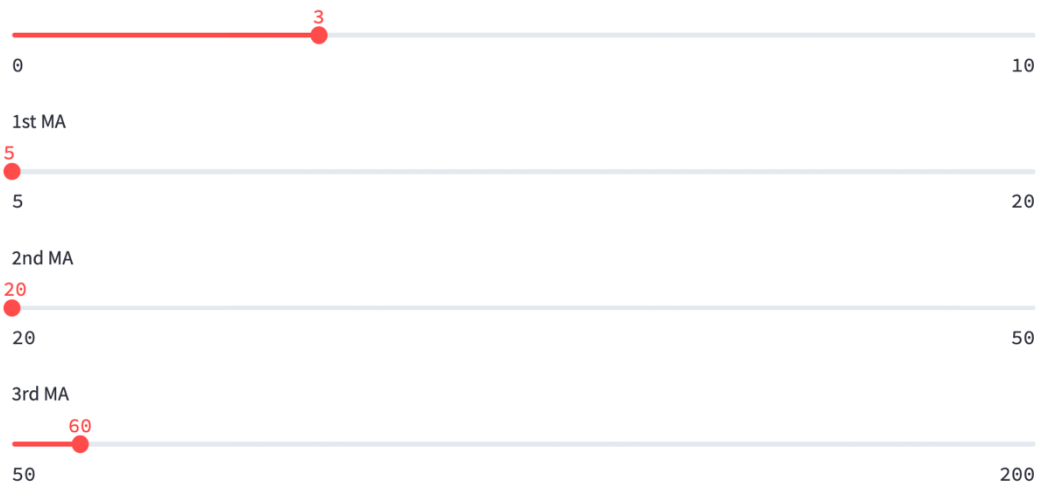


Sentiment analysis contains the pie charts plotted on the aggregated results from Vader sentiment intensity analysis for the news articles collected through Google News API.



In stock market analysis Users can set the time period that specify the number of years the API must download the respective company's stock market data.

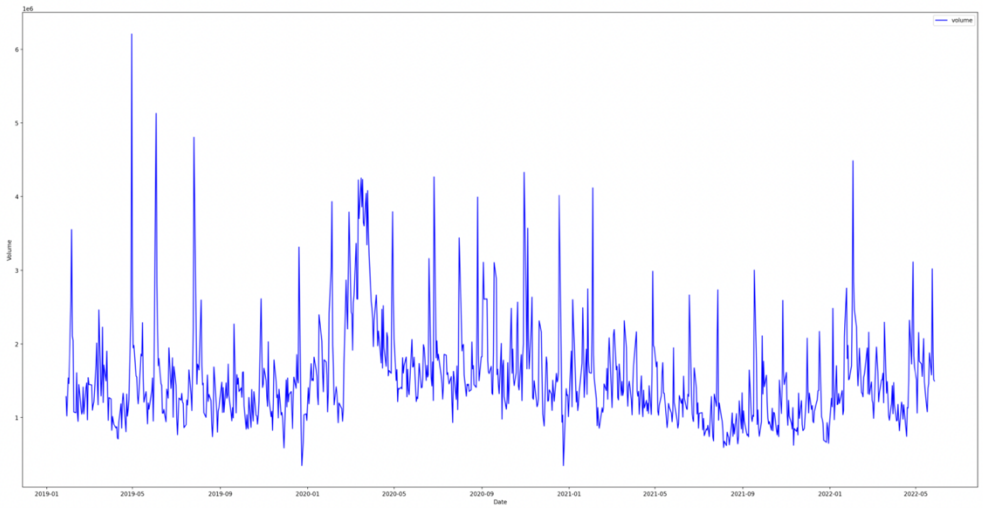
The time period ranges from 0 – 10 years. By default, the value would be 0 years, So the yfinance API would fetch the current years stock market data.



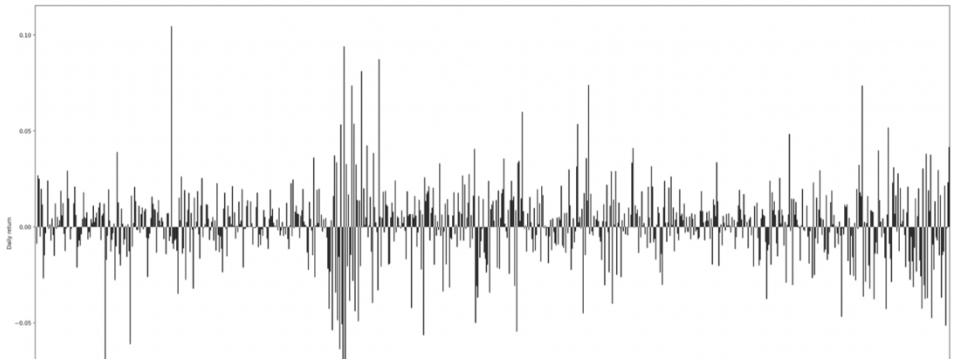
Moving averages for Google

The moving averages model for stock market prediction involves users to select the moving average window for Moving average 1,2 and 3 respectively.

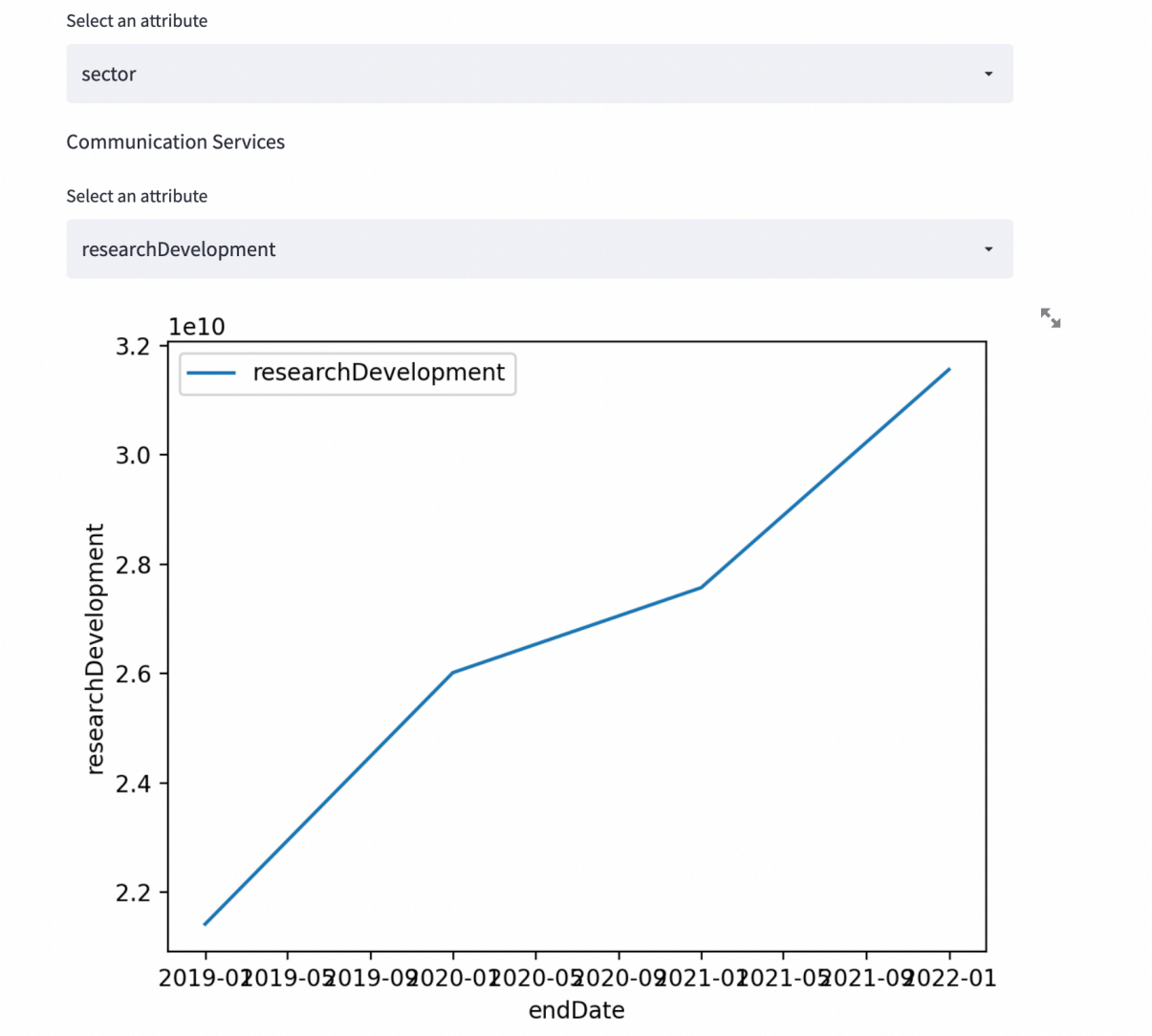
Volume for Google



Percentage Change for Google



Volume of the stock and Daily changes in the closing value of the stock are some of the important parameters in stock market analysis.



In the overall growth analysis, Users can select an attribute of their choice and view the value of the attribute for the respective company. Various indices of the company like amount spent for the research and development, Gross profits, net income can be plotted.

BIBLIOGRAPHY

1. Strader, Troy J.; Rozycki, John J.; ROOT, THOMAS H.; and Huang, Yu-Hsiang (John) (2020) "Machine Learning Stock Market Prediction Studies: Review and Research Directions," Journal of International Technology and Information Management: Vol. 28 : Iss. 4 ,
2. Naman Adlakha; Ridhima; Avita Katal, IEEE, 2021, “Real Time Stock Market Analysis”, https://ieeexplore.ieee.org/document/9526506.
3. Financial Markets Prediction with Deep Learning. Jia Wang, Tong Sun, Benyuan Liu, Yu Cao, Degang Wang [v1] Mon, 5 Apr 2021 19:36:48 UTC.
4. Effectiveness of Artificial Intelligence in Stock Market Prediction based on Machine Learning Sohrab Mokhtari, Kang K. Yen, Jin Liu [v1] Wed, 30 Jun 2021 19:58:03 UTC.
5. The prediction system for data analysis of stock market by using Genetic Algorithm Ching-Te Wang, Yung-Yu Lin.
6. Stock Market Prediction Analysis by Incorporating Social and News Opinion and Sentiment Zhaoxia Wang, Seng-Beng Ho, Zhiping Lin.
7. Dashboards & Marketing: Why, what, How and Which Research is Needed? Koen Pauwels, Tim Ambler, Bruce H. Clark, Pat LaPointe January 2009 Journal of Service Research 12(2):175-189.

APPENDICES

Preprocessing:

Libraries installed:

For preprocessing the data that certain libraries are required to pre-process it. They include matplotlib.pyplot, numpy, seaborn

Displaying the data

For displaying the data obtained after preprocessing.

```
[ ]  import matplotlib.pyplot as plt
     import numpy as np
     import seaborn as sns
```

Grouping the individual Sectors

Taking individual sectors with the sum, across all quarters and grouping them by year

```
]  df_yearly_Agriculture_Forestry_and_Fishing = pd.DataFrame(df_yearly.groupby('Year')['Agriculture, Forestry and Fishing'].sum())
```

Merging the sectors

For merging the various sectors like “Electricity, gas, water supply and utility” under manufacturing and Trade, Hotels, Transport, Communication and Services Related to Broadcasting” under Construction and the other sectors like Financial, Real Estate and Professional Services and Public Administration, Defence and Other Services

```
[ ] df_yearly_sum = df_yearly_Mining_and_Quarrying.merge(df_yearly_Agriculture_Forestry_and_Fishing, on="Year", how="left")

[ ] df_yearly_Manufacturing = pd.DataFrame(df_yearly.groupby('Year')['Manufacturing'].sum())

[ ] df_yearly_sum = df_yearly_sum.merge(df_yearly_Manufacturing, on="Year", how="left")

[ ] df_yearly_Electricity_Gas_Water_Supply_and_Other_Utility = pd.DataFrame(df_yearly.groupby('Year')['Electricity, Gas, Water Supply & Other Utility'].sum())

[ ] df_yearly_sum = df_yearly_sum.merge(df_yearly_Electricity_Gas_Water_Supply_and_Other_Utility, on="Year", how="left")

[ ] df_yearly_Construction = pd.DataFrame(df_yearly.groupby('Year')['Construction'].sum())

[ ] df_yearly_sum = df_yearly_sum.merge(df_yearly_Construction, on="Year", how="left")

[ ] _Communication_and_Services_Related_to_Broadcasting = pd.DataFrame(df_yearly.groupby('Year')['Trade, Hotels, Transport, Communication and Services Related to Broadca

[ ] df_yearly_sum = df_yearly_sum.merge(df_yearly_Trade_Hotels_Transport_Communication_and_Services_Related_to_Broadcasting, on="Year", how="left")
```

Visualization

```
f, ax = plt.subplots(nrows=8, ncols=1, figsize=(15, 25))
plt.subplots_adjust(hspace = 0.35)
index=1

for i, column in enumerate(df_yearly_sum.drop('Year', axis=1).columns):
    sns.lineplot(x=df_yearly_sum['Year'], y=df_yearly_sum[column].fillna(method='ffill'), ax=ax[i], color='dodgerblue')
    ax[i].set_title('{}'.format(description[index]), fontsize=14)
    ax[i].set_ylabel(ylabel='GDP', fontsize=14)
    index = index + 1
```

Sector analysis line plots

To analyse the sectors with the help of line plots for knowing the relative trends in the industry against the normalized GDP values

```
[ ] plt.figure(figsize =(10, 5))
    plt.plot(Year,Mining,label='Mining',color = 'maroon')
    plt.plot(Year,Agriculture,label='Agriculture',color = 'pink')
    plt.plot(Year,Manufacturing,label='Manufacturing',color = 'green' )
    plt.plot(Year,Electricity,label='Electricity',color = 'red')
    plt.plot(Year,Construction,label='Construction',color = 'black')
    plt.plot(Year,Trade,label='Trade',color = 'yellow')
    plt.plot(Year,Financial,label='Financial',color = 'darkblue')
    plt.plot(Year,Public,label='Public',color = 'lawngreen')
    plt.title("Sector Analysis - Line plots")
    plt.xlabel("Year")
    plt.ylabel("Normalized GDP values")
    plt.legend(bbox_to_anchor=(1.05, 1))
    plt.show()
```

Heatmaps

A heatmap is a graphical representation of data that uses a system of color-coding to represent different values.

```
[ ] plt.figure(figsize = (13,13))
    ax = sns.heatmap(df_yearly_sum.corr().iloc[0:10, 0:10], annot=True, linewidths=.4)
    plt.tight_layout()
```

Data preparation

Downloading Stock data of (HDFC, Axis Bank, SBI, KVB).

```
[ ]  stock_list = ['AXISBANK.NS','HDB','SBIN.NS','KARURVYSYA.NS']

    end = datetime.now()
    start = datetime(end.year-10,end.month,end.day)

    for stock in stock_list:
        globals()[stock] = yf.download(stock ,start ,end)
```

Volume of stocks

The amount of stocks sold or bought for a particular time is plotted in a graph

```
[ ]  fig,ax = plt.subplots(2,2,figsize=(30,20))
     #axis
     sns.lineplot(x = axis.index,y=axis['Volume'],ax=ax[0][0],color='green')
     ax[0][0].set_title('axis')
     ax[0][0].set_xlabel(None)
     ax[0][0].set_xlabel('Volume')
     # hdfc
     sns.lineplot(x = hdfc.index,y=hdfc['Volume'],ax=ax[0][1],color='green')
     ax[0][1].set_title('hdfc')
     ax[0][1].set_xlabel(None)
     ax[0][1].set_xlabel('Volume')
     # sbi
     sns.lineplot(x = sbi.index,y=sbi['Volume'],ax=ax[1][0],color='green')
     ax[1][0].set_title('sbi')
     ax[1][0].set_xlabel(None)
     ax[1][0].set_xlabel('Volume')
     # kvb
     sns.lineplot(x = kvb.index,y=kvb['Close'],ax=ax[1][1],color='green')
     ax[1][1].set_title('kvb')
     ax[1][1].set_xlabel(None)
     ax[1][1].set_xlabel('Volume')

     plt.tight_layout()
```

Calculating percentage change

The percentage change involved in the stocks will help in calculating the risk involved.

```
axis['Daily return'] = axis['Adj Close'].pct_change()
hdfc['Daily return'] = hdfc['Adj Close'].pct_change()
sbi['Daily return'] = sbi['Adj Close'].pct_change()
kvb['Daily return'] = kvb['Adj Close'].pct_change()

kvb.head()
```

Distribution of returns

The distribution of the returns on investment depends on the rules in the economic system.

```
[ ] fig,ax = plt.subplots(4,1,figsize=(35,40))
    #axis
    sns.distplot(x=axis['Daily return'],ax=ax[0],color='orange')
    ax[0].set_title('axis')
    ax[0].set_xlabel(None)
    ax[0].set_ylabel('Daily return')
    # hdfc
    sns.distplot(x = hdfc['Daily return'],ax=ax[1],color='orange')
    ax[1].set_title('hdfc')
    ax[1].set_xlabel(None)
    ax[1].set_ylabel('Daily return')
    # sbi
    sns.distplot(x = sbi['Daily return'],ax=ax[2],color='orange')
    ax[2].set_title('sbi')
    ax[2].set_xlabel(None)
    ax[2].set_ylabel('Daily return')
    # kvb
    sns.distplot(x =kvb['Daily return'],ax=ax[3],color='orange')
    ax[3].set_title('kvb')
    ax[3].set_xlabel(None)
    ax[3].set_ylabel('Daily return')

    plt.tight_layout()
```

Correlation between all stocks

Stock correlation describes the relationship that exists between two stocks and their respective price movements.

```
fig = sns.PairGrid(close_stocks)
fig.map_diag(plt.hist,color='green')
fig.map_upper(sns.scatterplot,color='orange')
fig.map_lower(sns.kdeplot,cmap='inferno')
```

Model

Stock Price Prediction using machine learning helps to discover the future value of company stock and other financial assets traded on an exchange.

```
[ ] plt.figure(figsize=(10,7))
    plt.plot(y_test,color='dodgerblue',label='original')
    plt.plot(y_predict,color='red',label='predicted')
    plt.title('Predicted values')
    plt.show()
```

Candlestick

Each candlestick represents one day's worth of price data about a stock through four pieces of information: the opening price, the closing price, the high price, and the low price.

```
[ ] cs=pl.Candlestick(x=gj_df['Date'],low=gj_df['Low'],high=gj_df['High'],close=gj_df['Close'],open=gj_df['Open'])
    fig=pl.Figure(data=[cs])
    fig.show()
```

Sentiment analysis

NLP Processing steps:

Natural Language Processing (NLP) is a field of data science and artificial intelligence which can be used to process and understand texts in natural language.

```python
from nltk.corpus import stopwords
en_stopwords = stopwords.words('english')

def remove_stopwords(text):
    li = []
    for word in text:
        if word not in en_stopwords:
            li.append(word)
    return li
```

Removing the special characters:

```python
from nltk.tokenize import RegexpTokenizer

def remove_punct(text):

    tokenizer = RegexpTokenizer(r"\w+")
    lst=tokenizer.tokenize(' '.join(text))
    return lst
```

Lemmatization

The process of grouping together the inflected forms of a word so they can be analysed as a single item, identified by the word's lemma, or dictionary form.

```python
from nltk.stem import WordNetLemmatizer
from nltk import word_tokenize,pos_tag

def lemmatization(text):

    result=[]
    wordnet = WordNetLemmatizer()
    for token,tag in pos_tag(text):
        pos=tag[0].lower()

        if pos not in ['a', 'r', 'n', 'v']:
            pos='n'

        result.append(wordnet.lemmatize(token,pos))

    return result
```

NLP Vader Classification

Valence Aware Dictionary and Sentiment Reasoner is a lexicon and rule-based sentiment analysis tool that is specifically attuned to sentiments expressed in social media.

```python
#Sentiment Analysis
def percentage(part,whole):
    return 100 * float(part)/float(whole)

#Assigning Initial Values
positive = 0
negative = 0
neutral = 0
#Creating empty lists
news_list = []
neutral_list = []
negative_list = []
positive_list = []

#Iterating over the tweets in the dataframe
for score in sentiments['scores']:

    if score['neg'] > score['pos']:
        negative += 1 #increasing the count by 1
    elif score['pos'] > score['neg']:
        positive += 1 #increasing the count by 1
    elif score['pos'] == score['neg']:
        neutral += 1 #increasing the count by 1

positive = percentage(positive,len(sentiments['text'])) #percentage is the function defined above
negative = percentage(negative,len(sentiments['text']))
neutral = percentage(neutral,len(sentiments['text']))

#Converting lists to pandas dataframe

#using len(length) function for counting
print("Positive Sentiment:", '%.2f' % positive, end='\n')
print("Neutral Sentiment:", '%.2f' % neutral, end='\n')
print("Negative Sentiment:", '%.2f' % negative, end='\n')

#Creating PieCart
labels = ['Positive ['+str(round(positive))+'%]' , 'Neutral ['+str(round(neutral))+'%]','Negative ['+str(round(negative))+'%]']
sizes = [positive, neutral, negative]
colors = ['yellowgreen', 'blue','red']
patches, texts = plt.pie(sizes,colors=colors, startangle=90)
plt.style.use('default')
plt.legend(labels)
plt.title("Sentiment Analysis Result for stock= "+company_name+"" )
```

Streamlit:

Streamlit is an open source app framework in Python language. It helps us create web apps for data science and machine learning in a short time.

Example Snippet:

```
st.title("Stock Market Analysis")

start = '2010-01-01'
end = '2019-12-31'

st.subheader("Date from 2010 - 2019")

user_input = st.text_input("Enter Stock Ticker", 'AAPL')
df = data.DataReader(user_input,'yahoo',start, end)

st.write("Data values of "+user_input)
st.write(df.head())

st.write("Describing the data values of"+user_input)
st.write(df.describe())
```

```
st.subheader('Closing Price Vs Time chart with ma100')
ma100 = df.Close.rolling(100).mean()
fig = plt.figure(figsize = (12,6))
plt.plot(df.Close,'r')
plt.plot(ma100,'b')
#st.pyplot.legend(["close","ma100"])
st.pyplot(fig)
```