

STOCK FORECASTING USING MACHINE LEARNING MODEL

Project report submitted in partial fulfillment of the requirement of degree of

BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE & ENGINEERING

By

Charvi Sharda (181399)

UNDER THE GUIDANCE OF

Dr. Amol Vasudeva



JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY, WAKNAGHAT

MAY 2022

TABLE OF CONTENTS

Chapter 1 INTRODUCTION

- Introduction
- Problem Statement
- Objectives
- Methodology

Figure 1.4: Proposed framework

- Organization

Chapter 2 LITERATURE REVIEW

- Technical Analysis & Sentiment Embeddings for Market Trend Prediction [1]
- Prediction Models for Indian Stock Market [2]

NSE Stock Market Prediction Using Deep-Learning Models [3] 2.4

Aggregating multiple types of complex data in stock market prediction [4]

Chapter 3 SYSTEM ANALYSIS AND DESIGN

- Requirement Analysis
- Technical Requirements
 - Jupyter Notebook [12]
 - NumPy [13]
 - Pandas [14]
 - Matplotlib [15]
 - Seaborn [16]
 - Scikit-Learn [17]
 - Keras [18]
 - TensorFlow [19]
 - Pyramid [20] •

Data Analysis I>

- Firm's Historical Stock Data

Sentiment Analysis

Contextual Semantic Search (CSS)

Data Preprocessing

Figure 3.5: Features extracted from the Date attribute # implies Open == Close, so assign neutral sentiment # rare scenario Sentiment = 0
 train set = $df[:\text{math.floor}(.8 * df.\text{shape}[0])]$
 test_set = $df[:\text{math.ceil}(.2 * df.\text{shape}[0])]$ X_train = train_set.drop

Machine Learning Models

Linear Regression • $Y(\text{pred}) = b_e + b_{rx}$ Figure 3.6.1: Relation between Input and

Output using linear regression

Support Vector Machines Figure

3.6.4: Hyperplanes in SVM Figure

3.6.5: Support Vectors Figure

3.6.6: Regularization Values

Figure 3.6.7: Relation between good margin and bad margin 3.6.3 Auto-ARIMA

Chapter 4 PERFORMANCE ANALYSIS

Simple Moving Average

Linear Regression

K Nearest Neighbors

Auto-ARIMA

Chapter 5

CONCLUSION

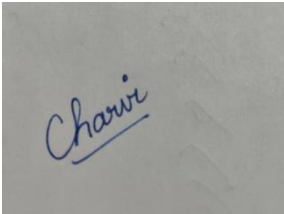
Conclusion

Future Scope

REFERENCES

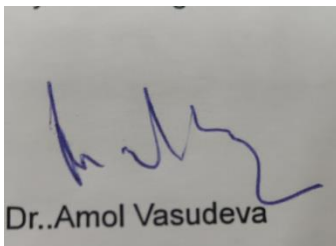
DECLARATION

We hereby declare that the work reported in the B.Tech Project Report entitled '**STOCK FORECASTING USING MACHINE LEARNING MODELS**' submitted at **Jaypee University of Information Technology, Wagnaghat, India** is an authentic record of our work carried out under the supervision of Dr. Amol Vasudeva. We have not submitted this work elsewhere for any other degree or diploma.



Charvi Sharda (181399)

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.



Dr..Amol Vasudeva

Date:

Head of the Department/Project Coordinator

Chapter 1

INTRODUCTION

Introduction

The securities exchange is fundamentally a collection of different purchasers and merchants of stock. A stock (otherwise called shares all the more normally) by and large speaks to possession asserts on business by a specific individual or a gathering of individuals. The endeavor to decide the future estimation of The financial exchange is known as a securities exchange forecast. The conjecture is expected to be strong, exact and compelling. The system should fill in according to the certifiable circumstances and should be proper to genuine settings. The system is furthermore expected to consider every one of the variables that might impact the stock's worth and execution. There are various strategies and techniques for completing the conjecture structure like Fundamental Analysis, Technical Analysis, Machine Learning, Market Mimicry, and Time plan point coordinating. With the progress of the electronic period, the figure has moved into the inventive area. The most observable and promising technique incorporates the use of Artificial Neural Networks, Recurrent Neural Networks, which is basically the use of AI. Artificial intelligence incorporates man-made cognizance which empowers the structure to take in and improve from past experiences without being altered again and again. Standard methodologies for assumption in AI use estimations like normal converse Propagation, generally called Backpropagation botches. As of late, various researchers are using a more prominent measure of outfit learning strategies. It would use minimal expense and time slacks to expect future highs while another framework would use loosened highs to predict future highs. These assumptions were used to

approach stock expenses. The Protection trade esteem figure for a short period of time windows has every one of the reserves of being an erratic system. The stock worth improvement throughout a broad time interval by and large develops an immediate twist. People will buy stocks whose expenses are expected to rise in the near future. The weakness in the monetary trade prevents people from placing assets into stocks. Thus, there is a need to exactly expect the monetary trade which can be used in a veritable circumstance.

The philosophy used to anticipate the protection trade consolidates a period plan close to a particular assessment, AI showing and expecting the variable protection trade. Completely, financial exchange investigation is parceled into two segments - Fundamental Analysis and Technical Analysis. Central Analysis incorporates taking apart the organization's future efficiency in view of its continuous business climate and money-related execution. Specialized Analysis, on the other hand, consolidates examining the outlines and using quantifiable figures to recognize the examples in the protection trade. As you would have estimated, our consideration will be on the specific assessment and portrayal part. We'll use a dataset from Google stock Price test and train.

The datasets of the protections trade gauge model consolidate nuances like the end esteem opening worth, the data what's more, various elements that are supposed to predict the thing factor which is the expense in a given day. The previous model used standard methods for determining things like multivariate examination with an assumption time course of action model. The Protection trade figure beats when it is treated as a backslide issue yet performs well when treated as a portrayal. The point is to structure a model that adds from the market information involving AI procedures and measure the future models in stock worth of development. The Support Vector Machine (SVM) can be used for both request and backslide. It has been seen that SVMs are logically used all together based on issues like our own.

Problem Statement

Cash related exchange measures are fundamentally depicted as trying to pick the stock's worth and present a fantastic suggestion for individuals to be aware and imagine the market

and the stock's costs. It is by and large introduced utilizing the quarterly money-related degree utilizing the dataset. Thus, contingent upon a lone dataset may not be sufficient for the gauge and can give a result which is misguided. Accordingly, we are contemplating the examination of AI with various blends of datasets to expect the market and the stock examples. The issue with assessing the stock expense will stay an issue in the event that an unmatched securities exchange want figure isn't proposed. Foreseeing how the cash related exchange will perform is incredibly badly designed. The improvement in the monetary trade is by and large constrained by the assessments of thousands of examiners. Monetary trade gauging requires an ability to expect the effect of progressing occasions on the examiners. These events can be political events like a declaration by a political trailblazer, a touch of information on a stunt, etc. It can similarly be an overall event like sharp improvements in money related structures and things, etc. All of these events impact corporate pay, which subsequently impacts the sensation of examiners.

It is past the degree of, for all intents and purposes, all examiners to precisely and dependably envision these hyperparameters. All of these components make stock worth the desire irksome. At the point when the right data is assembled, it by then can be used to set up a machine and to create a perceptive result.

worth the irksome desire. At the point when the right data is assembled, it by then can be used to set up a machine and to create a perceptive result.

Objectives

Our goal is to execute different AI calculations like convolutional neural systems, Naive Bayes Classifier, Support Vector Machines, Random Forest Classifier etc for the examination of the monetary trade.

We base this on considering each property in our dataset and applying the as of late referred to dolls for the ideal result. In that capacity, we would have the choice to research which quality generally expects an enormous amount of work in the development and reduction of the stock. Moreover, this endeavor would help us with perceiving that using which estimation we can predict the outcome with the most outrageous precision and exactness. By this, we can see the potential gains and disadvantages of each and every computation used.

Methodology

Machine Learning is the use of man-made mindfulness (PC based information) which enables structures to subsequently take in and update reality by keeping up a fundamental descent of the unequivocal changes possible. ML revolves around the improvement of PC programming which may get to data comparatively as it uses the information it adapted. One way towards information begins with discernments or data, for example, points of view, responsibility, or bearing, with a conclusive objective to look for cases in data and pick better decisions later on subject to the models which we give. The fundamental target exists in allowing the PCs to handle commonly keeping up a key descent from an individual's intercession or help and alter practices in a like manner. With the help of Machine Learning procedures we endeavored to make money related trade desires continuously less troublesome and exact. We proposed a way to execute the AI systems and to anticipate the extraordinary outcomes

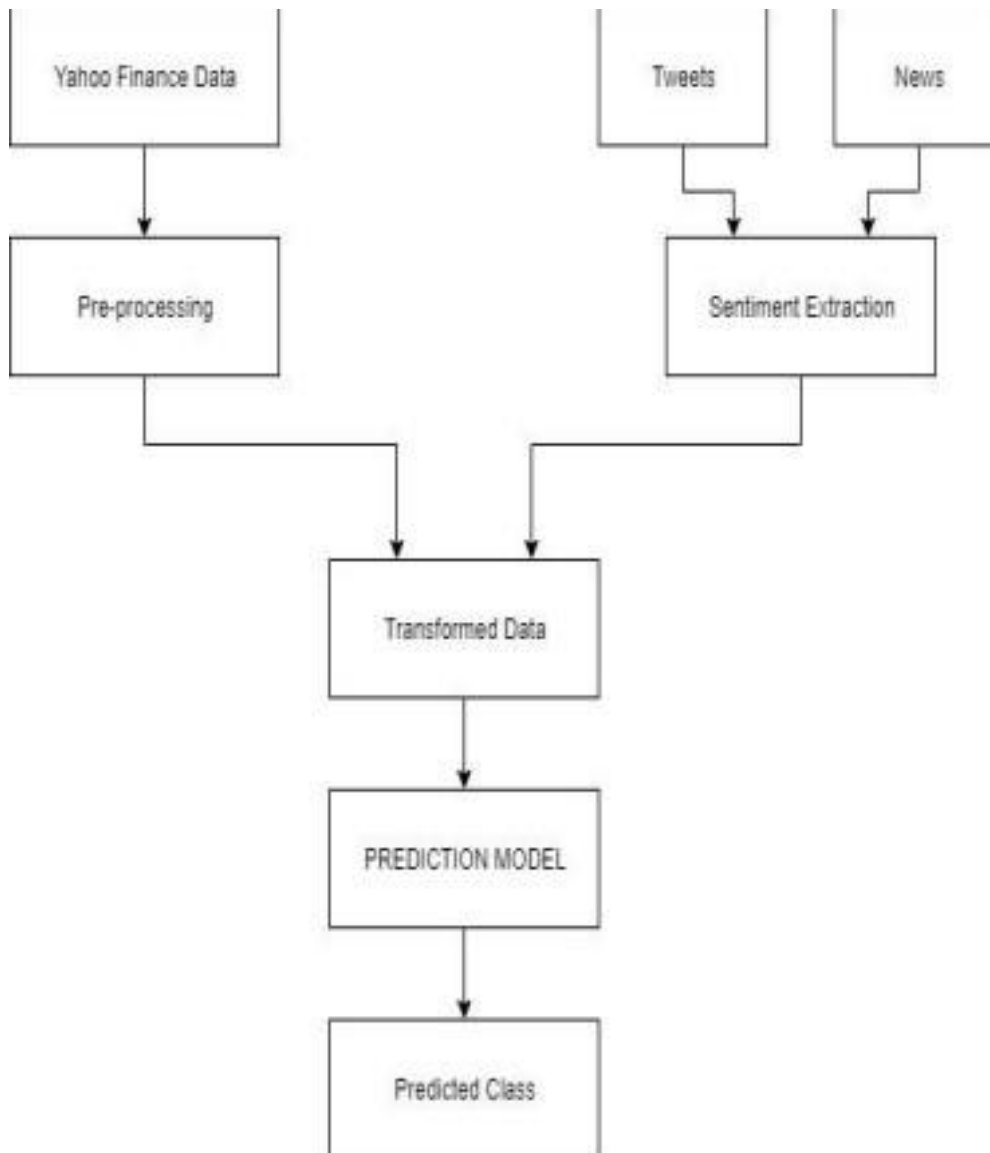


Figure 1.4: Proposed framework

We are working on two types of data -

(a) Firm's historical stock price data (scalar data) (b) Stock news (compositional data)

The two types of data are extracted from different means, the scalar data (the firm's previous

market price) is directly downloaded from Yahoo Finance. The compositional data (NEWS and Tweets) is extracted from the various websites such as Bloomberg, etc. whereas for social media sentiment recent tweets concerning the firm were taken from Twitter using the Twitter API. The Firm's historic data has attributes like Open, High, Low, Close, Adjusted Close, etc. Along with it the date features such as month_of_the_year, month_starting, is_weekend, etc. are also extracted. Tweets consist of numerous abbreviations, emojis and pointless information like pictures and URLs. Thus, tweets are preprocessed to address feelings of openness. For preprocessing of tweets, we utilized three phases of separating: Tokenization, Stopword expulsion and regex coordinating for evacuating unique characters.

Tokenization: Tweets are part into singular words dependent on the space and unessential images like emojis are evacuated. We structure a rundown of individual words for each tweet. Stopword Removal: Words that don't communicate any feeling are called Stopwords. Subsequent to parting with a tweet, words like a, is, the, with and so on are expelled from the rundown of words. Regex Matching for unique character Removal: Regex coordinating in Python is performed to coordinate URLs and is supplanted by the term URL. Regular tweets consist of hashtags(#) and @ tending to different clients. They are additionally supplanted appropriately. For instance, #Apple is supplanted with Apple and @Elon Musk is supplanted with 'client'. Drawn out word indicating exceptional feelings like coooooooool! is supplanted with cool! After these stages the tweets are prepared for assessment.

The preprocessed scalar data and the sentimental data are then combined to make the transformed data so as to make the prediction better using both the different data. On the transformed data, different Machine Learning Models are implemented such as Linear Regression , SVM, Auto ARIMA, LSTM etc. With the help of the Prediction models we can easily find the predicted values of the stock for the next day.

Organization

In Chapter 1, we have talked about the securities exchange, how the financial exchange changes, and furthermore the variables influencing the adjustment in the financial exchange. Likewise, we called attention to the plans that we would use to break down the adjustment in the securities exchange in the report.

In Chapter 2, we have talked about the examination papers that we have alluded to so as to show signs of improvement in comprehension of our task. The papers chiefly center around strategies utilized in AI and different explores in this domain.

In the Chapter 3, we have referred to the potential necessities that are the equipment and programming framework that what language we have used and where are we going to actualize it alongside the libraries required alongside insights regarding the stage utilized.

In Chapter 4, we have examined the calculations in subtleties and furthermore the methodologies used to foresee the result and viability of our outcome. Executions and the consequences of the yields have been talked about.

In Chapter 5, we have given the ends that have been obtained from this investigation and the future extent of this undertaking.

Chapter 2 LITERATURE REVIEW

Technical Analysis & Sentiment Embeddings for Market Trend Prediction [1]

Author: Andrea Picasso, Simone Merello, Yukun Ma, Luca Oneto, Erik Cambria

Publication: Elsevier, June 2019

Summary: The creators in their paper have proposed models to foresee the costs of stock for the main twenty organizations in the NASDAQ100 file by putting together their work with respect to the Efficient Market Hypothesis (EMH) and Adaptive Market Hypothesis (AMH). The stock information was taken from the Google Finance API and the news were recovered from Intrinio API. The information was standardized using Loughran and McDonald's, just like AffectiveSpace Dictionaries. The information was taken care of as a contribution to various characterization models, in particular, Random Forest, Support VectorMachines and a four layer feed-forward Neural Network. It was seen that the NN classifier had the option to group both the positive and negative examples when contrasted with the

SVM and RF classifier because of its better review.

Advantages:

The Loughran & McDonald's and AffectiveSpace Dictionaries are sentiment analysis dictionaries which contain words specially related to Stocks thus, the sentiment extraction is much more accurate.

The use of both EMH & AMH results in a more positive outcome as compared to using them separately

Disadvantages:

The model's performance is predicted on the basis of a trading simulation and not real life situations, thus, the performance metrics might be inaccurate.

Prediction Models for Indian Stock Market [2]

Author: Aparna Nayak, M. M. Manohara Pai, Radhika M. Pai

Publication: Elsevier, 2016

Summary: The Authors have proposed an expectation model in view of the standard of hypotheses with the assumption that set of experiences will in general rehash the same thing. Utilizing this methodology, they can foresee the protection market development or pattern, i.e., whether the cost will be going up or down utilizing past information and virtual entertainment information. The verifiable information comprises authentic costs for an organization got from Yahoo Finance. Likewise, they have proposed two models - a day to day expectation model that considers both the previous information as well as the opinion information to anticipate the chart for the following day, and a forecast model intended to gauge month to month that utilizes just the verifiable information to anticipate the patterns for the following 30 days. The creators have utilized Supervised Learning Algorithms, to be specific, Boosted Decision Tree, Logistic Regression and Support Vector Machines for the two models.

Advantages:

The use of both the historical data as well as social media sentiments is sought to have predicted trends much more accurately than either used separately.

Disadvantages:

The proposed model is only able to predict the trends in the stock price and thus, unable to predict the amount that can be gained by the users.

The sentiment dataset might be sparse, meaning that news for each day was not present thus resulting in inaccurate trend predictions.

NSE Stock Market Prediction Using Deep-Learning Models [3]

Author: Hiransha M, Gopalakrishnan E.A., Vijay Krishna Menon, Soman K.P.

Publication: Elsevier, 2018

Summary: The authors in this paper have proposed different profound learning models at the stock cost expectation on the stock costs for organizations from the National Stock Exchange. The system was prepared with the stock costs of a solitary organization and forecasts were made for different organizations utilizing this model. The creators had prepared four kinds of profound neural systems, to be specific, MLP, RNN, LSTM and CNN. These models were then contrasted and

12

ARIMA model and it was seen that exhibition of NN design is better than that of ARIMA.

Advantages:

CNN was observed to be a better model as it was able to capture the patterns in the provided dataset.

Disadvantages:

2.2.5.1 The advantages of using a hybrid network were not considered for making predictions.

2.4 Aggregating multiple types of complex data in stock market prediction

[4] 2.4.1 Author: Huiwen Wang , Shan Lu , Jichang Zhao.

Publication: Elsevier, 2018

Summary: In this paper, the creators proposed a structure that totals three kinds of information - the exchanging volume (scalar information), intraday return arrangement (practical information) , speculators' feelings from web based life (compositional information) . Through their structure and the observational investigations on the Chinese Stock market they attempted to conjecture whether the market goes up or down at the opening the following day. While the system is model-autonomous, they chiefly investigate the Logistic relapse in this examination. As far as isometric log ratio change, useful head segment and calculated relapse, they build up the estimation methodology of the structure in amassing complex information. Numerical recreation tests show that the proposed structure is compelling. Explicitly it is discovered that the intraday returns sway the accompanying opening of the bearish market and the bullish market.

Advantages:

The increase in volume, and especially types of data in the finance area provides chances to understand the stock market more efficiently and makes the price prediction results better. 2.4.4.2 Since the framework proposed here is model independent, the framework can be combined with any predictive model.

2.4.4.3 Through the combination of the framework with the predictive model, the efficiency of the model is increased.

Disadvantages:

The correlation among the time series is neglected as the observations are treated independently.

Predicting Market Performance with Hybrid Mode [5]

Author: Mehak Usami, Mansoor Ebrahim, Kamran Raza, Syed Hasan

Adil 2.5.2 Publication: IEEE , 2018

Summary: In this paper creators proposed the Hybrid Model to anticipate the presentation of the Karachi Stock Market (KSM). The model involved four sub-models that were completely founded on various AI procedures. Each sub-model utilized 6 information traits including fuel value, ware, outside trade, loan fee, overall population feeling and related NEWS. The various models, for example, Auto-Regressive Integrated Moving Average (ARIMA) and Simple Moving Average (SMA). The Bolster Vector Machine, Radial Basis Function (RBF), Artificial Neural Network's two variations including Single Layer Perceptron and Multi-layer Perceptron were utilized to plan four distinctive sub-models. The outcomes anticipated by all the sub-models were converged in the Hybrid Model. The Hybrid model predicts the market execution based on yield of every four expectation strategies. Utilizing this cross breed model, the expectation was made.

Advantages:

Using the hybrid model was beneficial as it increased the prediction accuracy.

Disadvantages:

The model was giving anomalies for the large data set.

Using Twitter trust network for stock market analysis [6]

Author: Yefeng Ruan , Arjan Durrezi , Lina Alfantoukh

Publication: Elsevier, 2018

Summary: In this paper, the creators utilized the irregular market returns as truth for the trust the executives framework. Hence they confirmed the speculation that the client's notoriety,

worked by trust among them, utilizing our trust board framework, helps in

improving forecasts of irregular stock returns. In light of tweets posted by the clients, they chose eight firms which were the best eight referenced firms in the informational index. Correspondingly, those eight firms' protections trade data was assembled from Yahoo! Accounts. For the Twitter clients, they changed the trust the leaders framework and constructed a client-to-client trust plan. Considering this client-to-client trust association, they decide clients' ability or reputation in a direct way. To check whether Twitter incline information could help with inspecting monetary trade, for each firm, they analyzed Pearson association coefficients between Twitter evaluation valence and the affiliation's abnormal returns. With the help of the association, makers were weighted and isolated by their reputation or power in the whole organization. By taking into account the auto-association property of surprising stock returns, an immediate backslide model is created, in which recent days' odd returns were considered as a control variable. It was seen that by using the trust-organized power-based methodology to weight tweets, the immediate backslide improved .

Advantages:

The suggested framework gave the better prediction results.

Along with the factual data of Yahoo! Finance, people's sentiments are also used to predict the stock price.

Disadvantages

The above prediction is for limited data only, it might be possible that the relation pattern used here can show anomalies for the different time period.

Chapter 3

SYSTEM ANALYSIS AND DESIGN

Requirement Analysis

In this project, EMH & AMH is used to predict the future price and trend in the stock price for a firm using the firm's historical data as well as recent financial news and social media posts concerning the firm in question. So, the dataset required by us is:

The firm's historical stock price data. This data can be downloaded directly from Yahoo Finance [10] and each transaction date consists of Open Price, High Price, Low Price, Close Price, Adjusted Closing Price and Volume Traded for that day.

The news and social media tweets for the day on which stock price is to be predicted. News can be collected from various websites such as Bloomberg, etc. whereas for social media sentiment recent tweets concerning the firm were taken from twitter using the TwitterAPI.[11]

Technical Requirements

Jupyter Notebook [12]

Jupyter Notebook is a kind of web application which is used to make code, perceptions, and so forth utilizing Python. By virtue of the mix of code and content parts, these files are the ideal spot to join an assessment portrayal, and its results, similarly as they can be executed to continuously play out the data examination.

NumPy [13]

NumPy is the critical gathering for real enlistment with Python. It contains despite various things: an unfathomable N-layered bunch object, present day (telecom) limits, and different obliging direct element based math, Fourier change, and optional number limits

.3.2.3 Pandas [14]

Pandas is an open source library for Python which gives a tip top, easy to utilize data

construction and data examination instruments. Pandas has a component called DataFrames which stores the datasets and upholds different procedures on the dataset.

Matplotlib [15]

Matplotlib is a Python 2D plotting library which is utilized to deliver excellent figures and charts. It very well may be utilized to handily create plots, histograms, bar diagrams, scatterplots, and so forth in short codes. Generally, the *pyplot* module is used to plot figures.

Seaborn [16]

Seaborn is a Python information depiction library dependent on matplotlib. It gives a general connection point to drawing in and illuminating evident plans. This library is immovably integrated with pandas data structures also.

Scikit-Learn [17]

Scikit-learn is a collection of simple and efficient open source tools for data mining and data analysis. It contains various modules for data preprocessing, machine learning models, and metrics.

Keras [18]

Keras is an open source brain structure library written in python and runs on tensorflow. It contains various executions of overall utilized brain engineer squares for example layers, inception limits, analyzers, etc thusly simplifying it to go after brain organizations.

TensorFlow [19]

Tensorflow is a free and open source library by Google for dataflow and differential programming over a scope of undertakings. It is an emblematic math library and is utilized for general AI applications.

Pyramid [20]

Pyramid is a python library with the main objective of bringing the Auto-ARIMA implementations to python

Data Analysis

Firm's Historical Stock Data

The data in consideration here is the firm's historical price data which has been taken directly from Yahoo Finance. For reference, here we are working on data for two firms only for now, namely, *Apple (AAPL)* and *Alphabet Inc. (GOOG)*.

The historical stock data has the following attributes:

Open: The starting price at which the stock is traded on a particular

day. *High*: The maximum price of the stock for a particular day.

Low: The minimum price of the stock for a particular day.

Close: The final price at which the stock is traded on a particular day.

Adj. Close: The stock's final value after factoring in things like dividend, stock splits and new stock offerings.

	Open	High	Low	Close	Adj Close	Volume
count	1259.000000	1259.000000	1259.000000	1259.000000	1259.000000	1.259000e+03
mean	145.892192	147.193177	144.612470	145.937164	140.822578	3.747304e+07
std	37.576745	37.949477	37.252473	37.600994	39.420413	1.846219e+07
min	90.000000	91.669998	89.470001	90.339996	85.651482	1.136200e+07
25%	112.474998	113.585003	111.535000	112.555001	105.759479	2.462325e+07
50%	139.850006	141.000000	139.029999	139.990005	134.710419	3.265740e+07
75%	174.940002	175.860001	173.889999	175.000000	171.792351	4.555750e+07
max	230.779999	233.470001	229.779999	232.070007	228.523819	1.622063e+08

Figure 3.3.1: Various metrics of Attributes (AAPL)

	Date	Open	High	Low	Close	Adj Close	Volume
0	2014-09-16	99.800003	101.260002	98.889999	100.860001	92.548836	66908100
1	2014-09-17	101.269997	101.800003	100.589996	101.580002	93.209496	60926500
2	2014-09-18	101.930000	102.349998	101.559998	101.790001	93.402184	37299400
3	2014-09-19	102.290001	102.349998	100.500000	100.959999	92.640594	70902400
4	2014-09-22	101.800003	102.139999	100.580002	101.059998	92.732346	52788400
5	2014-09-23	100.599998	102.940002	100.540001	102.639999	94.182167	63402200
6	2014-09-24	102.160004	102.849998	101.199997	101.750000	93.365494	60171800
7	2014-09-25	100.510002	100.709999	97.720001	97.870003	89.805222	100092000
8	2014-09-26	98.529999	100.750000	98.400002	100.750000	92.447876	62370500
9	2014-09-29	98.650002	100.440002	98.629997	100.110001	91.860641	49766300

Figure 3.3.2: Sample Attribute Values (AAPL)

	Open	High	Low	Close	Adj Close	Volume
count	1260.000000	1260.000000	1260.000000	1260.000000	1260.000000	1.260000e+03
mean	869.453559	876.894638	861.803848	869.624324	869.624324	1.746216e+06
std	226.558635	228.881276	224.592748	226.902369	226.902369	8.609993e+05
min	493.295654	494.618011	486.225067	491.201416	491.201416	5.272000e+05
25%	701.167511	708.322769	693.409973	700.500000	700.500000	1.237500e+06
50%	829.094971	833.565002	826.490021	830.695007	830.695007	1.525350e+06
75%	1079.169952	1091.835022	1067.545044	1079.607452	1079.607452	1.989000e+06
max	1274.000000	1289.270020	1266.295044	1287.579956	1287.579956	1.116490e+07

Figure 3.3.3: Various metrics of Attributes (GOOG)

	Date	Open	High	Low	Close	Adj Close	Volume
0	2014-09-15	571.371277	573.375793	566.654236	571.530884	571.530884	1597500
1	2014-09-16	571.191772	579.907837	571.092041	578.362122	578.362122	1480300
2	2014-09-17	578.421936	585.911377	577.190308	583.168884	583.168884	1692800
3	2014-09-18	585.392822	587.925842	583.398254	587.656616	587.656616	1444500
4	2014-09-19	589.880493	594.846863	587.885986	594.447937	594.447937	3736600
5	2014-09-22	592.194153	592.322754	581.862488	585.761780	585.761780	1689500
6	2014-09-23	585.243225	585.243225	579.409241	579.538879	579.538879	1471400
7	2014-09-24	579.867981	588.015625	578.930542	586.380066	586.380066	1728100
8	2014-09-25	585.941284	586.370117	572.607910	573.485474	573.485474	1925900
9	2014-09-26	574.482788	577.664001	573.086609	575.519897	575.519897	1443600

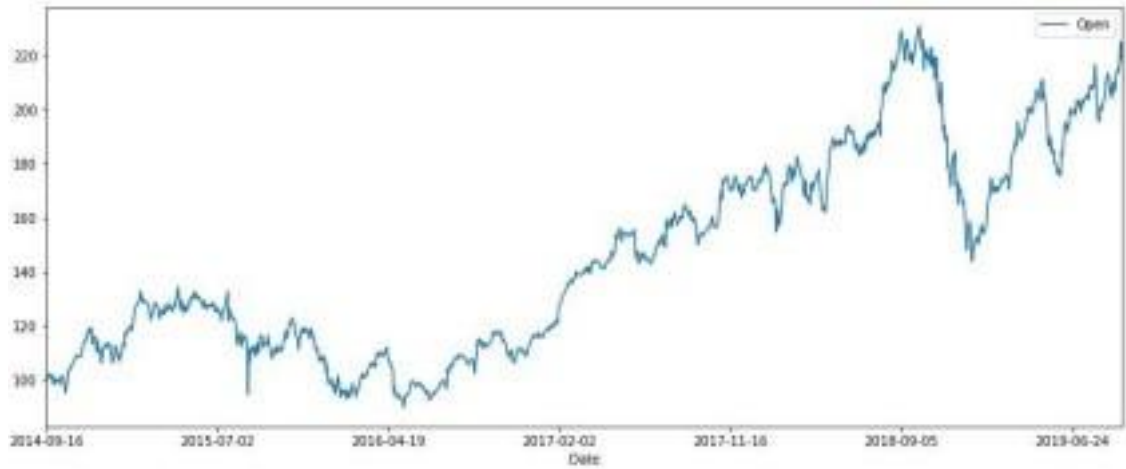
Figure 3.3.4: Sample Attribute Values (GOOG)

Also, the historical price dataset has no null values, that is, there are some values associated with each attribute for every day the stock market was open in the taken time frame.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1259 entries, 0 to 1258
Data columns (total 7 columns):
Date          1259 non-null object
Open          1259 non-null float64
High          1259 non-null float64
Low           1259 non-null float64
Close         1259 non-null float64
Adj Close     1259 non-null float64
Volume        1259 non-null int64
dtypes: float64(5), int64(1), object(1)
memory usage: 68.9+ KB

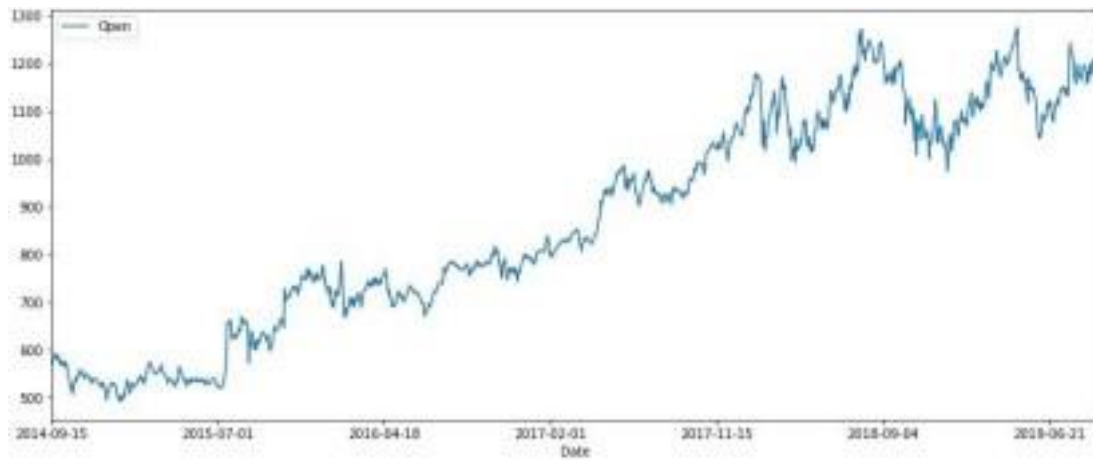
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1260 entries, 0 to 1259
Data columns (total 7 columns):
Date          1260 non-null object
Open          1260 non-null float64
High          1260 non-null float64
Low           1260 non-null float64
Close         1260 non-null float64
Adj Close     1260 non-null float64
Volume        1260 non-null int64
dtypes: float64(5), int64(1), object(1)
memory usage: 69.0+ KB
```

(a) AAPL (b) GOOG Figure 3.3.5: Information about the dataset



(a)

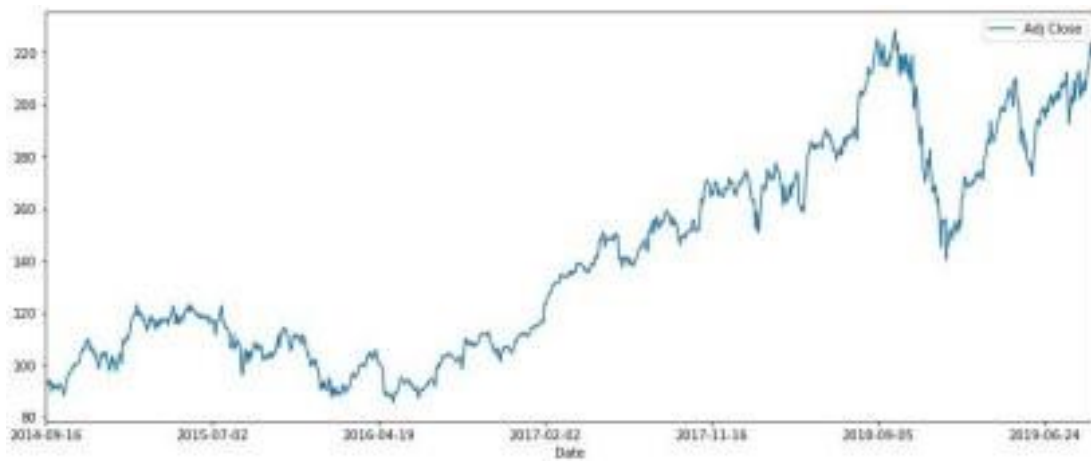
AAPL



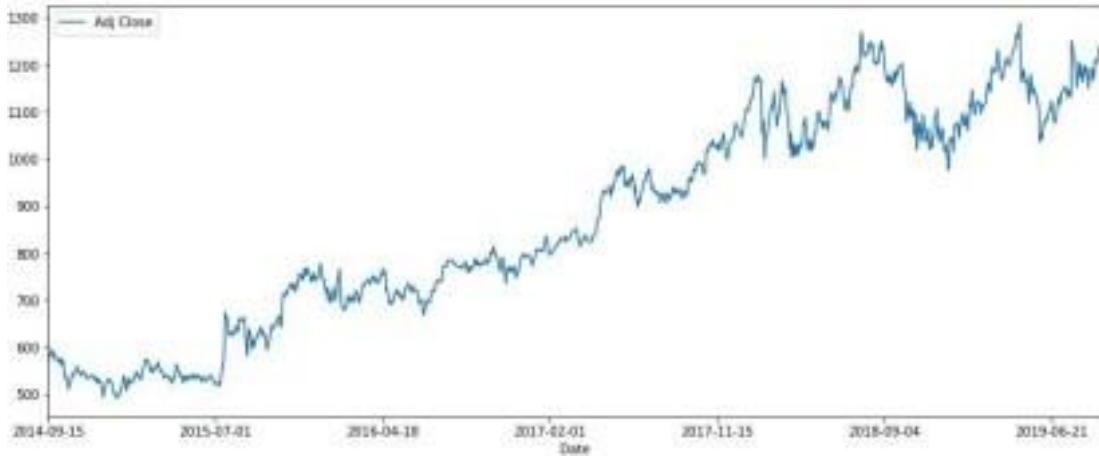
21

(b) GOOG

Figure 3.3.6: Historical View of Opening Prices

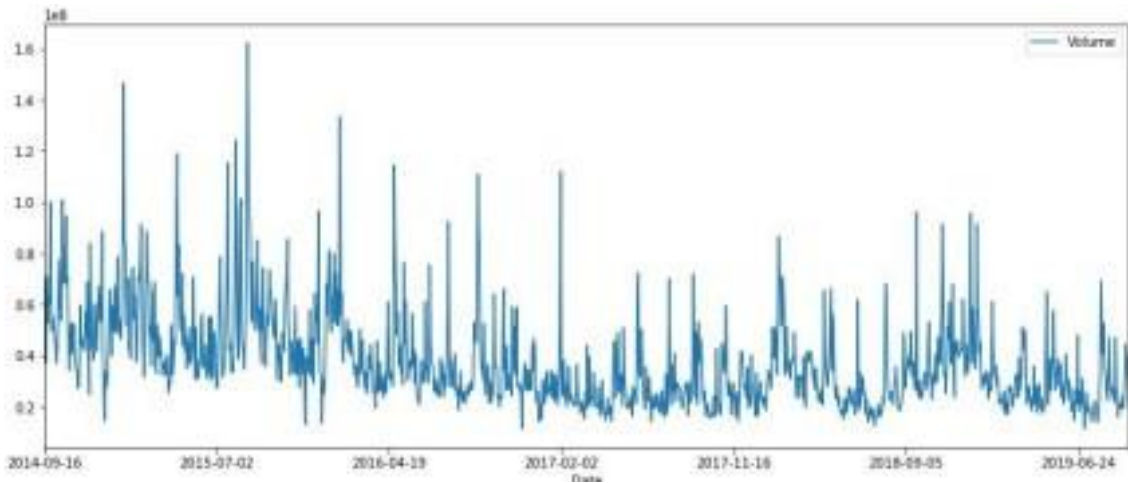


(a) AAPL

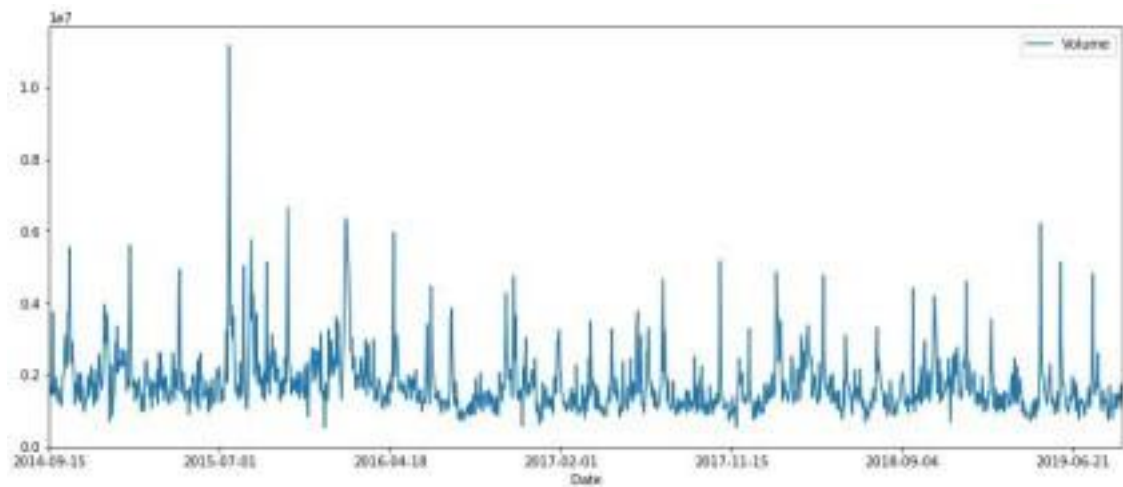


(b) *GOOG*

Figure 3.3.7: Historical View of Adjusted Closing Price

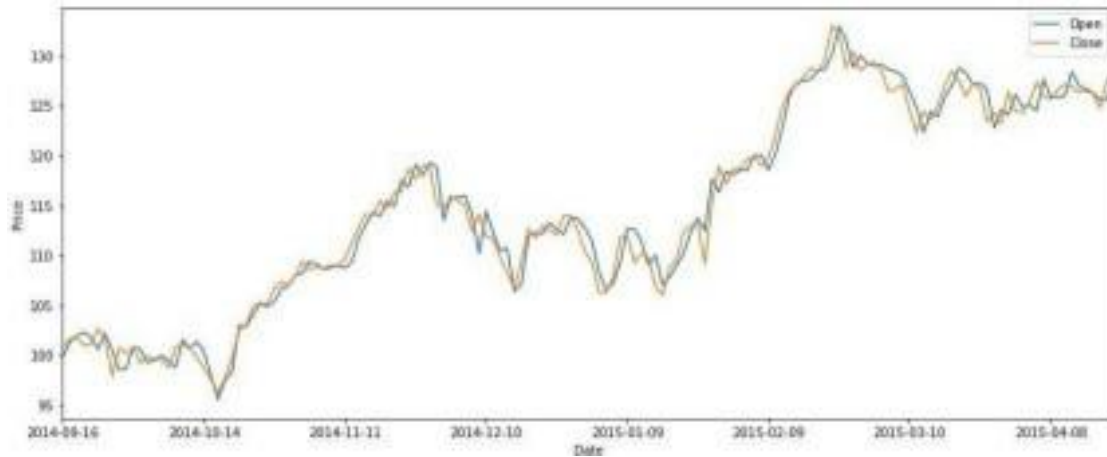


(a) *AAPL*



(b) *GOOG*

Figure 3.3.8: Historical View of Stock Traded Each Day Prices



(a) AAPL

Figure 3.3.9: Historical View of Opening Prices vs Closing Prices

(a) AAPL

(b) GOOG

Figure 3.3.10: Historical View of Difference between Opening & Closing Prices for Each Day

Sentiment Analysis

Sentiment Analysis is the cognizant meaning of a substance which sees and evacuates one-of-a-kind data in source material, and helps a business to get a handle on the social tendency of their image, thing or association while seeing on the web discussions. Regardless, examination of Online life streams is regularly restricted to just basic assessment and check-based estimations. This resembles basically starting to reveal what's underneath and abandoning those high worth bits of data that are holding down to be found. All things considered, what should a brand do to find that low hanging regular item?

With the ongoing advances in profound learning, the capacity of calculations to dissect content has improved impressively.

Intent Analysis

Purpose Analysis ventures up the game by examining the client's goal behind the message and distinguishing whether it is a question, objection, gratefulness and so on.

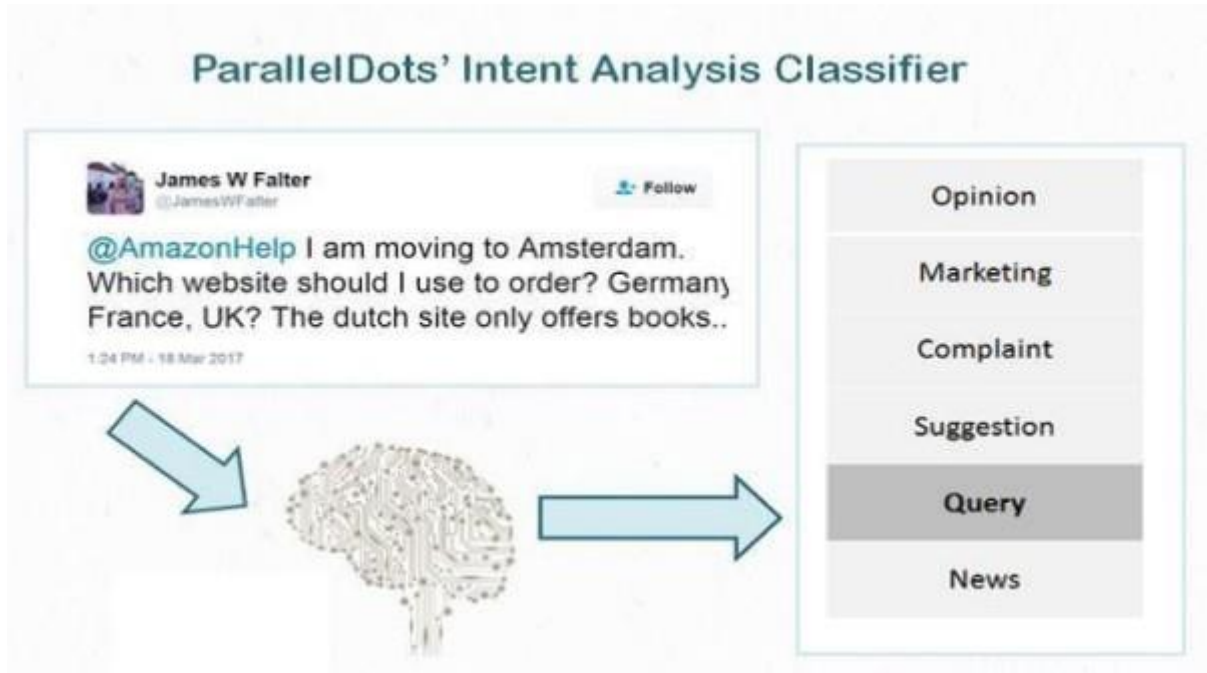


Figure 3.4.1: Intent Analysis

Contextual Semantic Search (CSS)

To determine noteworthy experiences, it is imperative to comprehend what part of the brand a client is examining about. For instance: Amazon would need to isolate messages that are identified with: late conveyances, charging issues, advancement related questions, item audits and so forth. Then again, Starbucks would need to arrange messages dependent on whether they identify with staff conduct, new espresso flavors, cleanliness input, online requests, store name and area and so forth. The way CSS works is that it takes a great many messages and an idea (like Price) as information and channels all the messages that intently coordinate with the given idea.

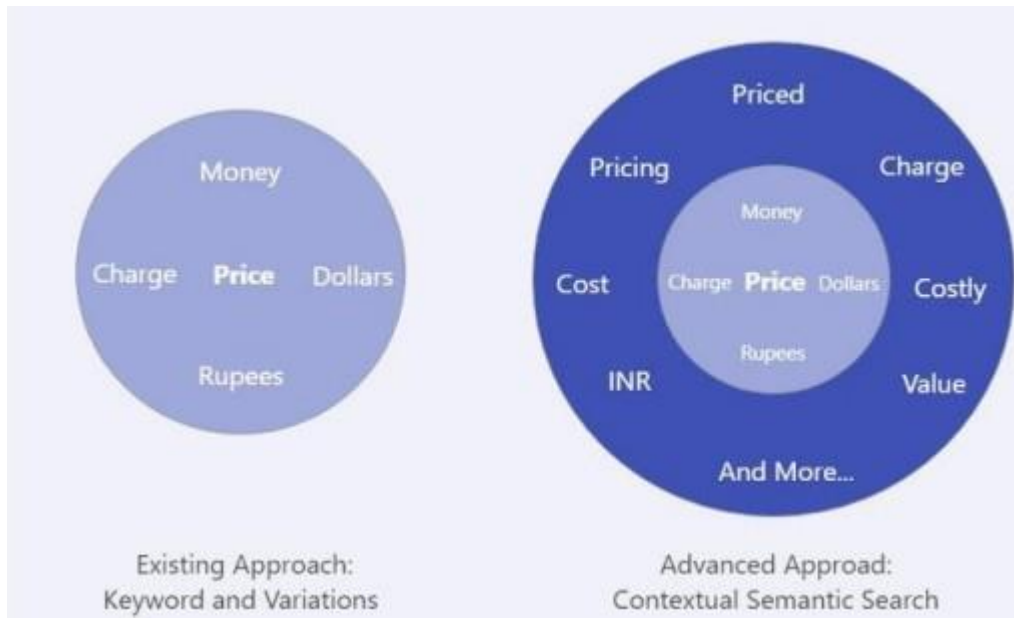


Figure 3.4.2: Difference between the existing approach and the advanced approach

Data Preprocessing

Data preprocessing is a data mining strategy that incorporates changing unrefined data into a sensible setup. Genuine data is routinely insufficient, clashing, and might be lacking in explicit practices or slants, and is most likely going to contain various errors. Information preprocessing is an exhibited methodology for settling such issues. True information may likewise be uproarious, that is, it might contain mistakes or anomalies. In the raw historical stock price dataset, there were various missing values for some dates. For the missing values, the opening price was updated as the closing price of the previous day and the closing prices and adjusted closing prices were updated as the opening prices of the next day. Also, various features were extracted from the Date attribute of the historical stock price dataset. The features extracted are as follows:

Year: Integer value storing the year for that stock price.

Month: Integer value in range [1, 12], storing the month for that stock

price. Day: Integer value storing the day from the date for that stock price.

Week: Integer value in range [1, 52 (or 53)], storing the week number for that date. Day of

Year: Integer value in range [1, 365 (or 366 in case of a leap year)], storing the day number

in that year for that stock price.

Day of Week: Integer values ranging from 0 for Monday to 7 for Sunday representing the day of the week, taken from the date.

Is Month Start: Binary integer value, 1 for denoting month starts and 0 if any other day of the month.

Is Month End: Binary integer value, 1 for denoting month ends and 0 if any other day of the month.

Is Quarter Start: Binary integer value, 1 for denoting quarter starts and 0 if any other day of the quarter.

Is Quarter End: Binary integer value, 1 for denoting quarter ends and 0 if any other day of the quarter.

Is Year Start: Binary integer value, 1 for denoting year starts and 0 if any other day of the year. Is Year End: Binary integer value, 1 for denoting year ends and 0 if any other day of the year.

Figure 3.5: Features extracted from the Date attribute

Likewise, in the twitter news dataset, the tweets comprise of different abbreviations, emojis and other pointless information, for example, pictures, URLs, and so on. In this way, these tweets are pre-processed to speak to the feelings or conclusion of general society. For preprocessing the tweets, different phases of separation were applied as depicted in the Methodology.

Machine Learning Models

Linear Regression

Direct relapse is utilized for finding straight connections among targets and at least one indicator. There are two kinds of straight relapse Simple and Multiple.

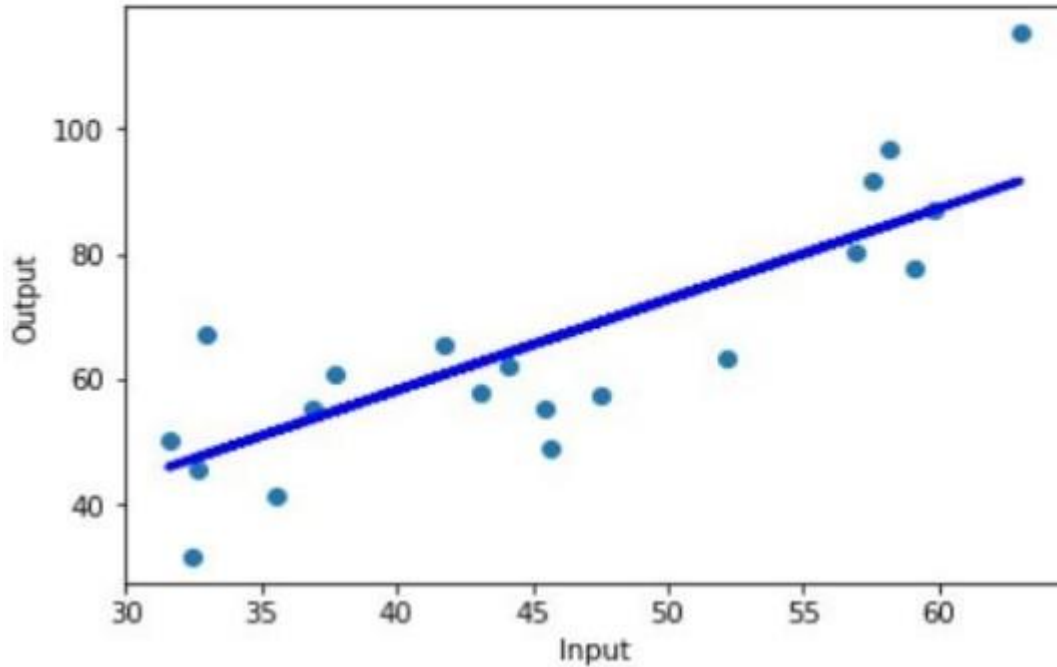
Simple Linear Regression: Essential straight apostatization is valuable for tracking down the relationship between two reliable parts. One is marker or self-administering variable and the other is reaction or ward variable. It searches for genuine connections in any case, not deterministic connections. Relationship between two components should be deterministic on the off chance that one variable can be most certainly granted by the other. For instance, involving temperature in degrees Celsius it is feasible to imagine Fahrenheit unequivocally. Legitimate relationship isn't accurate in picking the relationship between two factors. For instance, affiliation has a few spots in the extent of height and weight.

The center thought is to acquire a line that best fits the information. The best fit line is the line for which the complete expectation blunder is as low as could reasonably be expected. Mistake is the separation between the point and the relapse line. The condition for the straight relapse is given below:

$$Y(\text{pred}) = b_0 + b_1 * x$$

The qualities b_0 and b_1 must be picked with the goal that they limit the mistake. In the event that the entirety of squared blunder is taken as a measurement to assess the model, at that point the objective is to get a line that best decreases the mistake.

$$\text{Error} = \sum_{i=1}^n (\text{actual_output} - \text{predicted_output}) ** 2$$



: Relation between Input and Output using linear regression

Figure

Multiple Linear Regression: Multiple Linear Regression (MLR), likewise referred to just as a different relapse is a measurable strategy that uses a few highly illustrative factors to foresee the result of a reaction variable. The objective of different direct relapse (MLR) is to demonstrate the direct connection between the logical (autonomous) factors and reaction (subordinate) variables.

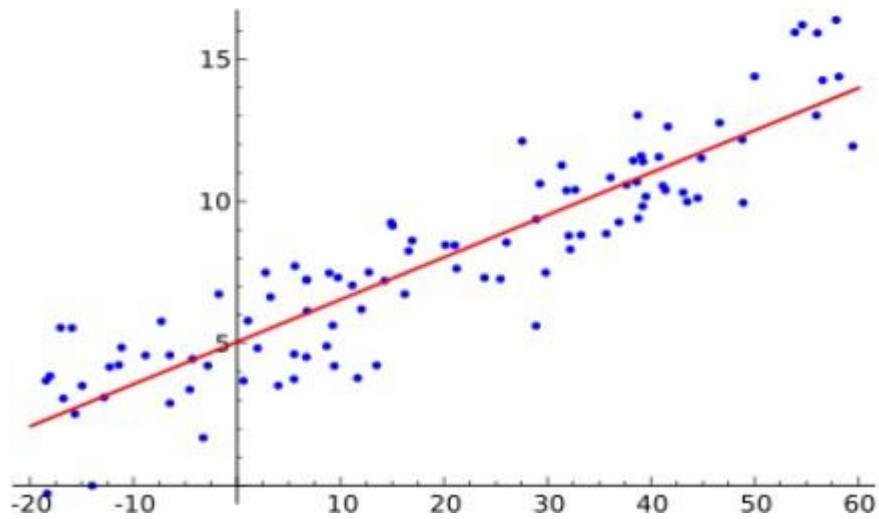


Figure 3.6.2: Multiple Regression

Support Vector Machines

A Support Vector Machine (SVM) is a separating classifier formally depicted by an isolating hyperplane. Around the day's end, given a name for planning data (worked with learning), the assessment gives out an ideal hyperplane which designs new models. In two layered space this hyperplane is a straight line that partitions a plane in two regions where each class lies on one or the other side.

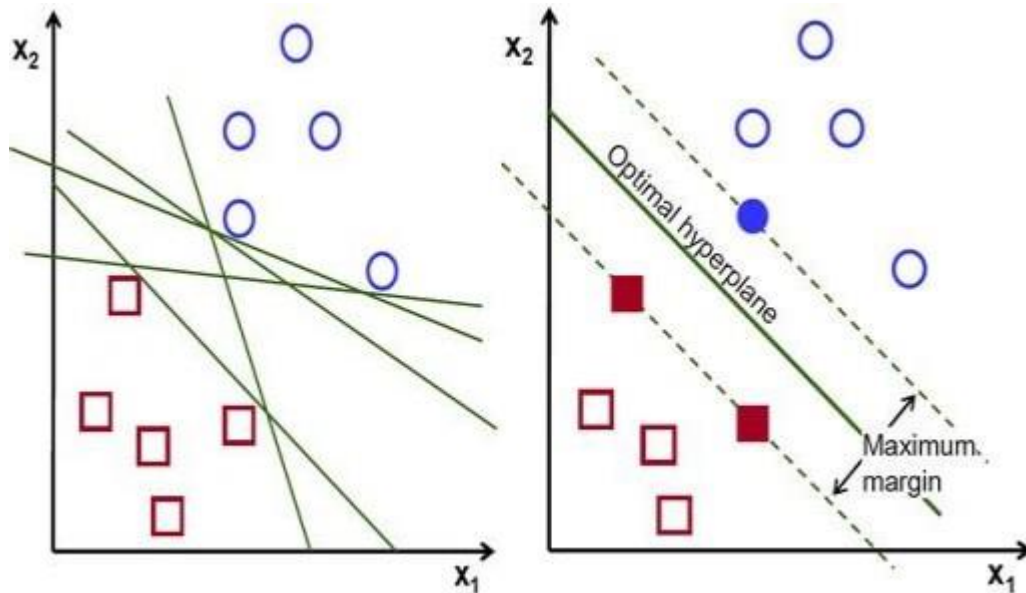


Figure 3.6.3: SVM
Hyperplanes and Support Vectors

Hyperplanes are choice restricts that assist with orchestrating the server ranches. Server ranches falling around one or the other side of the hyperplane can be credited to different classes. Similarly, the piece of the hyperplane depends upon the proportion of features. If the proportion of information features is 2, by then the hyperplane is just a line. In case the proportion of data features is 3, by then the hyperplane propels toward a two-layered plane. It gets hard to imagine when the proportion of features outmaneuvers.

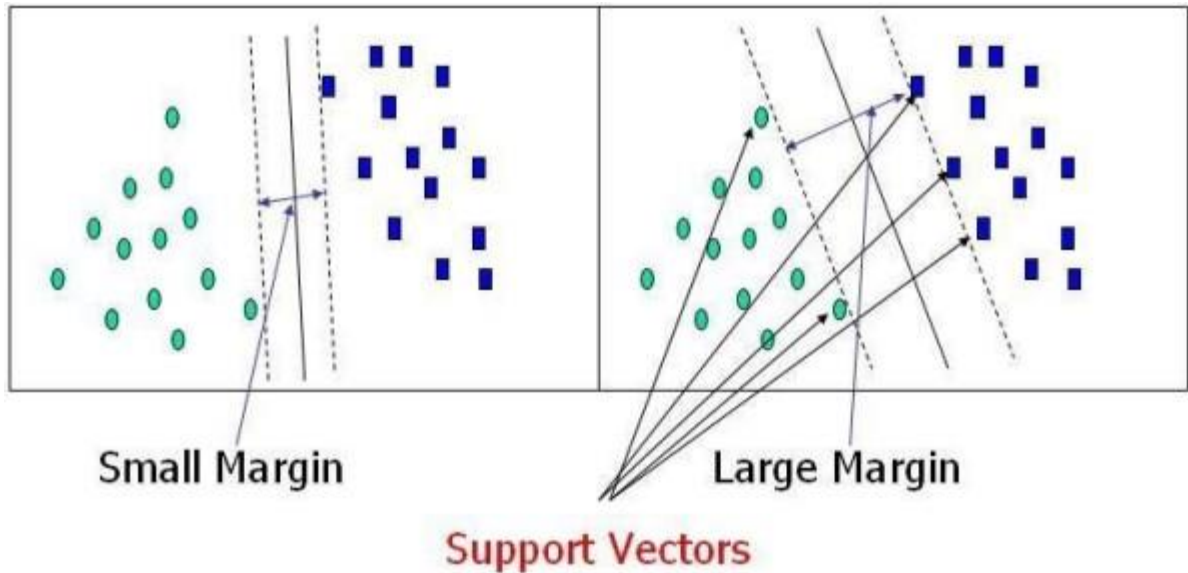


Figure 3.6.5: Support Vectors

Tuning parameters: Kernel, Regularization, Gamma and Margin.

Kernel - The learning of the hyperplane in direct SVM is finished by changing the issue utilizing some straight factor based math. For straight pieces the condition for assumption for another information utilizing the touch thing among information (x) and each help vector (x_i) is settled as follows:

$$f(x) = B(0) + \sum(a_i * (x, x_i))$$

This is a condition that involves calculating the internal consequences of another data vector (x) with all help vectors in getting ready data. The coefficients B_0 and a_i (for every data) must be assessed from the readiness data by the learning estimation.

The polynomial part can be composed as:

$$K(x, x_i) = 1 + \sum(x * x_i)^d \text{ and exponential as } K(x, x_i) = \exp(-\text{gamma} * \sum((x - x_i)^2)).$$

3.5.2.3.2 Regularization - The Regularization boundary tells the SVM improvement the aggregate we need to avoid misclassifying each arranging model. For enormous assessments of C , the streamlining will pick a humbler edge hyperplane. That hyperplane makes an unparalleled showing of getting all the arranging places assembled reasonably. Then again, a little assessment of C will make the streamlining expert search for a more prominent

edge-detaching hyperplane, whether that hyperplane misclassified more center interests.

The photos below (same as picture 1 and picture 2 in portion 2) are two different regularization boundaries. Left one has some misclassification as a result of lower regularization. Higher worth prompts results like the right one.

An edge is a parcel of a line to the closest class. A good advantage is one where this section is more noteworthy for both the classes. Pictures under obliterate visual occasions of good and dreadful edge. A typical edge permits the fixations to be in their particular classes without crossing the point to different classes.

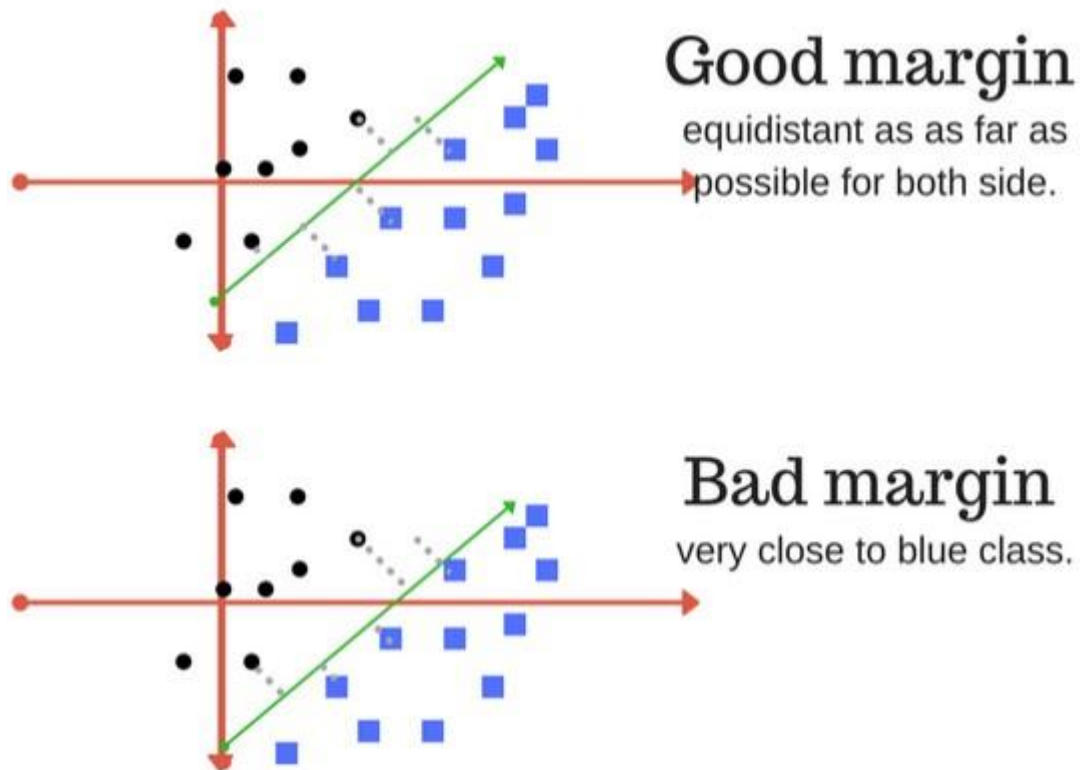


Figure 3.6.7: Relation between good margin and bad margin

3.6.3 K Nearest Neighbors - The k-closest neighbors calculation (k-NN) is a non parametric procedure utilized for social occasion and losing the faith. In the two cases, the data contains the k closest preparing models in the section space The yield relies on whether k NN is utilized for demand or break faith:

In k-NN gathering, the yield is a class interest. A thing is mentioned by a more noteworthy number of votes from its neighbors, with the article being entrusted to the class regularly utilized among its k closest neighbors (k is a positive whole number, normally little). On the off chance that $k = 1$, the thing is simply given out to the class of that solitary closest neighbor. In a k-NN lose the faith, the yield is the property of the thing. This worth is the customary of the assessments of k closest neighbors. k-NN is a sort of occasion based learning or languid recognizing, where the cutoff points are basically approximated locally and all calculation is yielded until depiction. Both for arranging and breaking faith, a steady technique can be to relegate weights to the obligations of the neighbors, with the objective that the nearer neighbors offer more to the common than the more removed ones. For instance, an ordinary weighting plan incorporates providing each neighbor with a load of $1/d$, where d is the division to the neighbor. The neighbors are taken from an incredible number of articles for which the class (for k-NN gathering) or the property assessment (for k-NN lose the faith) is known. This can be considered the status set for the figuring, in any case no express arranging steps are required. A quality of the k-NN count is that it gets precarious with the nearby figure of accessible information. The information strategy is fixed, which suggests that the mean and change shouldn't move with time. A strategy can be made fixed by utilizing log changes or separating the arrangement.

The information given as data should be a univariate game-plan, since arima utilizes the previous qualities to predict the future qualities.

Chapter 4

PERFORMANCE ANALYSIS

We will start with the simplest model - a simple moving average model. This will serve as a basis of reference for all the other models. The various models that were implemented are:

Simple Moving Average

Linear Regression

K Nearest Neighbors

Simple Moving Average

Simple moving average is the simplest of all models. All it does is take the average of the latest set of k values from the date in consideration and then assigns the calculated average as that date's prediction. This model is quite inaccurate and serves as a basis of reference for all other models.

Date	Predictions	Adj Close
2017-12-28	145.094777	71.285904
2017-12-29	145.232942	72.121513
2018-01-02	145.369392	71.107201
2018-01-03	145.501385	69.545288
2018-01-04	145.629751	69.924515

Table 4.1.1: Simple Moving Average Predictions & Actual Closing Prices

Root Mean Squared Error: 68.8011882284805
Mean Absolute Error: 67.91652761640805
Explained Variance Score: 0.237357167661857
Mean Squared Error: 4733.603501650803
R2 score: -28.84774907174063

Table 4.1.1: Simple Moving Average Error Metrics

From the above figure, we can clearly see that the simple moving average model has a high RMSE value thus making it highly inefficient.

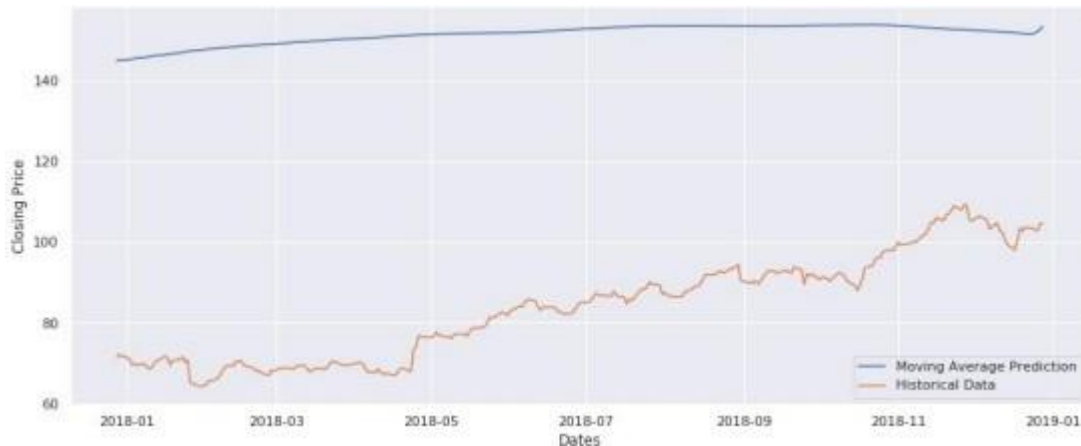


Figure 4.1.1 Simple Moving Average Closing Price vs Average Dates

Linear Regression

Next, we experimented by applying a Linear Regression model on our dataset and the following results were obtained.

Date	Predictions	Adj Close
2017-12-28	72.747998	71.285904
2017-12-29	70.931917	72.121513
2018-01-02	63.938697	71.107201
2018-01-03	64.517062	69.545288
2018-01-04	65.035337	69.924515

Table 4.2.1: Linear Regression Predictions & Actual Closing Prices

Root Mean Squared Error: 9.067108327895989
 Mean Absolute Error: 7.414512378325771
 Explained Variance Score: 0.4880989490443489
 Mean Squared Error: 82.21245342980079
 R2 score: 0.4816091631483087

Table 4.2.2: Linear Regression Error Metrics



Figure 4.2.1 Linear Regression Closing Price vs Average Dates

From the above figure and error values, it can be clearly said that Linear Regression performs much better than simple moving averages and hence can be a model of choice.

K Nearest Neighbors

After Linear Regression, the next model of interest was K Nearest Neighbors and its implementation gave the following results.

Date	Predictions	Adj Close
2017-12-28	97.282794	71.285904
2017-12-29	92.568584	72.121513
2018-01-02	100.729163	71.107201
2018-01-03	91.003315	69.545288
2018-01-04	84.842712	69.924515

Table 4.3.1: KNN Predictions & Actual Closing Prices

Root Mean Squared Error: 18.230457636926396
Mean Absolute Error: 15.386477406525573
Explained Variance Score: 0.05575564944684264
Mean Squared Error: 332.34958565176794
R2 score: -1.095631168341722

Table 4.3.2: KNN Error Metrics



Figure 4.3.1 KNN Closing Price vs Average Dates

Here, it can be seen that KNN performed significantly better than the simple moving average but has slightly more RMSE value than the Linear Regression model.

Chapter 5 CONCLUSION

Conclusion

In the past couple of years, it has been seen that most people are placing assets into the protection trade to acquire cash easily. At the same time monetary experts have a high chance of losing all the money contributed. So a convincing and insightful model is required for the region to see the value in future market plans. There exists different canny models that discuss the plan of the market whether it is up or down, yet they neglect to give accurate outcomes. An undertaking has been made to gather a viable farsighted model of money-related trade where the example for the next day is envisioned. By taking into account the different models like perpetual up/down, volume exchanged every day and besides combining the hypotheses of the affiliation, a model has been constructed and endeavored with various cash-related exchange information accessible via open source. On considering the dataset, it was seen that the Linear Regression model has the least blunder values. Subsequently, it may very well be inferred that, on the considered dataset, Linear Regression performed better compared to the Moving Average, KNN .

	Moving Average	Linear Regression	K Nearest Neighbors	
RMSE	68.80	9.06	18.23	
MAE	67.91	7.41	15.38	
MSE	4733.60	82.21	332.34	
R2	-28.84	0.48	-1.09	

Table 5.1.1: Error Metrics for various Models

The dataset which was been considered for opinion investigation may be deficient which infers we probably won't have news/tweet for a particular association for quite a while. In such cases Principle fragment examination with various components can be applied. The impact of intra-day esteem advancement at the next day's stock expense can be considered to work on the accuracy.

Future Scope

The models that were used were not optimized, therefore, optimizing the model parameters can help in better fitting the model to the dataset and accuracy can be improved. Also, the data being sparse can also be a cause for some discrepancies.

Finally, various other models such as SVM, LSTM and Artificial Neural Networks can be applied to the dataset as they were also found to be producing better results on time series data.

REFERENCES

- [1] Andrea Picasso, Simone Merello, Yukun Ma, Technical Analysis & Sentiment Embeddings for Market Trend Prediction, Expert Systems with Applications, Elsevier, Volume 35, 2019, <https://doi.org/10.1016/j.eswa.2019.06.014> , pp.60-70.
- [2] Aparna Nayak, Prediction Models for Indian Stock Market, Procedia Computer Science, Elsevier, Volume 89, 2016, <https://doi.org/10.1016/j.procs.2016.06.096> , pp. 442-450. [3] Hiransha M, NSE Stock Market Prediction Using Deep-Learning Models, Procedia Computer Science, Volume 132, 2018, <https://doi.org/10.1016/j.procs.2018.05.050>, pp.1351-1362.
- [4] Huiwen Wang, Shan Lu, Aggregating multiple types of complex data in stock market prediction, Knowledge-Based Systems, Elsevier, <https://doi.org/10.1016/j.knosys.2018.10.035> , pp. 193-204.
- [5] Yefeng Ruan,, Using Twitter trust network for stock market analysis, Knowledge Based Systems, Elsevier, vol 125, April 2018, <https://doi.org/10.1016/j.knosys.2018.01.016>, pp. 207- 218.
- [6] Shashank Gupta, Sentiment Analysis: Concept, Analysis and Applications, <https://towardsdatascience.com/sentiment-analysis-concept-analysis-and-applications-6c94d6f58c17>
- [7] Saishruthi Swaminathan, Linear Regression – A Detailed View, Medium, February 2018 <https://towardsdatascience.com/linear-regression-detailed-view-ea73175f6e86> Yahoo Finance, <https://finance.yahoo.com/>
Twitter API, <https://developer.twitter.com/en/docs>
Jupyter Notebook, www.jupyter.org
NumPy, <https://numpy.org/devdocs/user/quickstart.html>
Pandas, <https://pandas.pydata.org/pandas-docs/stable/>
Matplotlib, <https://matplotlib.org/3.1.1/users/index.html>
Seaborn, <https://seaborn.pydata.org/tutorial.html> SciKit Learn, <https://scikit-learn.org/stable/documentation.html> Keras, <https://keras.io/getting-started/sequential-model-guide/> Tensorflow, <https://www.tensorflow.org/tutorials> Pyramid, <https://pypi.org/project/pyramid-arma/>

Plag Report