AUTOMATED ALGORITHMIC STOCK TRADING

ARUN PRASATH SIVA THANU PILLAI

Final Thesis Report

FEBRUARY 2020

Acknowledgement

I would like to express my gratitude to all those who have guided us in the completion of the project. I record my heart full of thanks and gratitude to my mentor for providing me an opportunity to carry this project, along with purposeful guidance.

I am very grateful to my LJMU faculty member Dr. Manoj Kumar, for being a great guide and teacher and Upgrad learning platform and the support staffs including student mentors. LJMU faculty member has provided immense support, constant encouragement and exemplary guidance at every stage of the project. His expertise in the areas was very insightful and his technical guidance was very thoughtful which lead me to gain more deeper knowledge of the research areas. He has always emphasized in thoroughness and clarity in our approach and we are thankful to him, for helping to put the project in perspective.

Last but not the least, special thanks to my wife Subathra Devi for her moral support extended to me throughout the duration of the project work.

Table of Figures

List of Tables

List of Abbreviations

Abbreviations	Expansion
LSTM	Long Short Term Memory
BP	Back Propagation
SOFNN	Self-Organizing Fuzzy Neural Networks
EMH	Efficient Market Hypothesis
MA	Moving Average
EMA	Exponential Moving Average
ROC	Rate of Change
RSI	Relative Strength Index
RNN	Recurring Neural Networks
BoW	Bag of Words
TD-IDF	Term Frequency-Inverse Document Frequency
SA	Sentiment Analysis
NLP	Natural Language Processing
ANN	Artificial Neural Networks
MLP	Multi Layer Network
AMC	Asset Management Company
AR	Auto Regressive method
MA	Moving Average
ARMA	Auto Regressive Moving Average
ARIMA	Auto Regressive Integrated Moving Average
SDA	Stacked Denoising Auto Encoder
P/B	Price-to-Book Ratio
BVPS	Book Value Per Share
EPS	Earnings Per Share
PEG	Price/Earnings to Growth ratio
P/E	Profits to Earnings Ratio
RoE	Return on Equity
RSI	Relative Strength Index
CAPM	Capital Asset Pricing Model
CNN	Convolutional Neural Networks

SMA	Simple Moving Average
NLP	Natural Language Processing
LMS	Lowest Middle Square
CEFLANN	Computationally Efficient Functional Connection
FRPCA	Fuzzy Robust Principal Component Analysis
KPCA	Kernel Based Principal Component Analysis
PCA	Principal Component Analysis
EPCNN	Evolving Partially Connected Neural Networks
NN	Neural Networks
AFSA	Artificial Fish Swarm algorithm
RBFNN	Radial Basis Neural Network
ABC-RNN	Artificial Bee Colony-Recurring Neural Networks
GRU	Gated Recurrent Units
ILDA	Interdependent Latent Dirichlet Allocation
NSE	National Stock Exchange
NYSE	New York Stock Exchange
SENSEX	S&P BSE Sensex index
BPNN	Back Propagation Neural Networks
KNN	K-Nearest Neighbors
RLS	Recursive Least Square
NAV	Net Asset Value
SVM	Support Vector Machine
VWAP	Volume Weighted Average Price
DOW	Day of Week
RMSE	Root Mean Square

Abstract

The prediction of the share market is of great importance in the recent times in order to help in maximizing the profit of stock purchase while keeping the risk at low. With latest advancement in technologies, the opportunity to gain a steady fortune from the share market is increased, which also helps experts to find out the most critical indicators to make a better prediction. Machine learning has many applications, one of which is to forecast time series.

Unfortunately predicting the stock market is most challenging task due to the high uncertainties and volatility nature of stock market. Stock prices are affected by many other factors other than stock data such as economic, geo-political issues such as trade war or cross-border conflict. Volatility of the stock market is high, or the fluctuation is high because of the above-mentioned factors which in-turn impacts the outcome of the prediction. The factors which impacts the prediction outcome are usually categorized as physical factors (vs) physiological, rational and irrational behavior etc., All these factors combined together makes predicting the share prices with high accuracy becomes very difficult.

Several technical analysis tools have been used by data and finance analysts. The time series analysis is used by most of them while making the predictions. Nevertheless, these methods cannot be trusted fully, so there is a necessity to provide the supportive method for stock market prediction. These tools and techniques need lot of technical details and news flows for individual stock. Even after gathering lot of technical details and analyzing several data points, they still would not give reliable predictions and thereby, causing more confusions among stock traders especially beginners.

Artificial Neural Network (ANN) was found to be the most practical consideration. Neural network models have the features and customizable parameters which makes possible to have wide number of features along with the cross-validation sets. The analysis of historical stock data sets and extracting certain trends would be crucial in figuring out the right prediction methodology to forecast the stock value for a given timeframe. RNN will be fed with a pre-processed historical stock data which will be getting trained based on time series forecasting model. Once training is completed, using neural network layer predictions for next trading day(s) will be done. Based on these predictions a potential Buys and Sells for given stock can be generated for a swing trade.

The motivation for this project arises from the fact that, investors are constantly looking for an advanced model that is accurate and adaptable to market fluctuations. This project is carried out to determine a successful strategy for predicting stock prices based on past financial time series data and also based on domain-specific financial data along with sentimental analysis. The first model discussed here uses the “New York Stock Exchange” on Kaggle and an LSTM architecture.

Table of Contents

1 Introduction

In the recent years, artificial neural networks (ANNs) have become predominant in machine learning predictions (Makeshwar et al. 2010) [1]. Historically, they are also examples of directions in which functional challenges like estimation, grouping, clustering, optimization, etc. can be addressed and resolved (Hertz et al. 1991) [2]. The neural multilayer network (MLP) is the common feed-forward feed network. It was widely used in the estimation of time sequences. The primary objective of this research is to use dynamic neural networks to predict and classify time series details.

The prediction method is used to identify potential values or occurrences depending on certain past and present data information such as temperature prediction, stock levels prediction, earthquake forecast, publicity and revenue forecast. Artificial neural networks are well known for optimal performance and can be used as a prediction tool. The Classification techniques that classify a collection of data into groups of items that share similar actions into small subsets of training sets. Kohavi (1995) [3] describes the classifier algorithm which has a feature capable of mapping data to the label type. Classification approaches were deemed to be the most effective decision-making tools and are commonly used for data processing. Typically, time series are considered as spontaneous behaviour pattern which has certain trends associated with it, while taking into account various internal and external factors affecting such time series.

Various research methodologies were used to study the behavioral aspect of the time series prediction in terms of accuracy on the basis of the data collection. Nevertheless, the time-series are so complicated that no specifics on the device are known to produce these time-series, such even conventional approaches cannot solve such problems.

1.1 Problem Statement

- Analyze various machine learning techniques used in the prediction of stock prices or stock market index.
- To understand and apply Algorithmic Trading strategies and generate good enough returns.

1.2 The Fuss about Predictability

The predictability of stock market is a highly controversial topic over many years. Bachelier (1900), Cootner (1964) and Fama(1965) [5] have observed that, the stock market nature is of independent of past and present information and hence unpredictable, thus proving the efficient-market hypothesis. People who trade and make investment in stock market directly by purchasing shares or indirectly by making investments through Mutual Funds AMCs have a very staunch desire to make large money. As the technology is advancing, the opportunity to gain a steady fortune from the share market is increased.

The Stock market is highly volatile because of several factors which can be of physical or psychological as well. All these factors combined together makes predicting the share prices with high accuracy becomes very difficult.

Several technical analysis tools have been used by data and finance analysts. Even after gathering lot of technical details and analyzing several data points, they still would not give reliable predictions. The analysis of historical stock data sets and also by extracting certain trends would help to predict the future value of the stock. Based on these predictions a potential *Buys* and *Sells* for given stock can be generated for a potential swing trade.

1.3 'The Efficient Market Hypothesis"

The efficient market hypothesis postulates that stock prices are a function of information and rational expectations, and that newly revealed information about a company's prospects is almost immediately reflected in the current stock price. As per "Efficient Market Hypothesis (EMH)"[3], the market prices are driven by new information and follow a random walk pattern. This hypothesis is widely accepted by the research community as a central paradigm governing the markets in general, several people have attempted to extract patterns in the way stock markets behave and respond to external stimuli.

Burton Malkiel, in his seminal 1973 work "A Random Walk Down Wall Street" stated, "a blindfolded monkey throwing darts at a newspaper's financial pages could select a portfolio that would do just as well as one carefully selected by experts" or in simpler terms stock prices prediction by looking at price history is not feasible.

Malkiel claimed that variations in stock prices correspond to a statistical process called a "random walk" , which states that each day's deviations from the central value are completely random and unpredictable.

1.4 An Art of Stock Market Predictions

A supervised learning model is to predict stock movement direction can combined along with the model based on technical analysis and sentimental analysis from news and social media. Technically two categories of predictions are available

- **Econometric models:** This includes classical econometric models for predicting the future stock prices. Some of the common methods are the auto-regressive method (AR), the moving average model (MA), the auto-regressive moving average model (ARMA), and the auto-regressive integrated moving average (ARIMA).

- **Soft computing-based models:** The term soft computing covers artificial intelligence. These techniques include artificial neural networks (ANN), fuzzy logic (FL), support vector machines (SVM) and others. Deep Neural to extract abstract features from the data have been developed.

1.5 Stock Prediction Methodologies:

1.5.1 Fundamental Analysis:

This approach, carried out by fundamental analysts, is not about individual stock, but about the business. The analysts use various information about company's business model , revenue model and their balance sheet and year-on-year turn-over, company's previous results, the income projection, etc., to decide to buy/sell that stock.

1.5.2 Technical Analysis:

The technical analysts are determining market prices on the basis of the past market trends (using time-series analyses). Using hybrid prediction models, most efficient forecasting or recommendation model can be built/developed.

1.5.2.1 Indicator Functions: Indicators functions helps in identifying technical analysis

i. **Moving Average (MA):** The average of the past n values till today.

ii. **Exponential Moving Average (EMA):** Gives more weightage to the most recent values while not discarding the older observation entirely.

iii. **Rate of Change (ROC):** The ratio of the current price to the price n quotes.

The technical analysis charts below [6] show how the EMA models the actual Stock Price.

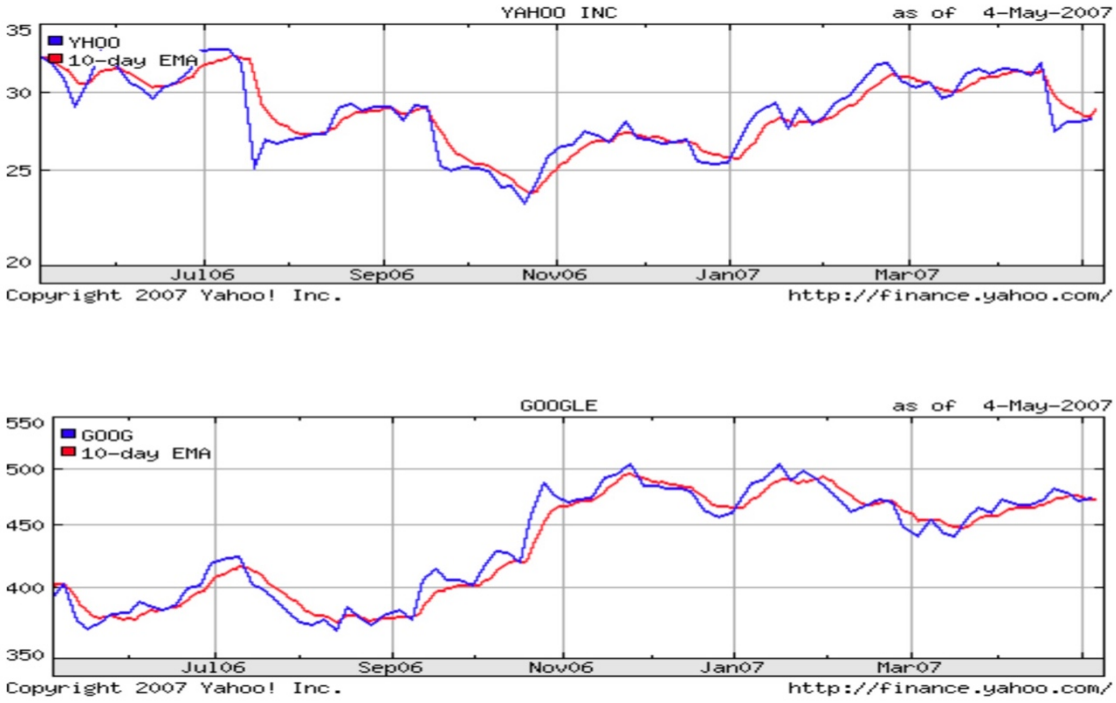


Figure 1 EMA Yahoo and Google chart

1.5.3 Sentimental Analysis

Major challenge in consuming sentimental data for prediction stock market is due to the nature of textual format off sentimental data instead of numerical format which makes prediction process difficult. The solution to this problem can be approached by combing time series data along with mining of text documents to extract the meaningful information which includes sentiments and also news data about the stock or company. It is a technique of analyzing the sentiment of a given set of input generally in the form of text.

Rui Ren, Desheng Dash & Tianxiang Liu (2018) mention in their paper that, the twitter data is split into sentences which will help in getting better polarization instead of a collective document of a given day. As per the literature work, the comparison between parsing the tweets

individually (have better determinant value) and tweet documents as a whole. The polarity of each tweet in the entire day is calculated as per below formula.

$$\frac{{}_{\Sigma}p_i - \Sigma n_i}{\Sigma t_i}$$

Figure 2 Polarity of Tweet - Formula

where pi is the ith positive tweet, ni is the ith negative tweet and ti is the ith tweet.

1.5.4 Deep Learning

- It is a subset of ML based on artificial neural.
- **Two Phase Approach :**
  1. **Training phase :**
     - Training the model on existing data with mean-squared error as the error metric.
     - Back propagation of the mean squared error multiplied by a stochastic gradient used to rectify the errors.
  2. **Prediction phase**

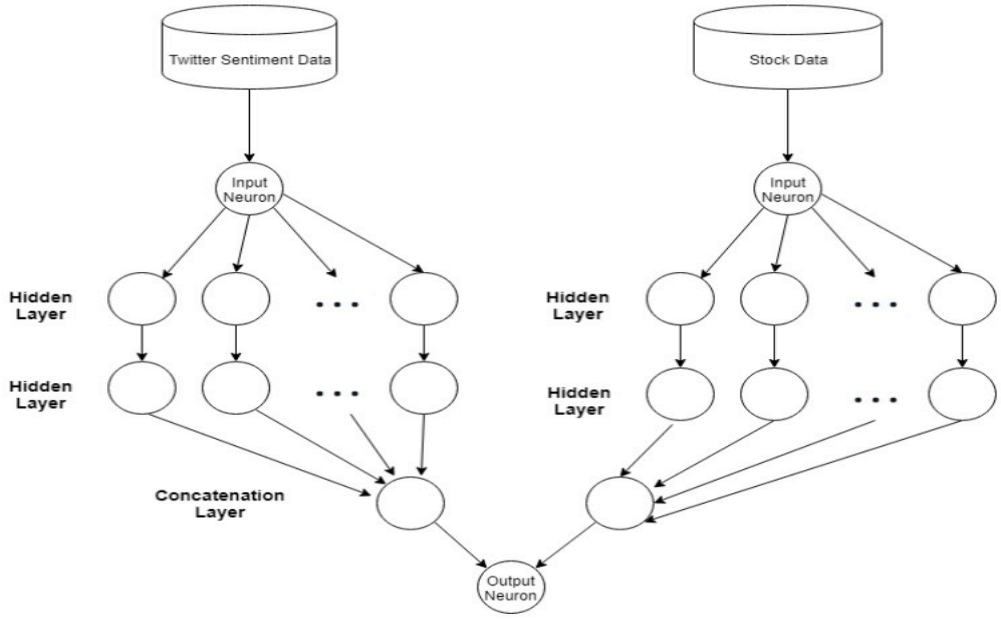


Figure 3 Twitter Data and Stock Data in Deep Learning Model

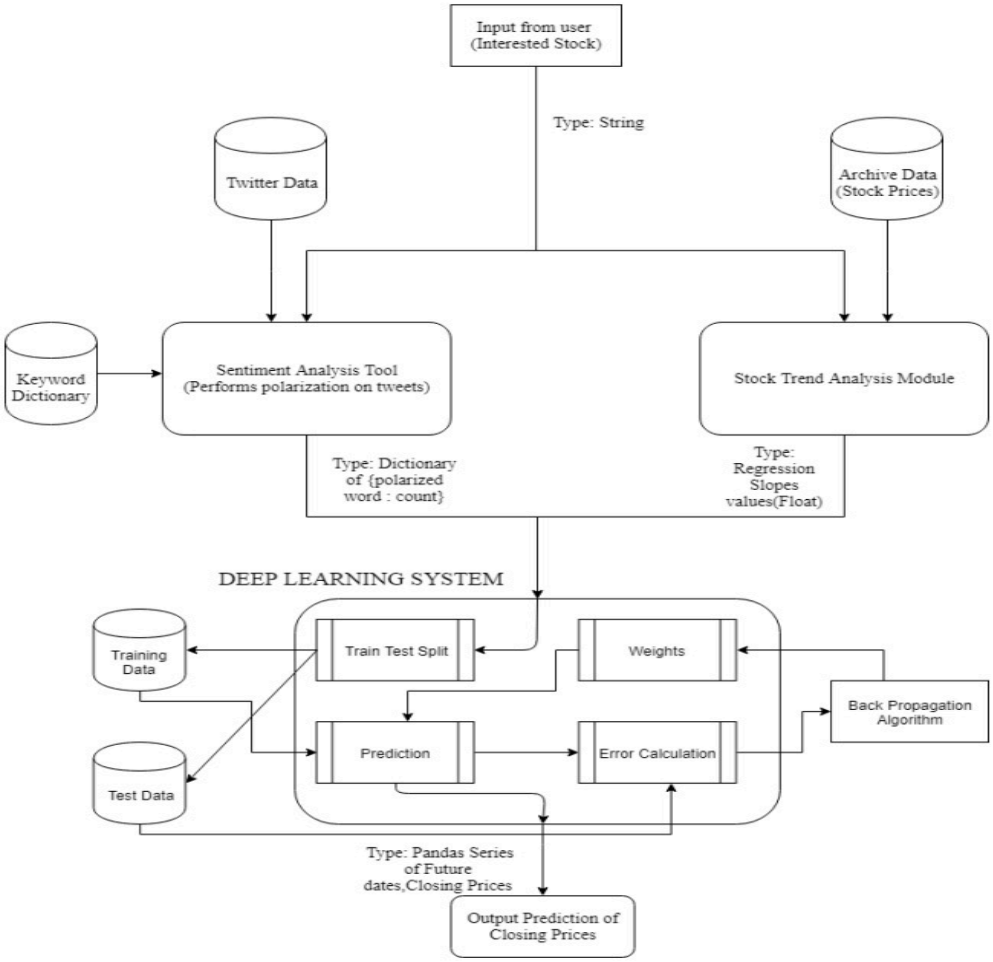


Figure 4 Deep Learning System

1.6 Past Works

Researchers used various statistical methods such as vector autoregression, “autoregressive moving average (ARMA)”[4] and “autoregressive integrated moving average (ARIMA)”[4] that seems to be efficient in evaluating the relationships between variables in statistical domain. Universal Approximation Theorem which states that any finite and continuous function could be approximated.

As per Wang and Leu(1996), they created hybrid model by combining two techniques such as using ARIMA model for extracting input features and RNNs (Recurrent Neural Network) for training mid-term stock market trend prediction system. The computational results of such fusion models outperformed the single econometric models and other individual forecasts. This encouraged further hybrid models, by combining different models such as per Patel, Shah, Thakkar and Kotecha (2015) [8]

1.7 Recent Trends

Recently LSTM (Long Short-Term Memory) networks have been used to find the trends in time series predictions. They are structured to identify and learn temporal patterns. LSTMs are an improvement of conventional RNNs, since they are better at learning long-term dependencies and also overcome the problem of gradient descent. They automatically detect suitable patterns in the given data. LSTMs have evidently yielded better results than many baseline models like multi-layer perceptrons, random forests etc. LSTM-based models have also been found to provide lower risks in trading strategies. Li, Bu and Wu (2017)[9] have mentioned in their paper that, other sequential data can also be fed into LSTM networks such as investor sentiment data.

1.8 Results

The different models applied and tested for stock prediction in this project have been evaluated on the basis of minimum squared error or mse of the model on test data. The predictions and actual data are compared by plotting against the time axis and hyper parameter tuning is done to achieve better performance.

1.9 Aim and Objectives:

The main aim of this research is to propose an algorithm-based approach for stock trading or prediction for sizing of their trading positions. The current challenges in the conventional prediction model such as not so good accuracy due to high volatile nature of stock market and the correlation between the stock price movement and sentiment or news data are addressed in this research thesis. This thesis proposed a hybrid model implemented using LSTM which consumes historical stock data along with sentiment data in the form or polarity (from various news articles and twitter data)

The research objectives are as follows:

- Analyze the data-set available for learning process
- To identify the learning methodology for prediction using LTSM
- Predict the stock price using the LTSM based RNN model
- Evaluate the performance of the model in terms of accuracy and the trading position with maximum profit (sell at high, but at low)
- Future trend of a stock will be analyzed from news articles as primary source of information.

1.10 Thesis Structure

Chapter 2 will discuss the **"Background and related research Research".** This chapter explains details about Stock Market and its domain specific details like Technical Analysis and Fundamental Analysis and also Sentimental Analysis too are well described. This chapter starts with explanation about the various data sources and how these data sources can be used for predicting stock price movement. Eventually more details on various prediction methodologies and the various techniques used to make prediction more efficient like RNN and then followed by LSTM.

Chapter 3 will give a brief description about various **Literature Review** happened around stock market price prediction in the past and also the recent research work in this domain. The literature review starts with review of the stock market fundamentals along with various prediction methodologies documented in various literatures. Eventually focus is more on LTSM and sentiment analysis and also the impact of market sentiments (news data and twitter data) which impacts the stock price over period of time.

Chapter 4 will describe more about the research methodologies used in the research thesis. The briefing of methodologies starts with Weka and YALE framework for data processing. Also explains in detail about the data gathering and pre-processing methodologies used to make the final DataFrame with meaningful data for feature extraction. Sentiment analysis helps in identifying the polarity of the twitter or stock news data which is used as one of the feature while forecasting the stock price based on historical data.

In Chapter 5 description about Model strategies used while predicting the time series data and the intricacies in various model strategies. This chapter also explains more about the advantages of different model architectures and the appropriate reason for choosing LSTM for stock price prediction in this thesis.

Chapter 6 describes about the various software tools/libraries used for forecasting stock prices based on various features which includes polarity and subjectivity of the stock news and twitter data. Most commonly used deep learning libraries like Keras and Tensorflow are used for building models and model fit function to forecast the stock price.

Chapter 7 describes more about the implementation details. This chapter explains in detailed about the LSTM network creation and Model building until data visualization off predicted outcomes. This is one of the most interesting chapter which briefs with the help of UML diagram about the use case and also more details about each step in the LSTM based stock prediction. This chapter begins with LSTM Network creation with the help of MATLAB and then various support/utility functions required for data processing, feature engineering and finally the stock price prediction.

Chapter 8 describes about the outcome of the prediction model and evaluating the performance metrics of the implemented algorithm. Performance metrics can be in terms of the right buy/sell signal indicated by the algorithm and various metrics like accuracy RMSE, LMS.

Chapter 9 describes more about the conclusions based on the previous chapter (where-in the outcome of the prediction model is evaluated based on performance metrics like accuracy of prediction. This chapter explains more about the assumptions, challenges faced while constructing this prediction model and also explains the learning associated with the challenges

Chapter 10 describes the expected outcome of the model and the comparison against various performance metrics. This chapter is trying to cover the impact of stock news and twitter data (sentiments) on stock price. The outcome of the model clearly indicates that, twitter sentiments has significant impact on stock price prediction in a short term as well long term as well. The model is benchmarked based on the accuracy at which it triggers buy and sell signals for the given timeframe.

Chapter 11 describes the overall summary of the thesis, starting from the literature survey and then finally the performance metrics of the implemented model. This chapter also explains about the future work . It is observed based on the implementation is that, best result is obtained when combination of Stock, Twitter and News Data are used.

Chapter 12 describes about the limitation and the challenges in the current model and how the outcome of the model can be improved in the Future Work section. Also there is an opinion about building UI model for portfolio management. based on the prediction outcome.

2 Background and related research

2.1 Finance Glossary

To get a better understanding of all this let's get done with the basic terminologies first

Share: A unit of ownership that entitles the holder to an equal proportion of the company's capital.

Stock: A collection of shares owned by a member that signifies his/her ownership of the company. Unlike shares, stocks can have different denominations.

Stock Trading: Buying and selling of stocks in the stock market with the stock trading plan which can achieve the best returns on their investment. There are many types of stock trading including Short-term, Intra-day, Long-term, swing trading etc.

Algorithmic Trading: The application of automated pre-programmed algorithms to execute orders in stock trading using various strategies like estimated moving averages, mean reversion.

Back-testing: The process of applying an analytical method or a trading strategy to real-time or historic data in order to check the efficiency of the strategy or method.

Time Series data: A sequence of data points recorded at regular time intervals, over a given period of time. Time is hence the primary axis here. For example, tracking applications and the Financial trading systems render time series data.

Sharpe Ratio: The term is coined by William F. Sharpe. This ratio helps investors to understand the return of an investment in an algorithmic trading.

Buy and Sell: Buy means buying stock(s) and Sell means selling stock(s). Investors take help of different algorithmic trading strategy to take decision when to Buy or Sell. Ultimate motive is to make profit though these.

Portfolio: Portfolio is collection of different financial assets such as stocks, commodities, bonds, currencies, cash equivalents etc.

Uptrend and Downtrend: Uptrend is basically when a company's stock return is increasing day by day. And Downtrend is just the opposite

2.2 Stock Market Prediction Brief

2.2.1 Stock Market :

A Stock market is a regulated platform for the sale at an negotiated price of commodities and derivatives; they are shares and only exchanged secretly. It is managed in accordance with a regulatory authority and participants dealing in the securities are registered with SEBI. The equity market, which includes trade between two creditors, is sometimes considered the secondary market. Share exchange puts together creditors to acquire and sell business stock. Share market rates are dependent on demand and availability. A strongly needed product will raise prices whereas a poorly marketed product would decrease prices.

Figure 5 Infy Technical Chart

2.3 Fundamental Analysis Based Stock Prediction

2.3.1 Introduction to Fundamental Analysis

Fundamental analysis discusses supply and demand-related variables. This information is to be collected and interpreted and used in stock prices before the information is included. There is a trading gap between the time between an incident and the subsequent consumer reaction. Fundamental forecasting is focused on business economic statistics which aims to estimate economies with economic records, such as yearly which quarterly accounts, accounting accounts and earnings statements, which businesses are expected to post annually.

The advantages of fundamental analysis:

- Its systematic approach and its ability to predict changes before they show up on the charts.
- Fundamental analysis is a superior method for long-term stability and growth.

Disadvantages of fundamental analysis:

- It becomes harder to formalize all this knowledge for purposes of automation (with a neural network for example), and interpretation of this knowledge may be subjective.
- It is hard to time the market using fundamental analysis.

2.3.2 Important Ratios for fundamental analysis:

Several ratios can decide the main principle of stock evaluation. Each has its own meaning. Any share not representing all these conditions as safe would not be a successful buy. The definition for each ratio are referred from Investopedia.com

- **The Price-to-Book Ratio (P/B):**. "Companies use the price-to-book ratio (P/B ratio) to compare a firm's market capitalization to its book value. It's calculated by dividing the company's stock price per share by its book value per share (BVPS). An asset's book value is equal to its carrying value on the balance sheet, and companies calculate it netting the asset against its accumulated depreciation."

$$P/B\ Ratio = \frac{Market\ Price\ per\ Share}{Book\ Value\ per\ Share}$$

- **Price-to-Earnings Ratio (P/E):** "The price-to-earnings ratio (P/E ratio) is the ratio for valuing a company that measures its current share price relative to its per-share earnings (EPS). The price-to-earnings ratio is also sometimes known as the price multiple or the earnings multiple. P/E ratios are used by investors and analysts to determine the relative value of a company's shares in an apples-to-apples comparison. It can also be used to compare a company against its own historical record or to compare aggregate markets against one another or over time"
- **The PEG Ratio:** "The price/earnings to growth ratio (PEG ratio) is a stock's price-to-earnings (P/E) ratio divided by the growth rate of its earnings for a specified time period.

The PEG ratio is used to determine a stock's value while also factoring in the company's expected earnings growth and is thought to provide a more complete picture than the more standard P/E ratio."

- **Dividend Yield:** "The dividend yield is the ratio of a company's annual dividend compared to its share price".

- **Returns on Equity (ROE):** "Return on equity (ROE) is a measure of financial performance calculated by dividing net income by shareholders' equity. Because shareholders' equity is equal to a company's assets minus its debt, ROE is considered the return on net assets. ROE is considered a measure of how effectively management is using a company's assets to create profits."

- **Debt to Equity Ratio:** "The debt-to-equity (D/E) ratio is calculated by dividing a company's total liabilities by its shareholder equity. These numbers are available on the balance sheet of a company's financial statements. The ratio is used to evaluate a company's financial leverage. The D/E ratio is an important metric used in corporate finance. It is a measure of the degree to which a company is financing its operations through debt versus wholly-owned funds"

2.4 Technical Analysis based Stock Prediction

2.4.1 Introduction to Technical Analysis:

It is based on mining rules and patterns (using charts) from the past prices of stocks which are called mining of financial time series. The basic principles include concepts such as the trending nature of prices, confirmation and divergence, and the effect of traded volume. Many methods for prediction of stock prices have been developed and are still being developed on the ground of these basic principles. Technical analysis looks for patterns and indicators on stock charts that will determine a stocks future performance.

In recent years, neural networks have been widely implemented to boost multi-variate prediction efficiency with time series issues. Neural networks have a strong capacity to generalize input and output values through mapping of defined patterns. Neural networks are typically reliable in time series prediction problems against noisy or incomplete data which are highly desirable properties for the stock market research, several neural network models have already been created.

The advantages of technical analysis:

- Mostly used for short term investment and intra-day traders who usually follows technical analysis.

Disadvantages of technical analysis

- Major drawback on technical analysis is that, there is no standard evidence on chart reading.
- Each analysts interprets the charts on their own especially when analysts reviewing charts using Elliot Wave theory, interpretation keeps changing per analysts resulted in wrong entry and wrong exit in market.

2.4.2 Important Parameters for technical analysis:

There are close to 52 different technical metrics/indicators are available. These metrics are sometimes called as indicators or oscillators which has been traditionally used by technical analysts to make short term investment prediction or intra-day trading purpose. Therefore, the parameters (the feature vectors of financial data) that most strongly predict the movement's existence need to be defined without the device complexity[5][6]. Definitions of various indicators are referred from Invesopedia.com

a) **Moving Average (MA):** "A moving average (MA) is a widely used indicator in technical analysis that helps smooth out price action by filtering out the "noise" from random short-term price fluctuations."

b) **Exponential Moving Average (EMA):** "Exponential moving averages (EMAs) are also weighted toward the most recent prices, but the rate of decrease between one price and its preceding price is not consistent. The difference in the decrease is exponential"

c) **Relative Strength Index (RSI):** "The relative strength index (RSI) is a momentum indicator that measures the magnitude of recent price changes to evaluate overbought or oversold conditions in the price of a stock or other asset. The RSI is displayed as an oscillator (a line graph that moves between two extremes) and can have a reading from 0 to 100"

2.4.3 List of Trading Strategies: Each with appropriate market environments and risks inherent in the strategy. Trading strategy is a technique of buying and selling in the markets that is based on predefined rules used to make trading decisions.

Trend-following Strategies	Algorithmic Trading Strategies
Statistical Arbitrage	Arbitrage Opportunities
Trading Range (Mean Reversion)	Technical Analysis
Swing Trading Strategy Scalping (Trading)	Day Trading
Trading the Signals Social Trading	Value Investing

Table 1 List of Trading Strategies

2.4.4 List of Portfolio Strategies: Portfolio strategies is an investment method for investors to use their assets to achieve their financial goals.

Long-term Investment	Capital Asset Pricing Model (CAPM)
Short-term Investment	Momentum Investment
Buy and Hold	Portfolio Optimization
Rebalance Portfolio	Dynamic Asset Allocation
Value Investment	Portfolio Allocation

Table 2 List of Portfolio Strategies

2.4.5 List Type of Risks: Risk measures are statistical method to define the individual stock or together to perform a risk assessment.

Trade Risk	Position Size Risk
Liquidity Risk	Overnight Risk
Market Risk	Volatility Risk

Table 3 List of Risk Strategies

2.4.6 List of Risk-Adjusted Returns Ratios Measurement: An investment's return by measuring how much risk is involved in producing that return.

Calmar Ratio	Gain Loss Ratio
Martin Ratio	Sharpe Ratio
Modigliani Ratio	Pain Ratio

Table 4 List of Risk-Adjusted Returns Ratios

2.5 Sentiment analysis for Stock Prediction:

The social mood plays a significant role in the price movement, in addition to historical prices. The general social sentiment about a company is now seen as a significant aspect influencing its stock price.

Sentiment analysis (SA) is a common method which is increasingly used to assess the feelings of social media users towards a subject. The most popular approach performing sentiment analysis is using data mining. The central idea is to adopt Deep Learning to determine investors' expectations about the price of stocks and the overall market based on their messages. A Crawler service can be used to read news articles or news data continuously about a specific stock and index those news polarity data for further correlation analysis. News Classifier can be used to segregate news data based on their associated sentiments as good (positive), bad(negative) and Neutral.

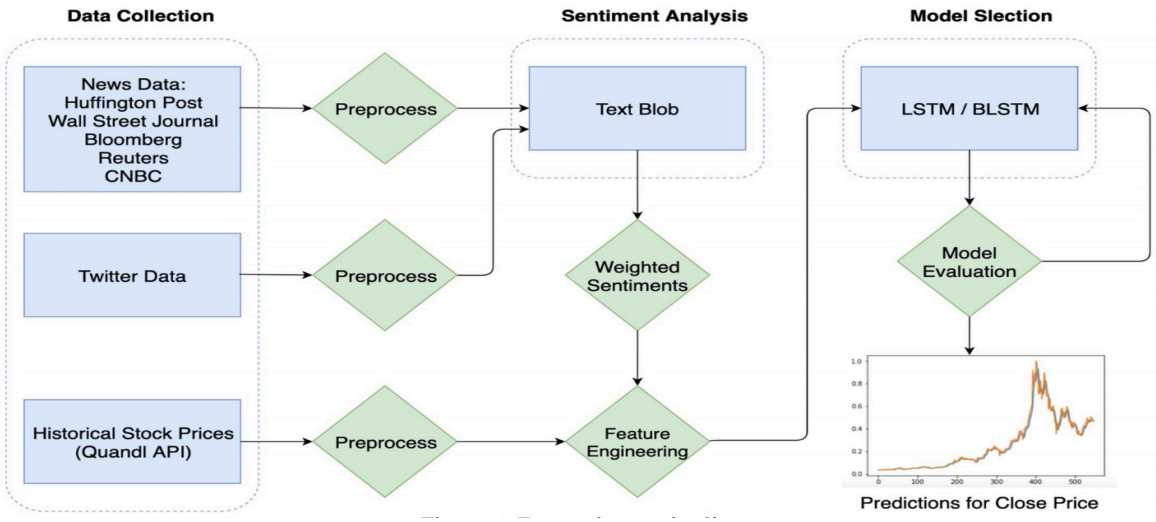


Figure 6 Prediction Model for Sentiment Analysis [12]

Geo-political and economic sentiments can have strong impact on Stock Exchange such as any trade war related news or sanctions on a particular country are considered as negative news which can impact respective country specific stock exchange. Supply and Demand is the business philosophy which can impact the rise/fall of a stock market. Regardless of shifts in supply and demand the market markets are rising and dropping every second. Machine learning can help to address issues related to portfolio and commodity values with greater accuracy, efficiency and accuracy. In large amounts of social network results, the representation of moods and emotions is extremely significant in calculating investors' opinions.

In forecasting equity values dependent on popular opinions, Twitter data recently acquired an impetus. The explanation is that Twitter data is now available and therefore affects market sentiment immediately and rapidly. Good reports and tweets regarding a business on social media will certainly inspire people to invest in the company's securities, thus increasing the company's stock price. A prediction model can be built using machine learning to define and evaluate the association between tweet material, product prices and predictive predictions.

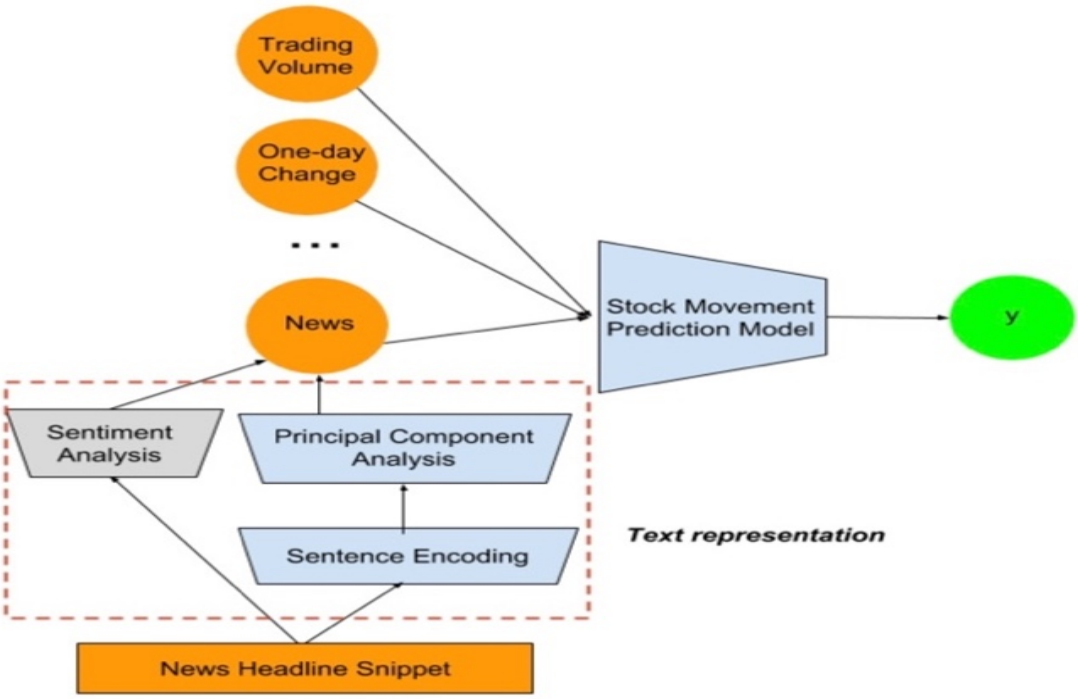


Figure 7 Sentiment Analysis Tool Overview

2.5.1 Sentiment classification techniques:

Three broad categories of sentiment classification techniques are:

- **Rules-based techniques:** Based on a pre-built dictionary of words and their polarities. Sentiment of the text is then defined by counting positive and negative words in the text.
- **Machine learning- based classifiers:** do not rely on deterministic rules. They transfer text to numerical representations (feature mapping) and map to sentiment labels.
- **Bag of Words (BoW):** In this method, text is represented as a bag of words (unigrams) or a collection of words (n-grams). Each word is mapped to a frequency in the document (Term Frequency-Inverse Document Frequency (TF-IDF) approach).
- **Word embedding** technique that maps each word, or collection of words, to a vector. The mapping is done in a way where semantically similar words are positioned closer to each other in the vector space, thus providing the to capture their relationship to one another.
- **Language modelling:** Used to initialize a first layer of neural network provides a shallow representation of the language. The network should still learn how to drive meaning from a sequence of words and relate them to sentiment labels through layers and embedding layer.
- **SDA (Stacked Denoising Auto Encoder)** is applied to reduce the dimension of features which is not sensitive to the noise. [11] An autoencoder is a type of artificial neural network used to do unsupervised learning of data coding. The aim of an auto encoder is to learn higher-level representation for a set of data, typically for dimension reduction.

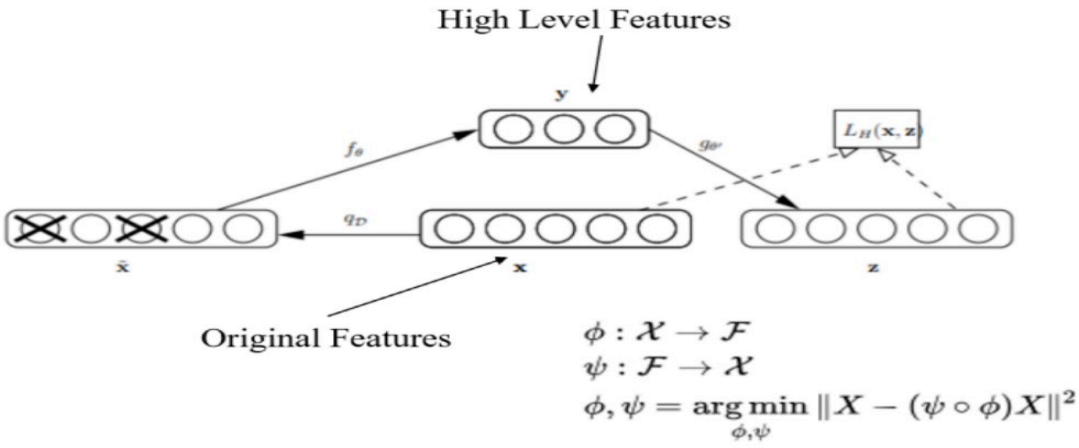


Figure 8 SDAE Auto Encoder Framework

2.6 Data Sources

2.6.1 Historical Stock Prices Data

Historical stock prices data can be availed through Quandl API and Yahoo Financials wrapper to get the data. The dataset contains following fields: Date, Open, High, Low, Close and Volume.

Twitter - Twitter is an online social networking and microblogging service that enables users & organizations to send/read messages, called "tweets". Users/organizations are free to post their content (tweets) to their followers(listeners) read them.

Twitter sentiments are generated classified according to the emotions associated with the tweet such positive, negative and neutral. Depending on the areas of application we can even make advanced classifications and judge if the user is angry, happy or sad. Twitter provides a unique functionality to access their tweets with the help of their APIs to filter out the required data.

Twitter Data

- twitter data from Internet Archive collection of twitter stream.
- It contains the tweets in a nested structure as: Year → Month → Date → Hour → Minute.

News Data

- News Dataset from Kaggle News Category Dataset and Kaggle US Financial News Articles.
- Kaggle News Category Dataset contains approximately 200k news articles of various categories from 2012 to 2018 obtained from Huffington posts.
- The fields in this dataset include date, authors, category, headline, short description and link. Since the headlines and the short description fields did not provide enough information regarding the articles. The news articles are from the following news publishers: Bloomberg.com, CNBC.com, reuters.com, wsj.com, fortune.com.

2.7 Sentiment Detection Algorithm:

Dictionary Based Sentimental Model:

Two kinds of terms sets, i.e. positive terms and negative terms are used to create the polarity dictionary. The score (polarity words of positive and negative polarity) for this text is determined by checking the occurrence of the terms in the dictionary.

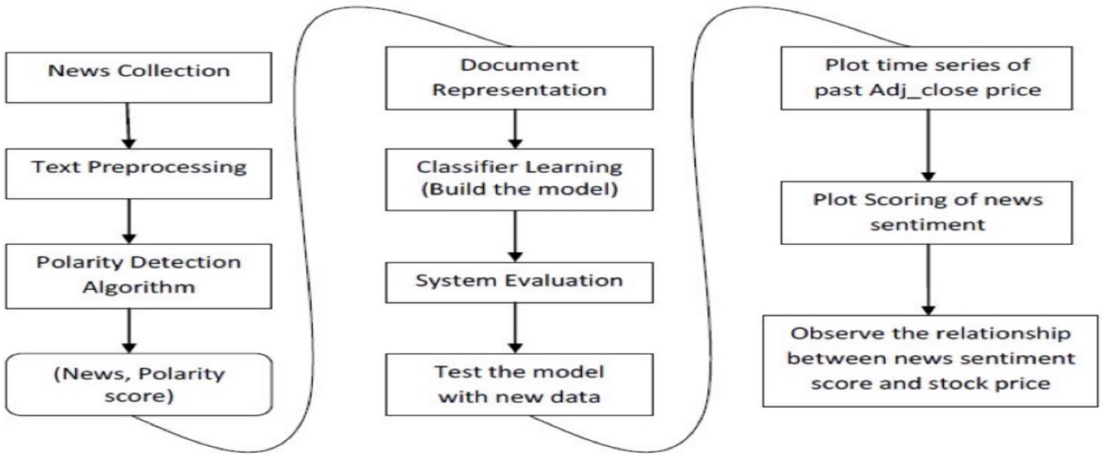


Figure 9 NLP Processing for News and Twitter Data [13]

2.8 **RNN (recurrent neural network) :**

Definition of RNN is referred from Wikipedia as follows

"A recurrent neural network (RNN) is a class of artificial neural networks where connections between nodes form a directed graph along a temporal sequence."[13]

- It can exhibit temporal dynamic behaviour.
- Since this is possessing feed-forward neural networks nature, RNN can use their memory to process variable length of inputs esp. useful for speech recognition and stock market prediction.
- Its deep learning model that is mostly used for analysis of sequential data (time series data prediction).
- RNN is a recursive function since for all data inputs it conducts the same function, although the output of the present input is based on the last one.

- **There are different application areas that are used:** Language model, neural machine translation, music generation, time series prediction, financial prediction, etc.

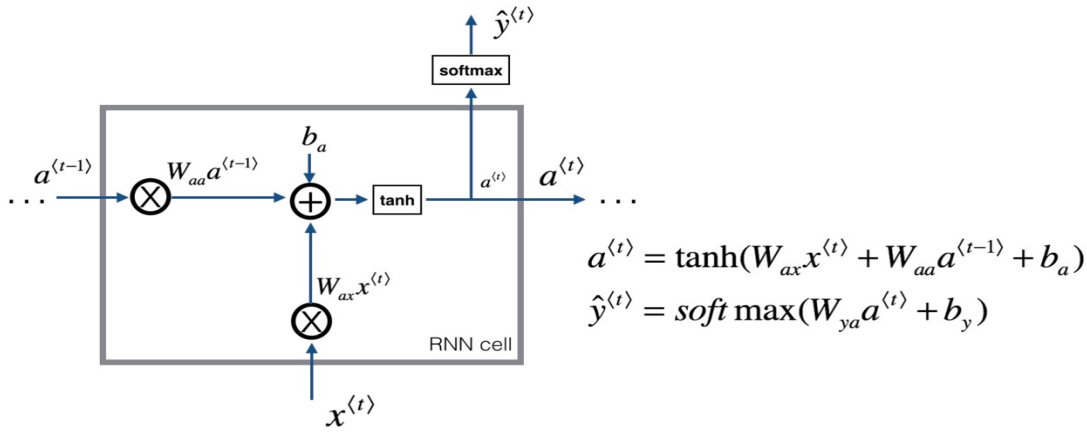


Figure 10 LSTM Cell Diagram [14]

2.8.1 How it is different than other neural network models

- Traditional neural network input's is not dependent on output.
- Suppose if we want to predict a day's return of a stock market, then it is necessary to know previous days' return index.
- What 'recurrent' means in RNN is repeating the same task for all the element in the sequence. RNN has 'memory cell', which store the information about the previous elements.
- RNN works best in field of Natural Language Processing and time series data.

Here unfold means unrolling i.e. expanding into the full network. For example, if an RNN is to be used in a 10-word sentence, the RNN could be unrolled into 10 layered neural networks, each layer represents one layer.

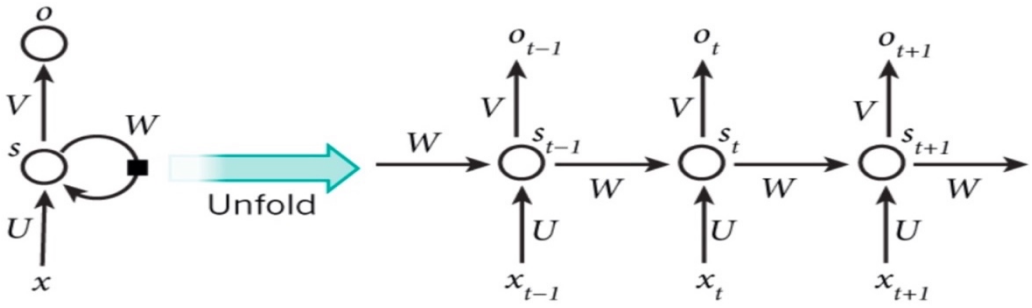


Figure 11 LSTM Cell with Unfolded Diagram [15]

2.8.2 Few RNN's Applications:

2.8.2.1 Language modelling and prediction

- the probability of a word occurring in a sentence in a particular position with respect to other words is considered.
- In language modelling, a sequence of words is the model and the output will be a sequence of predicted words by the model. The model basically computes how likely a sentence can be, which is an important step for machine translation module.
- Predicting the next word is the motive of generative model, which tries to generate new text by sampling from the output probabilities of the given input text.
- One of the methods for language modelling is using one hot encoding.

2.8.2.2 Machine Translation

In this one language is the input and output will be any other target language. The difference between machine translation and language modelling is that the output starts after the model sees the input completely as shown in the figure.

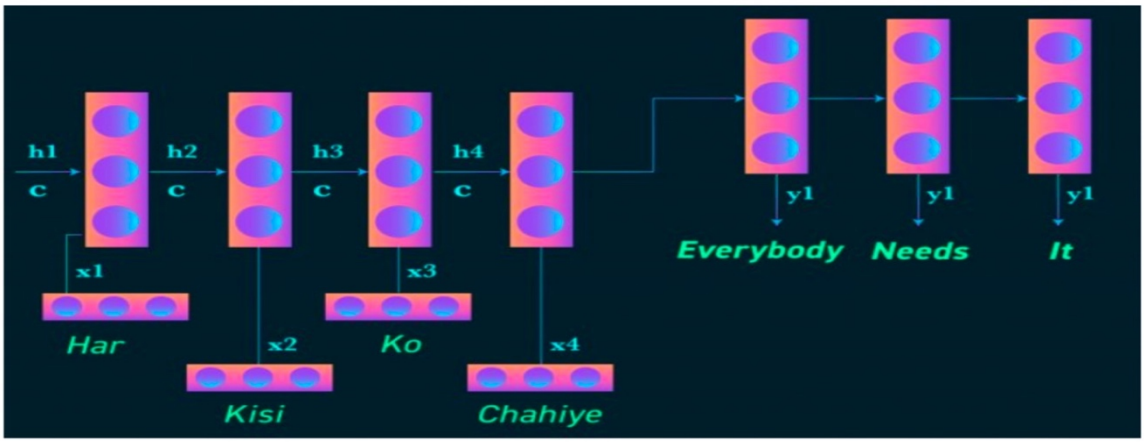


Figure 12 Machine Translation (RNN Application)

2.8.2.3 Image recognition and generating description

Convolution neural network (CNN) can be used with RNN to generate image description. Results can be very impressive after constructing the model properly.

2.8.2.4 Speech Recognition

Given phoneme (sequence of acoustic signal from sound wave) as input, it predicts the sequence of phonetic segments.

2.8.3 Training RNN

Like the other neural network RNN needs to be trained. Here backpropagation algorithm is also used with a slight modification as not only the current time step is required to compute the output but also all the previous time steps are also important.

2.8.4 RNN extension

Researchers have been developing so many things in RNN. Some of the developments are as follows.

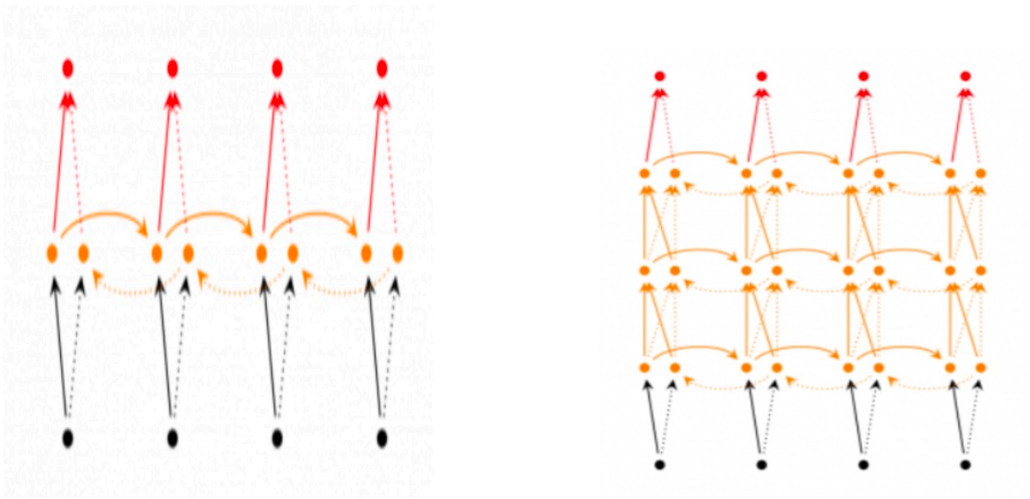


Figure 13 Bi-directional RNN vs Deep RNN

Bidirectional RNN	Bidirectional RNN requires elements of previous time steps as well as elements of future time steps. Bidirectional RNN comprises two RNNs stacked on the top of each other.
Deep (Bidirectional) RNN	It is similar to the bidirectional RNN with multiple layers are stacker per time step. This requires lots of training data but also gives higher learning rates.

Table 5 Bi-Directional RNN vs Deep RNN

2.8.5 Different types of RNN

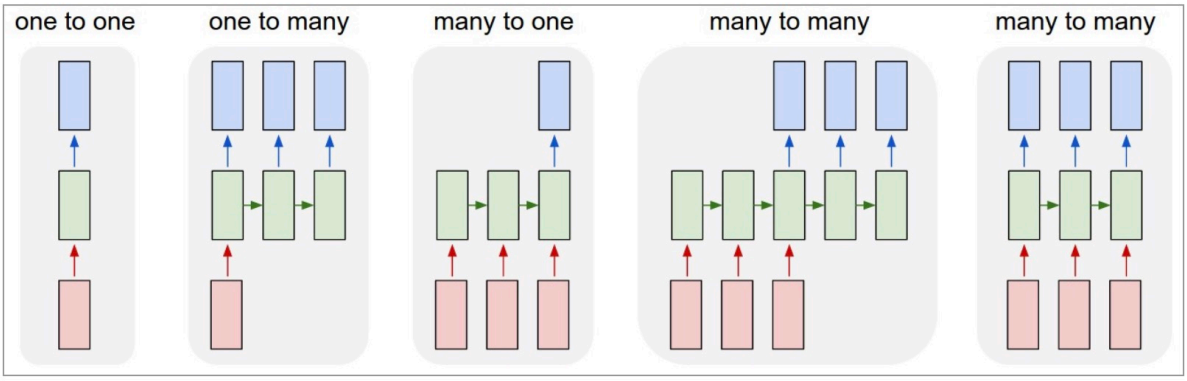


Figure 14 Types of RNN

One to one (vanilla)	it is the basic model of RNN
One to many	Here number of inputs is one and number of outputs is more than one. **Example: image captioning** Here an image is given, and task is to predict some sequence of words as the description of the model.
Many to one	Here more than one input layer is used, and only output is available. **Example: Sentiment classification** Suppose a paragraph i.e. a bulk of words is given, and the sentiment of the para has to be determined. In this type of case many to one model is used.
Many to many (where number of inputs is not equal to number of output)	Here number of inputs is more than one and number of outputs is more than one, but the number of input and output may or may not be equal. **Example: Machine Translation**
Many to many (where number of inputs is equal to number of output)	Here the number of inputs is equal to number of outputs. **Example: Video classification on frame level** Here first video is divided into frames and then frames are labelled.

Table 6 Types of RNN

2.8.6 Mathematical formulas of RNN

$$h_t = f_W(h_{t-1}, x_t)$$

Notice: the same function and the same set of parameters are used at every time step.

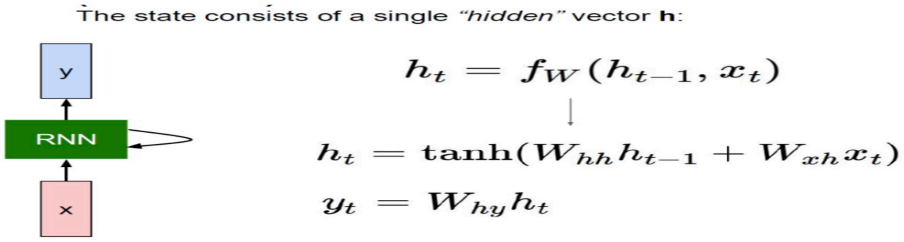


Figure 15 Mathematical Model of RNN

2.8.7 RNN: Computation Graphs

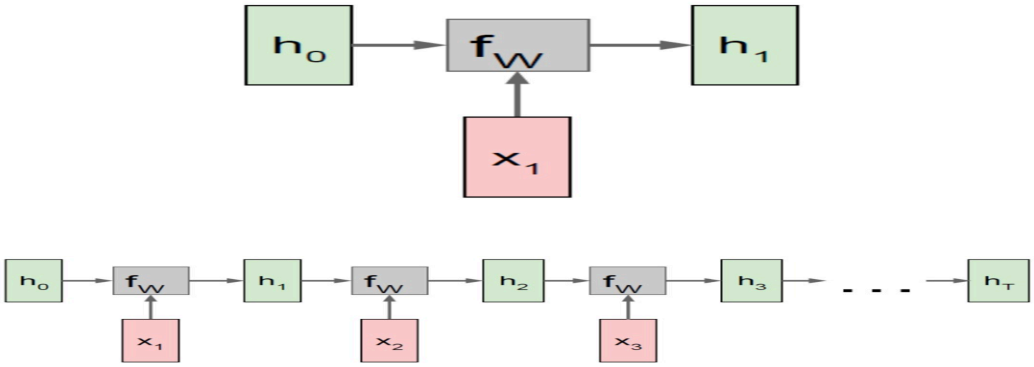


Figure 16 RNN Computation Graphs

2.8.8 Problems with RNN

2.8.8.1 RNN suffers from Gradient flow problem

- In a neural network a very basic but important task is to optimize the network. And by optimization it is meant to minimize the cost of learning and find the values of hyper parameters of the model like learning rate, batch size, no of epochs.

- Parameters are the properties of the training data that are learnt during training. These define the skill of the model. These are not usually set by the developer. Example: weights of a network.

- Hyperparameters are those which cannot be learnt from the model. But these are to be set before training. They help to estimate the parameters. Hyperparameters have to set by developer. Tuning hyperparameters helps the model to work more efficiently. Example: learning rate, decay, number of hidden layers, number of epochs etc.

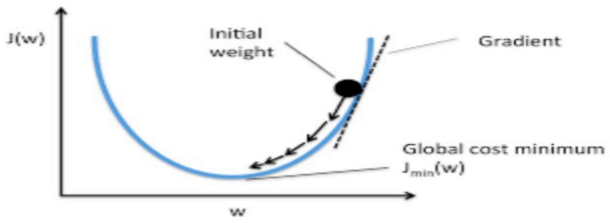


Figure 17 Gradient Descent

2.8.8.2 Mathematical procedure

- Suppose the initial values of the coefficient for the function is 0.0. Coefficient = 0.0 And suppose cost function is f, i.e. cost = f(coefficient)

- Derivative is used to calculate the optimization point. So here the derivation of the cost is calculated, i.e. basically the slope of a function at a given point (diff = derivative(cost))

A parameter is used to control over the update in each iteration. This parameter is called learning rate. This process is repeated until the cost of the coefficients is nearly zero.

- coefficient = coefficient – (lr∗ diff); here lr stands for learning rate.

- gradient descent algorithm often has problems of exploding gradient and vanishing gradient.

- Though gradient descent works quite similarly for RNN's, there are some add ones:

- The information propagates through timesteps in RNN.

- At each time step cost function has to calculated.

- After calculating et, it is propagated back into the network to update the weights.

- So, if the gradient is very small and as it is multiplied back in each timesteps it will just result near zero result and if the gradient is very high it will just explode in same way.

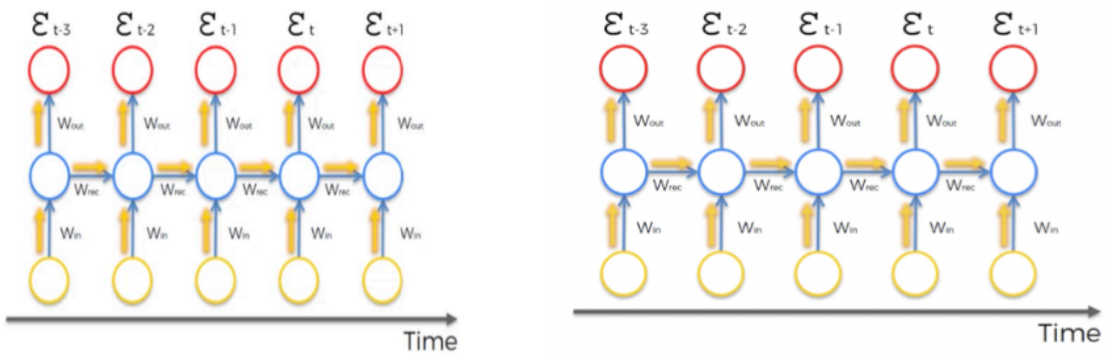


Figure 18 RNN with multiple Co-Efficient

2.8.8.3 Solutions to the vanishing gradient problems

In case of exploding gradient:

- Back propagating is stopped after some point
- Reduce gradient in some way
- Put a limit on gradient.

In case of vanishing gradient:

- Initial weight in a proper manner so that vanishing gradient is minimized.
- Having long short-term memory (LSTM) networks.

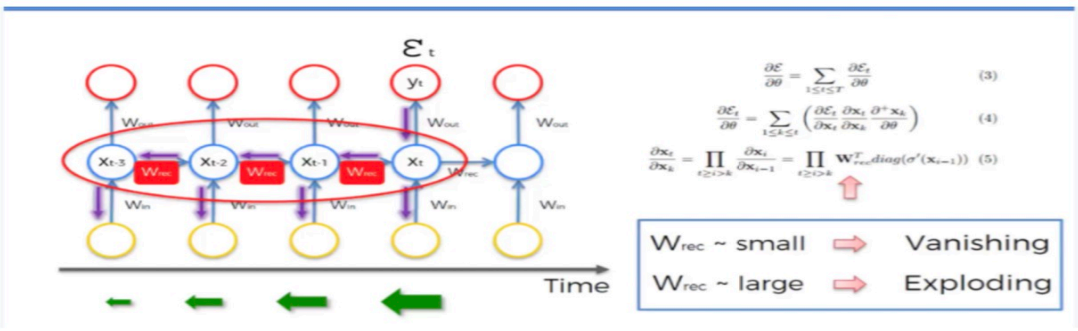


Figure 19 Gradient Descendant and Vanishing Problem

2.8.8.4 Gradient flow in RNN

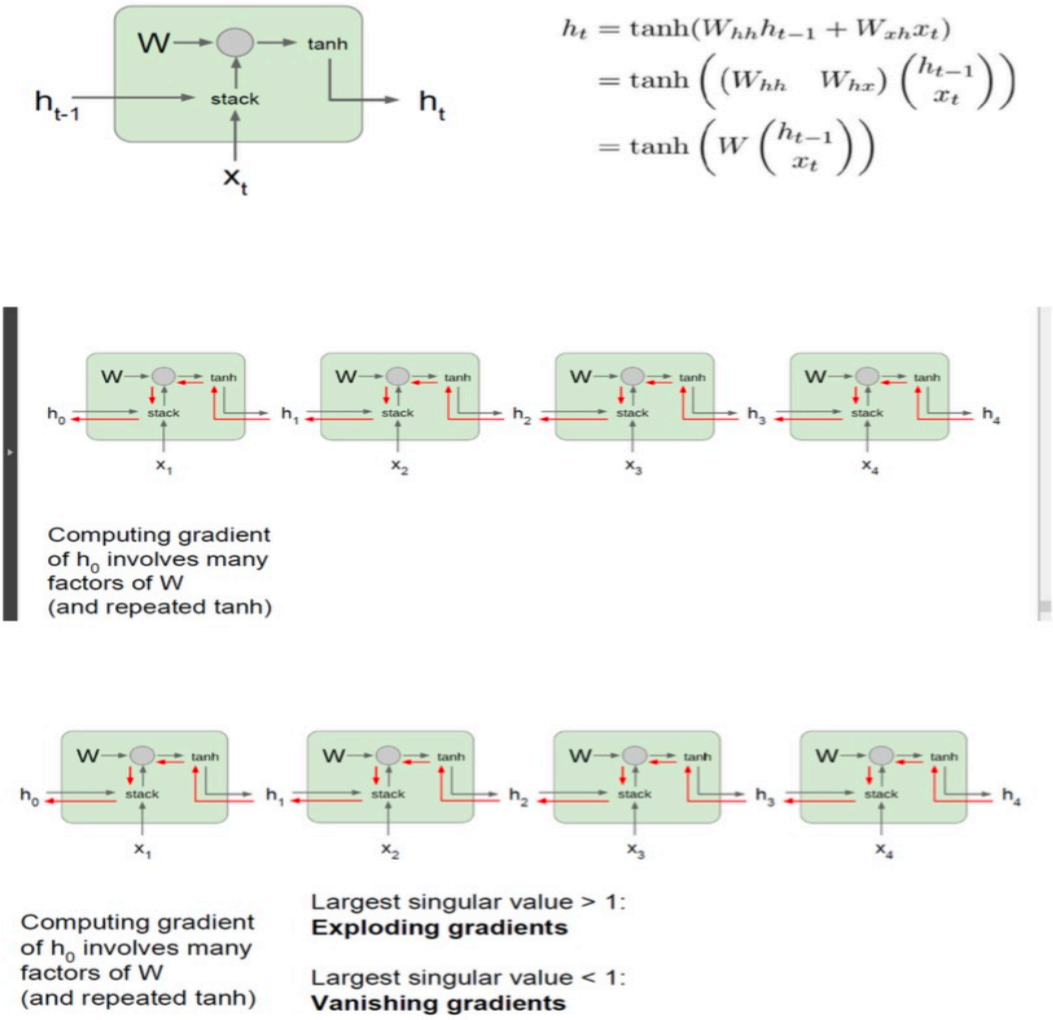


Figure 20 Gradient Flow in RNN

2.9 LSTM (Long Short-Term Memory)

- LSTM is a special kind of RNN, which extends memory. That's why it is perfect to learn from the patterns or trend that have very long-time difference in them. It also solves the Vanishing Gradient/ Exploding Gradient problem.

- LSTM is well-suited to classify, process and predict time series given time lags of unknown duration. It trains the model by using backpropagation

- It is a special type of RNN, capable of learning long-term dependencies.

- "Long short-term memory (LSTM) units are units of a recurrent neural network (RNN). An RNN composed of LSTM units is often called an LSTM network"[79]

- **There are different application areas that are used:** Language model, Neural machine translation, Music generation, Time series prediction, Financial prediction, Time series prediction, Speech recognition, Rhythm learning, Music composition, Grammar learning, Handwriting recognition, Time series anomaly detection.

- LSTM cell has four gates. Those are forge gate, input gate, gate to write the cell, output gate.

Figure 21 Vanilla RNN vs LSTM

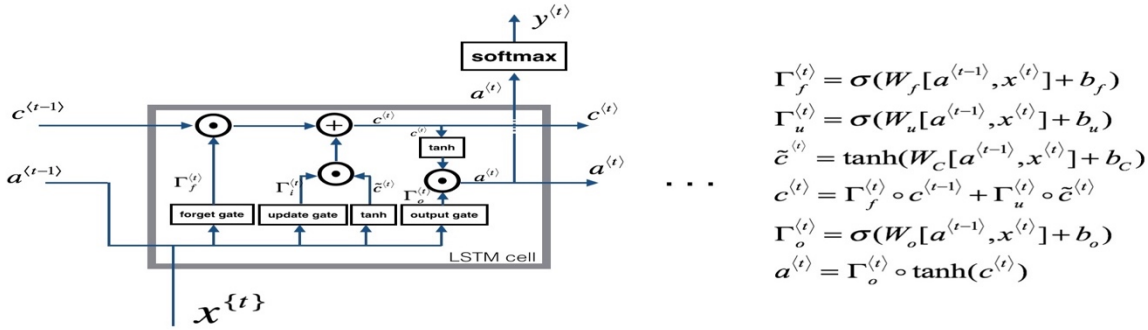


Figure 22 LSTM Cell with Softmax

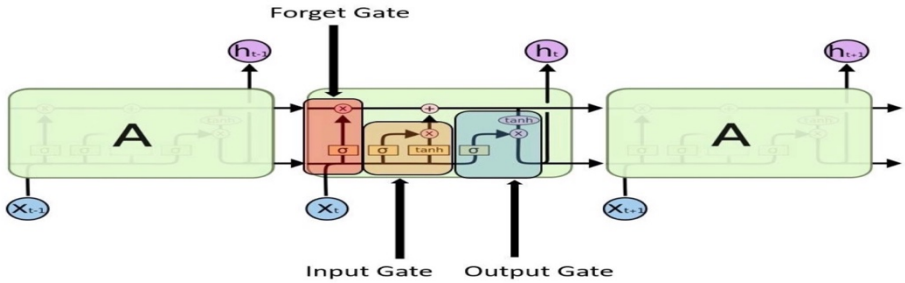
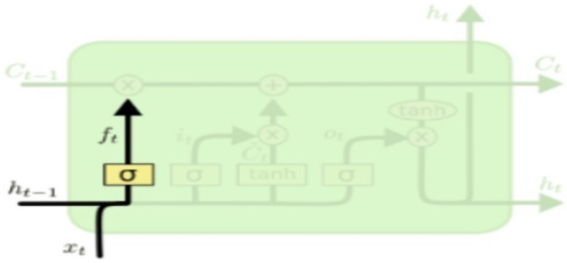


Figure 23 LSTM with four gates

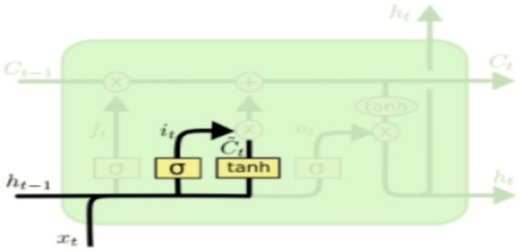
a) **Forget Gate:** After getting h(t-1) i.e. the output of previous step, this gate helps to take decision about how much information is to forget from the previous state. In this way it tries to keep only relevant stuff. It uses sigmoid function to keep the input between 0 and 1 both inclusive. The above diagram represents the Forget Gate.



$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \; + \; b_f\right)$$

Figure 24 LSTM Forged gates

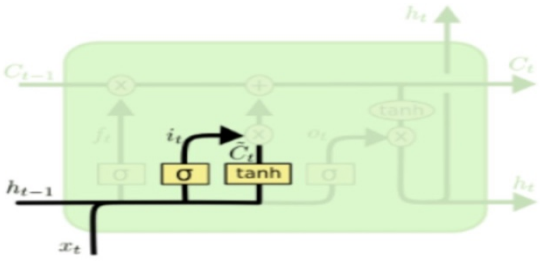
b) **Input Gate:** This gate decides whether to write the present cell from the present cell or not. tanh layer helps to create a vector to add to present state and sigmoid layer decides the values to be updated. The above picture describes the Input Gate.



$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \; + \; b_i\right)$$
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \; + \; b_C)$$

Figure 25 LSTM - Input Gate

c) **Output Gate:** It decides how much to revel as output. The following picture describe this cell.



$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \; + \; b_i\right)$$
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \; + \; b_C)$$

Figure 26 LSTM - Output Gate

2.9.1 LSTM Model in Sentiment Analysis

LSTM neural network is applied for sentiment analysis. Input is sentence vectors. Then split data into several parts by different companies. Put them into LSTM layer. Then add a dense layer. Output layer will output sentiment analysis result, value range from 0.0~1.0. And then a 4-quantile value is used. If value < value <= 75%quantile as 0(even); value > 75% quantile as 1(positive). Model design is showing below:

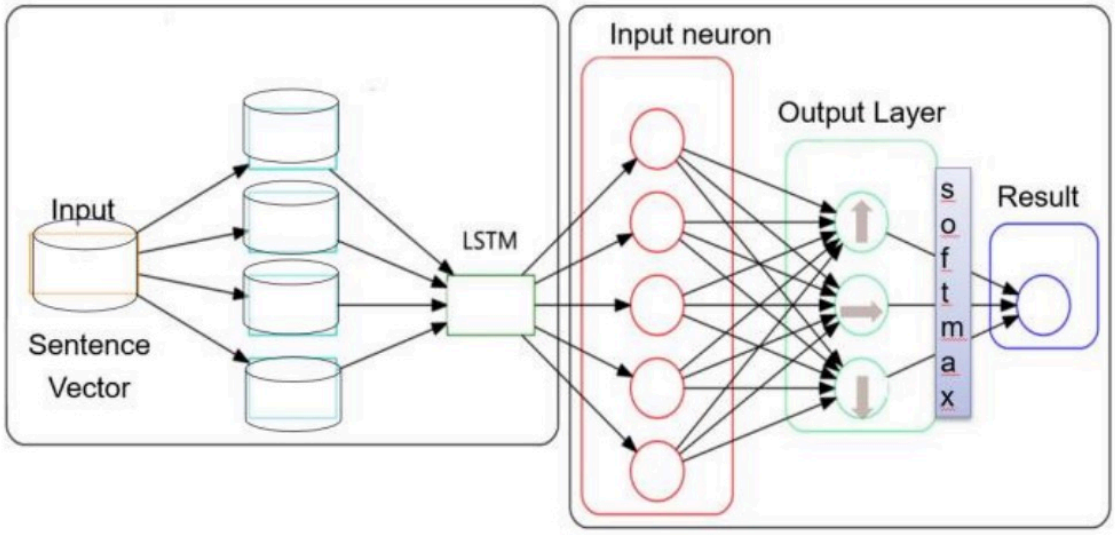


Figure 27 LSTM Sentiment Analysis Model [15]

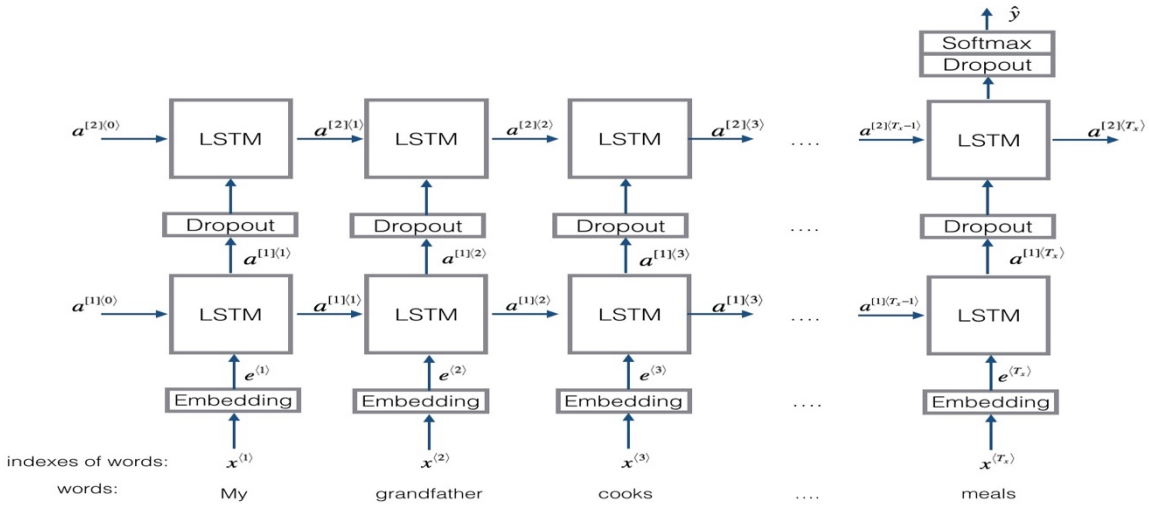


Figure 28 LSTM Sentiment Analyser [14]

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	(None, 10)	0
embedding_1 (Embedding)	(None, 10, 50)	20000050
lstm_1 (LSTM)	(None, 10, 128)	91648
dropout_1 (Dropout)	(None, 10, 128)	0
lstm_2 (LSTM)	(None, 128)	131584
dropout_2 (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 5)	645
activation_1 (Activation)	(None, 5)	0

Total params: 20,223,927
Trainable params: 20,223,927
Non-trainable params: 0

Figure 29 LSTM Sentiment Analysis Output Screenshot

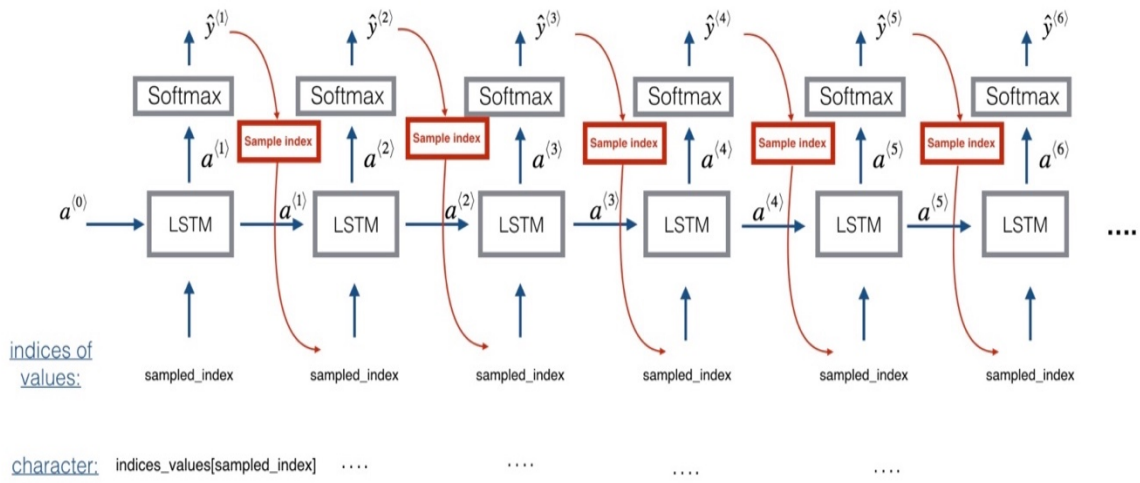


Figure 30 LSTM Softmax and Index Processing

2.10 Trading

A trading strategy states when to buy, sell or hold our financial assets in order to achieve better returns. It tells us to go long or short in markets at the given circumstances. There are two common trading strategies we deal with in this project:

2.10.1 Momentum Strategy

- Also known as divergence or trend trading, this strategy believes in the movement or lack of it, of a quantity i.e. it expects the movement to continue its trend.
- Stocks with an upward trend tend to move up and those with downward seem to go down.

Common examples of this strategy include:

a) **Moving Average Crossover b) Dual Moving Average Crossover c) Turtle Trading (Richard Dennis)**

2.10.2 Reversion Strategy

- Also known as convergence or cycle trading, this strategy begs to differ from the current trend and believes that the movement of an asset will eventually reverse.
- Mean Reversion strategy insists that stocks return to their mean prices and aims to exploit when stock prices deviate away from the mean. Some Common strategies in this category are**: a) Mean Reversion Strategy b) Pairs trading Mean Reversion Strategy**

Moving average crossover strategy of quantitative trading is used which uses two separate Simple Moving Averages (SMA) of a time-series with a short and a long lookback period.

If the short moving average exceeds the long moving average, then we buy a stock, or we go long and if the reverse happens then we exit, or we go short (sell the stock).

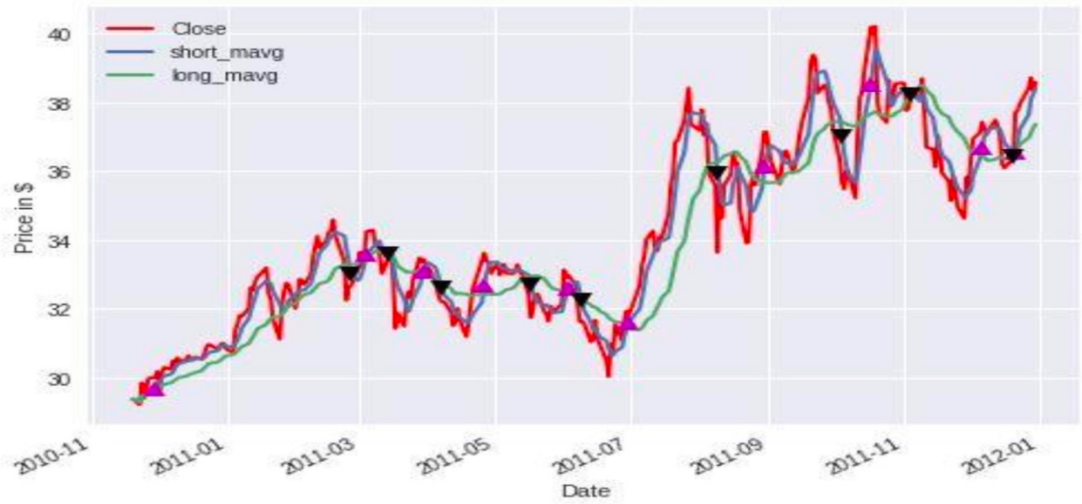


Figure 31 Stock MAVG for Short and Long

3 Literature Review

3.1 Introduction

In an effort to understand the advancement in the prediction of financial systems and stock markets using machine learning, this thesis aims to provide a systematic review of current research on various machine learning based prediction methodologies. I undertook an extensive literature review by examining relevant articles from major academic databases (IEEE Xplore, ACM, Springer digital library and Science-Direct). Key search terms include the key words 'Stock Prediction Methodologies', 'Artificial Neural Network', 'Machine Learning' and 'LSTM' and a wide range of other technologies.

In addition, I have reviewed the research projects related Stock predictions methodologies based on historical data, news data and also sentiment analysis etc, by searching from various journal databases and existing open source/funding projects. As a result, I have found a large number of journal articles and conference papers related to Stock Market Prediction methodologies studies and Sentiment Analysis based Market predictions respectively, and a number of opportunities for future researchers. A main contribution of this section is that it is the first attempt to categorize classic machine learning prediction methodologies into Neural network-based prediction method systematically; it reviews the current research of stock prediction, Sentiment analysis and also NLP, key enabling technologies, identifies research trends and challenges.

3.2 Problem Statement

The stock market is very volatile and depends on lots of different parameters hence it is very difficult to predict the future stock prices. There are few challenges in this research domain.

Challenges :

- It is very difficult to predict the future stock prices because the stock market is very volatile and depends on lots of different parameters hence.
- Complex natural language processing techniques are needed to get the sentiment for a particular company out of whole bunch of data.
- Identifying the exact features which would help in increasing the accuracy of the prediction to improve upon the existing models.
- Since stock data are highly volatile, prediction of share price can be achieved by combining multiple models together (Hybrid Model)
- It is very critical to identify the right machine learning model and discover the best performing parameters to predict the stock prices accurately.

Stock Returns Key issues :

There were so much research work had been done by researches in the past 15 years to analyse and identify the key factors related to stock returns, as shown below.

Predictability and Forecasting : This has been one of the major research areas in the past several decades, around 25% of the research papers out of 400 are focused on predictability**.**

1. Different models were analysed by researches for better predictability.
2. Avramov (2002)[16] used the Bayesian model which shows the importance of model uncertainty. The paper mentions that, large risks and losses are possible if the investors are not considering this model uncertainty.
3. In order to evaluate the statistical and economic importance of product return predictability, Wei and Zhang (2003)[17] used medium variance analysis. Also, the predictability of returns was calculated not to be linked explicitly to the price of fair properties.
4. Mcmillan (2001)[18] defined different variables that could lead to the estimation of stock prices in the context of financial or microeconomic sources.
5. Duan, Liu and Zeng (2013) [19] mentioned in his paper that, recommendations of behavioral analysis used for new forecasting approach.

Volatility and Variability : Over the last 20 years several studies were performed to study about uncertainty or variability. The analysis showed that the uncertainty or variation in stock profits is the most commonly appeared in journals about 31 % of research papers chosen.

- Umutlu, Akdeniz, and Alttay-Salin (2010) [20] were mentioning in their paper about the impact of financial liberalization in volatility of stock returns. It is evident from their paper that, the increase in financial liberalization when the volatility of stock returns decreases.

Inflation : Across globe many researchers were doing research in the area of understanding the inflation and its impact on global economies.

- Alagidede and Panogiotidis (2012) [21] were mentioning in their paper that, there is a direct relation between the stock returns and inflation and found out that, there is a positive relationship between these two factors.

Risk and Liquidity: Based on the empirical studies, there were not much research work were happening in this domain.

- Chen and Hill (2013) [22] mentioned in their paper that, during the study of stock returns and risk association, they noticed a consistent and coherent relationship between liquidity and stock yield and an essential positive connection between stock returns and liquidity.

3.3 Stock Market Prediction Methodologies

The analysis of market movements relies directly on speculation and stock data trading. The analysis of various stock market prediction technology is mentioned in this Portion. The categorization of various bond sector prediction strategies is shown in the figure.

Prediction based techniques : Various techniques are used for prediction and are classified accordingly as per below figure

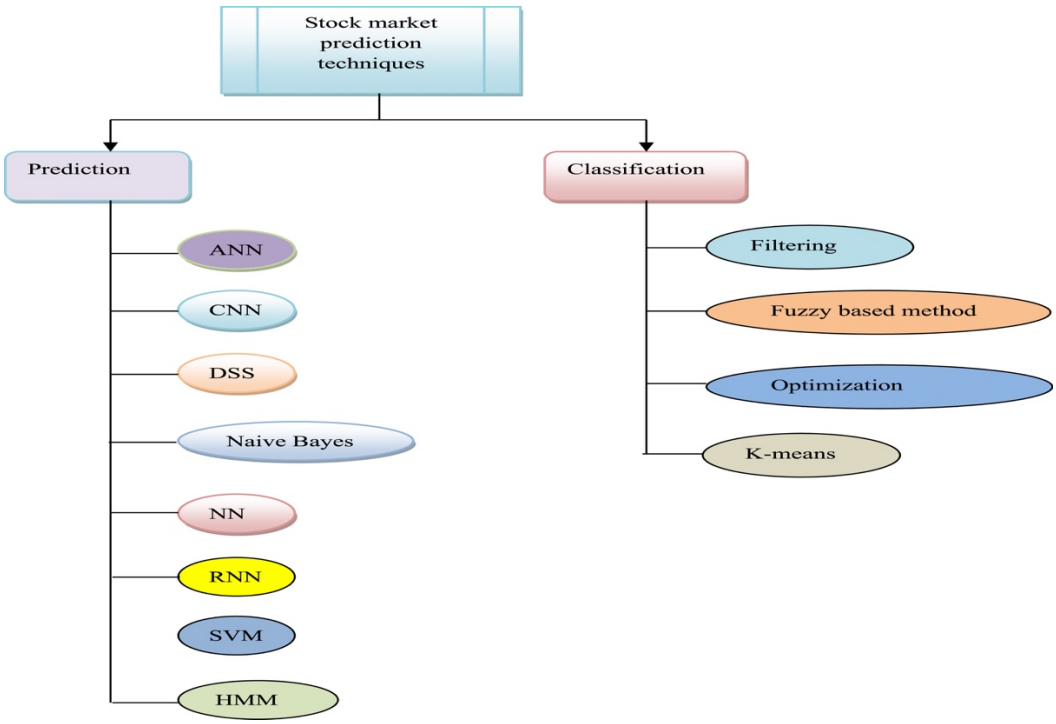


Figure 32 Stock Prediction Methodologies

3.3.1 Stock market prediction Methodologies Classifications:

As per Ticknor, J.L [23] has mentioned that, market trends and technical trends were used as indictors combined as input for predicting the cost of the stock and this model is called as "Bayesian regularized ANN" [23] which is used to forecast the financial market details about behaviour.

Rout, A.K [24] have established an innovative model for calculating time series estimates on the basis of Indian stock indices in the financial industry. This model incorporated the Lowest Middle Square (LMS), exercise function and weights of the neural networks. This model is known as the CEFLANN (CEFLANN) Computationally Efficient Functional Connection.

Zhong, X. and Enke, D [25] have developed three techniques model based on which to effectively mining data, namely "Fuzzy Robust Principal Component Analysis (FRPCA)" [25], "Kernel-based Principal Component Analysis (KPCA)"[25], and predicted market index return patterns on the basis of economic featuring. The findings were contrasted. Three approaches are used to forecast, these include: "Fuzzy Robust Main Component Analysis (FRPCA)"[25],"Kernel-Based Principal Component Analysis (KPCA)"[25] and "Principal Component Analysis (PCA)".[25]

3.3.1.1 CNN based prediction techniques :

"CNN are made up of neurons that have learnable weights and biases. Each neuron receives some inputs, performs a dot product and optionally follows it with a non-linearity"[78]. There are more hidden layers in a CNN than in a conventional neural network. Stock market forecasts are using CCN based deep learning methods.

The model developed by Vargas, M.R. et al., [26] to forecast lateral orientation using S&P's 500 index utilizing profound learning methodologies as a scientific predictor. Owing to deep learning policies that accelerate the trade phase, complicated data structures are analysed

3.3.1.2 HMM-based prediction techniques

Time series or time depending phenomena are well predicted using HMM. This helps in predicting and forecasting stock market.

The paper published by Dhingra, B, Gupta, A. [27] were explaining about the prediction value dependency on stock ethics and based on previous results which is called as Posteriori HMM.

3.3.1.3 Neural Network based prediction techniques :

Chang, P.C. [28] had developed a model that was built to forecast market patterns with feedback as technical indicators that help to refine the EPCNN algorithm for learning and training weightage.

Thomas Fischer and Christopher Krauss have done their research in their paper *"Deep learning with long short-term memory networks for financial market predictions"*. This paper deploys LSTM for predicting S& P 500 stocks and compares the performance of random forest, a deep neural net and a logistic regression classifier.

Pang, X et al. had developed a model to evaluate stocks utilizing the data in real time from the livestock industry. NN based on Deep LSTM and automated encoder were developed to forecast market patterns.

The neuro fuzzy design focused on Elliott Wave Theory has been established by Atsalakis, G.S. et al [30]. That is called the "Wave Analysis Stock Prediction" (WASP) system which is valuable for precise equity price forecasts. This algorithm may be used to forecast the stock market correctly.

Chatzis, S.P.[31] had built the prediction model utilizing a suite of machine learning approaches in which broad collections of data on everyday inventories and a wide variety of economies are chosen.

Shen, W [32] has developed a model based on "Artificial Fish Swarm algorithm (AFSA)" for training the system and forecasting market indexing with the Shanghai Commodity and also the model was designed to use "Radial Basis Neural Network (RBFNN) ".

Asadi, S [33] had developed a cognitive model which is combination of various methodologies such as GAs convergence, data reprocessing processes and the "Levenberg–Marquardt algorithm"[33]. The pre-processing processes of the data involve input variable collection and data transformation in order to boost their accuracy.

3.3.1.4 RNN based prediction techniques

Hsieh, T.J had designed a model based on "Artificial Bee Colony (ABC-RNN)" [35] algorithm for forecasting the stock prices. The first step in this process is to decompose the time series values of stock price and then removing the outliner and noise data from this data by using optimized RNN algorithm with fine-tuned weights and biases as hyper parameters.

Xie, X.K [36] and Wang [36], had designed a model which is primary suitable for intra-day trading using RNN based time series analysis method. Primarily RNN based time series data is used to identify the feature related to daily stock data for intra-day trading purposes. Precision and Average Profit were primarily used as metrics to measure the performance of this methodology.

Chen, W [37] had designed a model to predict the volatility of stocks in the Chinese stock market which is on RNN model along with "Gated Recurrent Units (GRUs)"[37]. The drawback with this model is that having less accuracy, because it doesn't consider the advanced algorithms like "Interdependent Latent Dirichlet Allocation (ILDA)" [37]

3.3.1.5 SVM based prediction techniques

Oztekin[38] had designed a method based on hybrid model of adaptive neuro-fuzzy inference systems along with ANN and SVM. The drawback with this method is that, it failed to consider for broader trading systems.

Zhang, X[39] had designed a model that can be used for predicting the stock indexes based on the news sentiment. This is hybrid model which includes sentiments about the stock, news events of the company. This inclusive framework is more effective in identifying the impact of events and news sentiment on the given stock price and associated fluctuations.

3.3.2 Clustering techniques

3.3.2.1 Filtering based prediction techniques

Arévalo, R [40] had developed a model focused on trading laws that identify flag patterns. The flag design reflects existing patterns and EMA which was subsequently applied to screen the

trades. The downside of this model is that until some trade laws were made, it was not able to fix data snooping problems.

Ariyo, A [41] had designed predictive model based on the NSE) and NYSE stock market provision model ARIMA was created. The findings of ARIMA models are focused on sophisticated forecasting techniques for a short-term prediction.

Srinivasan, P and Ibrahim had established the volatility of the SENSEX index using the Garch model for a certain time. This process showed superior output indicators when forecasting the return of the SENSEX graph.

3.3.2.2 K-means based clustering technique

Ramon Lawrence had utilized neural networks to forecast asset values. This article is an inquiry into the of neural networks in market price forecasting. Neural networks can forecast business direction more effectively than conventional approaches as they can uncover correlations in nonlinear and unstable structures. Different market assessments are addressed and contrasted with neural network results, such as technological analysis, fundamental analysis and regression. In comparison to the chaos theory and networks, the Efficient Market Hypothesis (EMH) is often discussed. Initial recommendations for the usage of neural networks on capital markets[44].

Vivek Rajput, Sarika Bobde [45] had mentioned in their paper "Predictive equity market utilizing hybrid strategy" about a model to predict movement of stocks using data mining and the NSE prediction process. The technique suggested would yield two outputs, one from emotion analysis. A successful prediction is made by analysing both the data. For the analytical study, this paper takes account of stocks with highest capitalization in all primary sectors[45].

Li Xiong, Yue Lu Hybrid Method [46] had proposed a model **Hybrid ARIMA-BPNN Time Series Chinese Stock Market Analysis**. The estimation of asset values is a problem owing to the complexities of time series trends. The Back Propagation Near-Neural (BPNN) and Autoregressive Integrated Moving Average (ARIMA) models are common time-series predictions models. The convergence of both the models (can be linear or non-linear time-series) successfully catches the variations embedded in a time sequence.

There are two variants of this paper: one for CNN financial reporting, and one for SI-RCNN for LSTM, and another for IRNN for LSTM technical indicators. Every model's output is used as the input of a trading agent who buys stocks the day after the model forecasts the rate to increase, then he sells stocks the day before and buys the next day. The proposed procedure indicates that financial news is critical in stabilizing the results and is almost unimproved in comparison with numerous technological indicator sets[46].

Marios Mourelatos, Thomas Amorgianiotis, Christos Alexacos, Spiridon Likothanassis had identified that, dependencies between various financial indices is also to challenge modern approaches. The difficulty and the vast volume of past financial knowledge brought up the need for innovative approaches to the issue. This article proposes a model based on deep learning along with "long-term memory networks"(LSTM) [71] for the purpose of modelling and trading the financial indices [47]

Mohammad Asiful Hossain, Rezaul Karim, Ruppa Thulasiram, Neil D B. Bruce, Yang Wang had developed Dynamic Deep Learning Modes. Most investors and academics have also been conscious of financial market forecasts. Popular ideas say that capital markets are largely a spontaneous move, and that attempting to forecast it is a foolish task. The estimation of product values is an extremely challenging task because of the number of factors involved. Both these issues are discussed in this paper[48].

J.J.Z. Xie, Z. G had published their research paper "Prospective equity index focused on a composite formula". This paper explores the forecast performance of the New York Stock Exchange model of ARIMA and artificial neural networks, using stock details. The empirical findings demonstrate the prevalence of the ARIMA paradigm in neural networks. The results shows conflicting viewpoints on the downside of neural networks and ARIMA model. [49].
A hybrid ARIMA-ANN moving average filter model can be developed to apply on time series-based data projection as per the literature mentioned by C. Babu and B. Narendra. Eswara Reddy.

A brief examination is carried out in general on supervised study and encompasses classical, non-sequential pattern classification framework which comprises of supervised sequence labelling and classes of sequence labelling tasks which come from different label sequence assumption.

P.J Werbos[52] had mentioned in his article, that aims at the general principle of back propagation, a straightforward approach employed widely in fields such as pattern detection and fault diagnostics.

Rudra Kalyan Nayaka, Debahouti Mishraa and Amiya Kumar Rath have been working on stock market pattern reversal study of Indian benchmarking indices using Naïve SVM-KNN-based model. This article suggests a hybridized support system for the Indian stock market indexes (SVM) method with K-Nearest Neighbor method. The aim of this paper is to understand how Indian stock-market using the two indexes BSE Sensex and CNX Nifty predicting closing price.

However, since several factors influence the financial markets, and financial data is time series data it is challenging to find the right moment to purchase or sell stock. Many time series models were therefore introduced for market price predictions, and in comparison, the previous time series approaches do have several problems using "integrated nonlinear feature selection (INFS)"[54] method for stocks [54] for the "Adaptive Neuro Fuzzy Inference System (ANFIS)"[54].

Liang-Ying Wei Deng-Yang Huang Shun-Chuan Ho Jyh-Shyan Lin Hao-En Chueh Chin-Sung Liu Tien-Hwa Ho[56] had proposed a feedback type of the "functional link artificial neural network (FFLANN)"[55] with "recursive least square (RLS) "[55].

Bahdanau and Y. van Merrienboer had mentioned in their paper that addresses the projected success of the New York Stock Exchange model and the artificial neural network models with reported market info. The empirical findings demonstrate the supremacy of the ARIMA paradigm in neural networks. These findings further solve and clarify conflicting views on the accuracy of the neural networks and the ARIMA model in literature[57].

G. Fr Zhang[58] have proposed the paper noticed that an artificial grammatical structure of a sentence is automatically created by the proposed recursive convolution network. [58]. The ARIMA is one of the most common linear models in time series foresees in the past thirty years. Recent research into artificial neural network (ANN) forecasting shows that ANNs may be a successful option to conventional linear approaches. ARIMA and ANN models are sometimes linked to the higher prediction dominance findings

A hybrid model based on both ARIMA and ANN models is suggested which can benefit the intensity of the ARIMA and ANN models. Specific data sets demonstrate that the integrated model can be an efficient way to improve the predictability of each of the separately used models [59]. Predicting stock market values through neural networks utilizing "Gated Recurrent Units (GRUs)"[61].

Predicting stock values is a crucial function in predicting time series, and is of significant importance to creditors, commodity traders and professionals. In recent times various prediction methodologies have been used to forecast the inventory price, including regression algorithms, that can be useful tools to provide good financial time forecasting accuracy. A new hybrid method has been recommended for optimizing the stock price prediction[62]which combines "Support Vector Regression"[62] and Hodrick Prescott filter[62].

Zabir Haider Khan, Tasnim Sharmin Alin, Md. Akter Hussain [63] have used ANN for stock market prediction and have used 5 fundamental input variable which are general index(GI), Net Asset Value(NAV), profit per earning(P/E) ratio, Earnings per share(EPS) and share volume. They have applied these parameters to NN and compared their outcome.

3.3.3 Research gaps and issues

Given the various methodologies used to estimate markets, some constraints need to be overcome in order to achieve an accurate equity market prediction. This segment deals with analysis discrepancies and difficulties with the numerous forecast techniques for the stock market.

Based on stock market forecasts, various challenges associated with NN are identified and documented below. ANN has not been considered an efficient scheme for stock market forecasts because neural models cannot handle high overhead computational costs due to the large neurons due to the correct adaptation of weight[64]. In [65], the NN administered both the assessments and the preparation at a diminished rate, impacting the efficiency of the forecast. In addition, the bypass that can be done using NN, which is stuck in the central minima and blackbox technique

The findings obtained by the stock predictive method established by NN[51]]were not optimized with accuracy due to the effect of the pattern misclassification and the usage of network parameters. The problem with the CNN stock forecasting approach is that, it was not ideal for incredibly comprehensive implementations. The degree of effective identification obtained by CNN was lower than the other state-of-the-art market forecast system[66]

In order to build an effective stock expert system on acquisition, the established decision-making support system [67]did not use the realistic information and methods gathered. In order to develop an effective model, ANN needed long-term preparation and had little reason to decide a solution[61].

A method of selecting features was built in SVM to forecast stock market patterns. The feature selection process did not specify the number of ideal features needed, thereby significantly influencing the accuracy of the program. The prediction method developed for SVM and NN relies on the correlation value of the feature chosen[63].

The study has determined that more academic publications were released in 2011. ANN was used to predict market trends in most papers, while fuzzy systems are used to aggregate inventory data. For forecasting market patterns in stock prediction systems, MATLAB is the most popular software method. Various index data are the most commonly used data sets for stock market prediction. Other data sets are also used for stock market analysis, such as Nasdaq, Istanbul stock bond, Teheran stock index.

The various drawback observed in the above literature review are primarily related to high volatility and the impact of news or sentiment data on stock price index. This has significant impact on the stock price movement , hence addressing this concern/drawback is the major motivation behind this thesis. This thesis describes the hybrid approach by consuming stock data along with news/sentiment data while forecasting stock price. This helps in improving the accuracy of the stock prediction model.

4 Research Methodology

4.1 Introduction: Data acquisition and pre-processing is probably the hardest part of most machine learning projects.

4.2 The Learning Environment:

The Weka and YALE Data Mining Environments were used for carrying out the experiments. The general setup used is as follows:

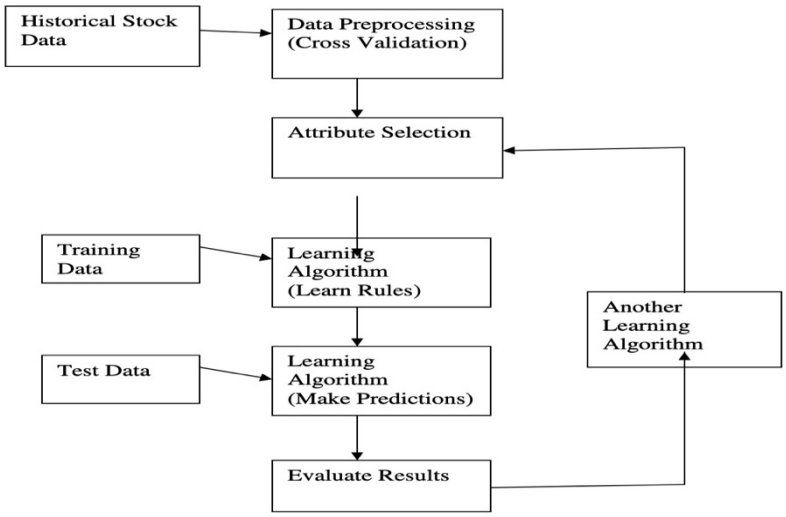


Figure 33 Historical Data & Training Data Set Learning Algorithm[69]

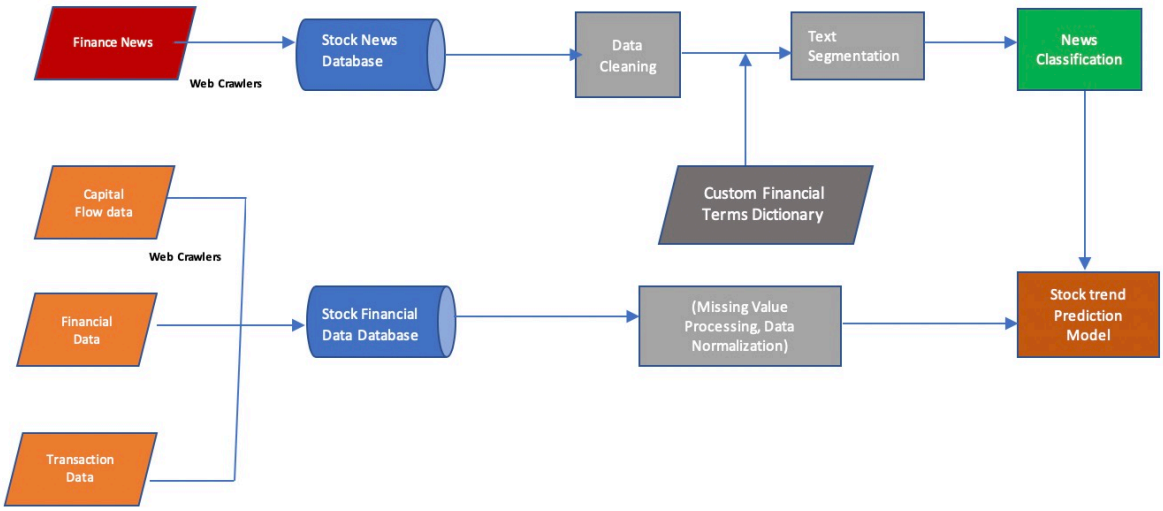


Figure 34 Stock Data Processing Pipeline

4.3 Data Gathering:

- **Stock indices:** As expected, several analysts forecast aggregate index market prices rather than estimate stock prices of specific firms. It represents the average shift of bond rates of benchmark indices.

- **Historical data**

  1. **Historical stock fundamentals:** Historical fundamental data is actually very difficult to find (for free, at least). Although sites like Quandl do have datasets available as paid service. Pre-process the historical price data

  2. **Creating the training dataset:** Ultimate goal for the training data is to have a 'snapshot' of a particular stock's fundamentals at a particular time, and the corresponding subsequent annual performance of the stock. Thus, algorithm can learn how the fundamentals impact the annual change in the stock price.

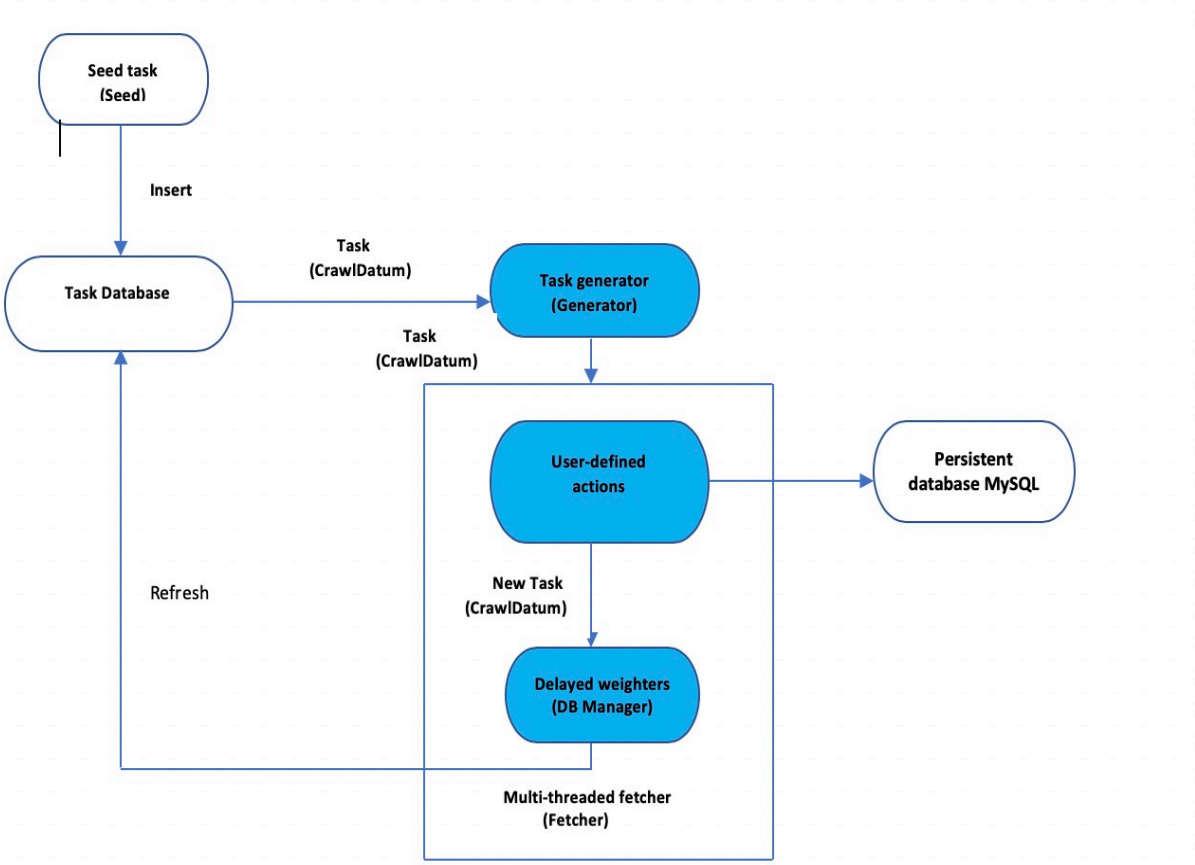


Figure 35 Web Crawler Diagram

4.4 Data Processing:

4.4.1 Article Filtering:

Reports retrieved from the API news collection include data in the section-represented categories. Any pages include other types of papers unrelated to stocks, such as genetics, notes of origin, schedule etc. Consequently, these posts have been deleted from such websites.

4.4.2 Merge stock indices with articles:

- The overall workflow to use machine learning to make stocks prediction is as follows:
- Acquire historical fundamental data – these are the features or predictors
- Acquire historical stock price data – this is will make up the dependent variable, or label
- Pre-process data
- Use a machine learning model to learn from the data
- Backtest the performance of the machine learning model
- Acquire current fundamental data
- Generate predictions from current fundamental data

4.5 Data Processing and Wrangling:

4.5.1 Data Collection:

Tweets was derived from the twitter Database on Apple, Google, and AAPL. The public's view on the stock of the company and the views on the company's goods and services. Keywords are specifically crafted for filtering and tweets collected such that over a span of time they reflect the exact feeling of the public towards Microsoft. News of Microsoft and messages on Twitter will also be found in new updates.

4.5.2 Data Pre-Processing:

The statistics obtained on equity markets was nonsensical as the economy does not operate during holidays. A basic method is implemented to estimate the missing details. Typically stock

data follows a concave structure. If then, the inventory value is x on a day and the corresponding value is y, which is absent. The first missed value is approx. (y+x)/2 and all holes are filled by the same formula.

4.5.3 Data Clean-up:

Tweets contain multiple acronyms, emoticons and redundant details such as images and URLs. Tweets are then prepared to reflect the best public emotions.

- **Tokenization:** Tweets are broken up and symbols are omitted such as emoticons, space and meaningful individual words will be used in future
- **Stop word Removal:** Terms such as a, is, and with etc. are then removed from the word list after breaking a message.
- **Regex Matching for special character Removal**

4.6 Sentiment Analysis:

The role of sentiment analysis is quite unique to the field. Tweets focused on the present mood are graded as optimistic, negative and neutral. For Optimistic, 0 for Neutral and 2 for Harmful Feelings, tweets are annotated as 1. A machine learning algorithm is equipped for the interpretation of the annotated non-human messages, which is derived from human messages.

4.6.1 Feature Extraction: Textual representations can be done using n-grams.

N-gram Representation:

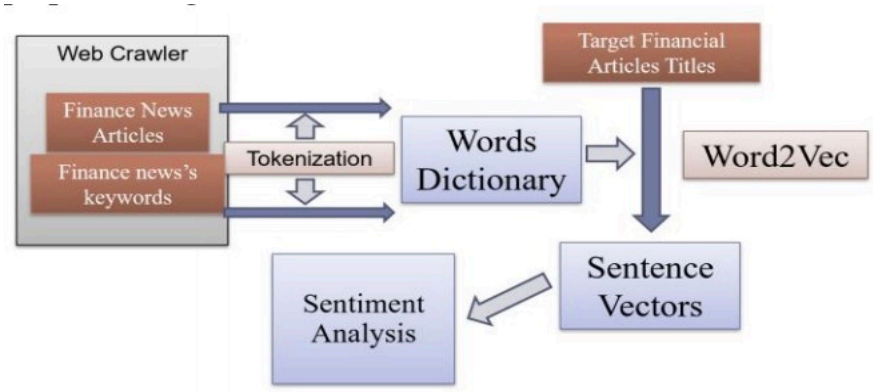


Figure 36 Sentiment Analysis Overview Diagram

4.6.2 Stock Trend Prediction

LSTM is also used to build trend prediction model. Sentiment analysis results + stock data are input of LSTM Model [70]. After feature selection and sentiment analysis, we got stock data at day t and sentiment label at day t. Combine them as the input. Output is trend prediction result -1/0/1 representing negative/even/positive.

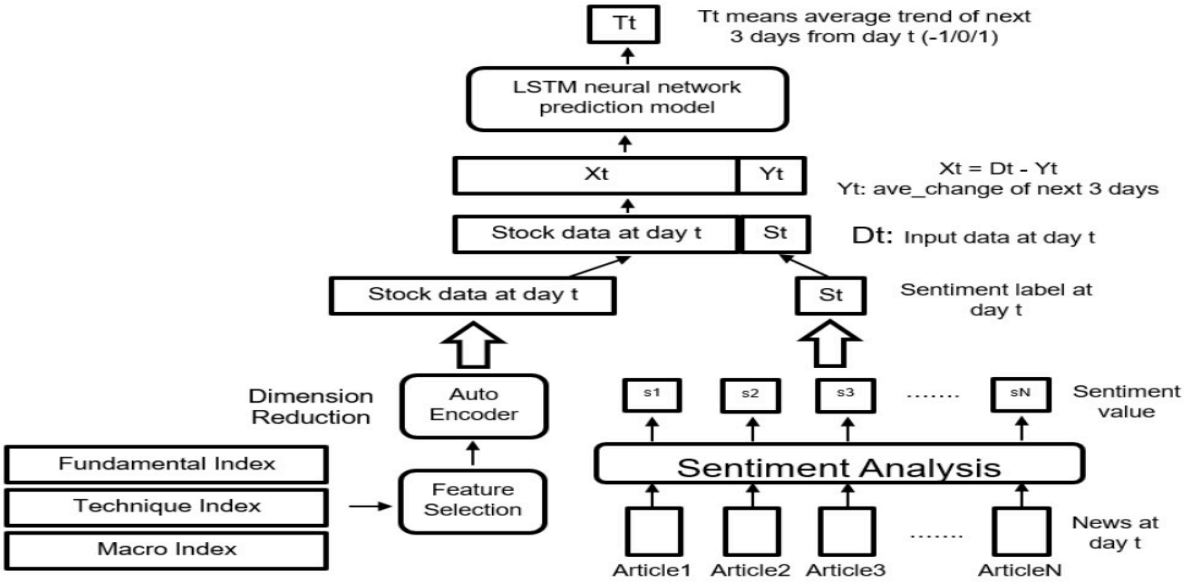


Figure 37 LSTM Model for Stock Index Processing with Sentiment Analysis[15]

4.6.3 Model Training:

Features extracted by using the above methods for tweets are fed into the classification and trained to estimate the movement of market price change versus volume and feeling of news and tweets. Apply linear regression to find a correlation between market price shift vs volume as well as news and tweets feelings.

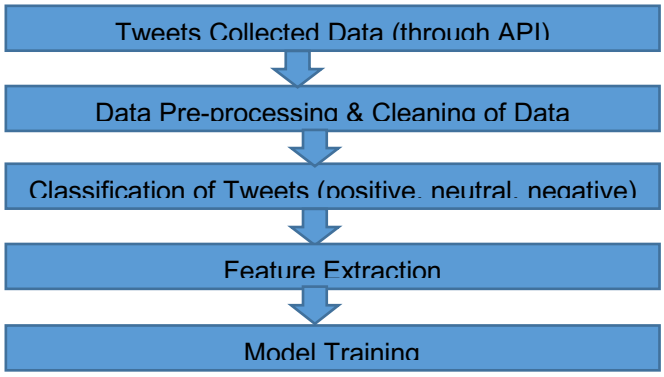


Figure 38 Twitter Sentiment Analysis

4.6.4 Experimental Design Dataset:

Tweets data and News articles from Bloomberg or any channel and Stock Information

4.6.5 Evaluation Measures

a) Measure correlation between stock price movement and tweets

b) Mean Squared Error for Linear Regression Model

c) Loss function and accuracy percentage for Classification model

4.7 Data Processing System Design:

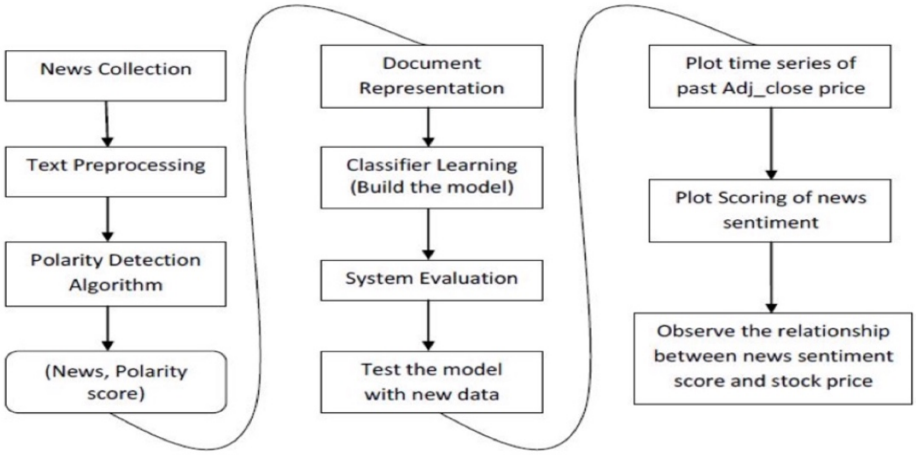


Figure 39 News Processing and Plot sentiment Scoring [13]

4.7.1 Data Sources

4.7.1.1 Historical Stock Prices Data

The dataset contains following fields: Date, Open, High, Low, Close and Volume.

Historical data sets of stock prices can only obtain per day at the minimum from Yahoo Finance otherwise it would have to be streamed from directly from the NASDAQ website, which I did not have the access to. Ideally hourly stock prices would have worked by matching the time series with the Twitter feeds. Data sets of stock prices were collected from the Yahoo Finance website for all three companies. Each set had seven columns consisting of Date, Open, High, Low, Close, Volume and Adjusted Close.

- Date is the day of trading.
- Open is the opening price of the stock at the start of the day's trading.
- High is the highest price of the stock form that day.
- Low is the lowest price of the stock from that day.
- Close is the closing price of the stock at the end of the day's trading.
- Volume the number of shares traded that day.
- Adjusted Close is the after-trading hours price. The difference between the open and close price.

First | Previous | Next | Last

Prices

Date	Open	High	Low	Close	Volume	Adj Close*
Apr 30, 2014	592.64	599.43	589.80	590.09	16,308,600	586.81
Apr 29, 2014	593.74	595.98	589.51	592.33	12,049,200	589.04
Apr 28, 2014	572.80	595.75	572.55	594.09	23,910,200	590.79
Apr 25, 2014	564.53	571.99	563.96	571.94	13,938,400	568.76
Apr 24, 2014	568.21	570.00	560.73	567.77	27,139,700	564.62
Apr 23, 2014	529.06	531.13	524.45	524.75	14,105,000	521.84
Apr 22, 2014	528.31	531.83	526.50	531.70	7,234,400	528.75
Apr 21, 2014	525.34	532.14	523.96	531.17	6,519,600	528.22
Apr 17, 2014	520.00	527.76	519.20	524.94	10,158,100	522.02
Apr 16, 2014	518.05	521.09	514.14	519.01	7,670,200	516.13
Apr 15, 2014	520.27	521.64	511.33	517.96	9,517,500	515.08
Apr 14, 2014	521.90	522.16	517.21	521.68	7,345,500	518.78
Apr 11, 2014	519.00	522.83	517.14	519.61	9,704,200	516.72
Apr 10, 2014	530.68	532.24	523.17	523.48	8,559,000	520.57

Figure 40 Sample Historical Data

4.7.1.2 Twitter Data

- Twitter data from Internet Archive collection of twitter stream.
- It contains the tweets in a nested structure as: Year → Month → Date → Hour → Minute.
- The dataset contains the fields as shown below: contributors, coordinates, created_at, delete, display_text_range, entities, extended_entities, extended_tweet, favorite_count, favorited,

filter_level, geo, id, id_str, reply_count, retweet_count, retweeted, retweeted_status, source, text, timestamp_ms, truncated, user etc.,

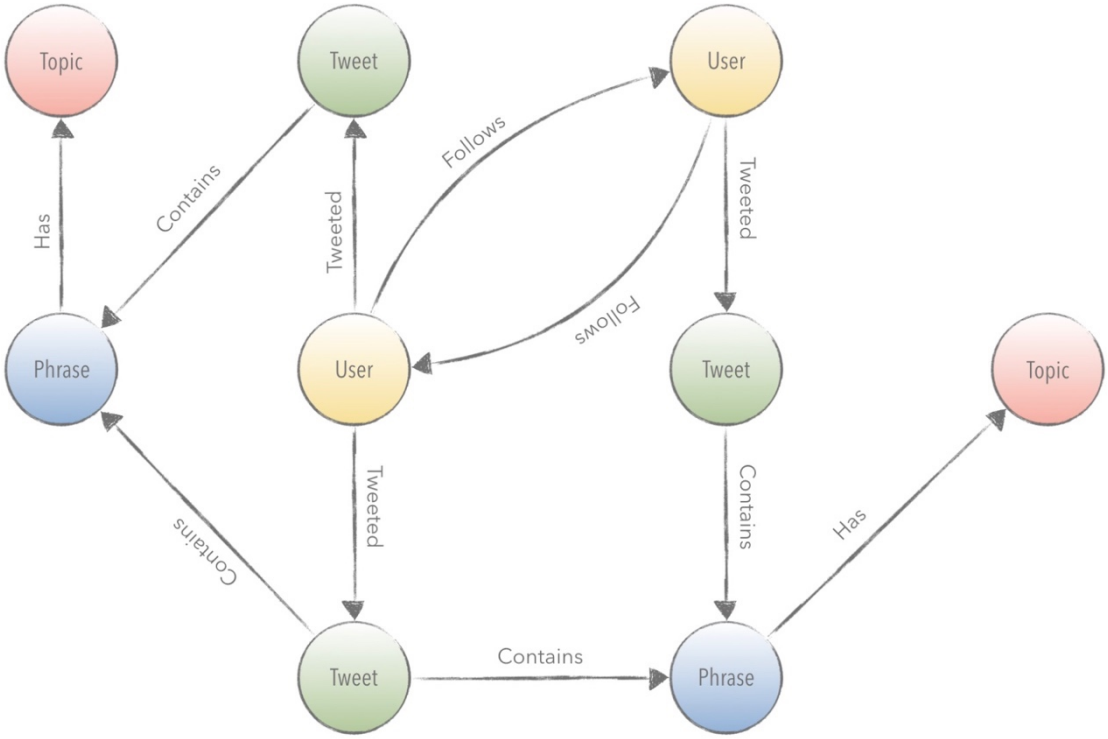


Figure 41 Sample Tweet Graph Model

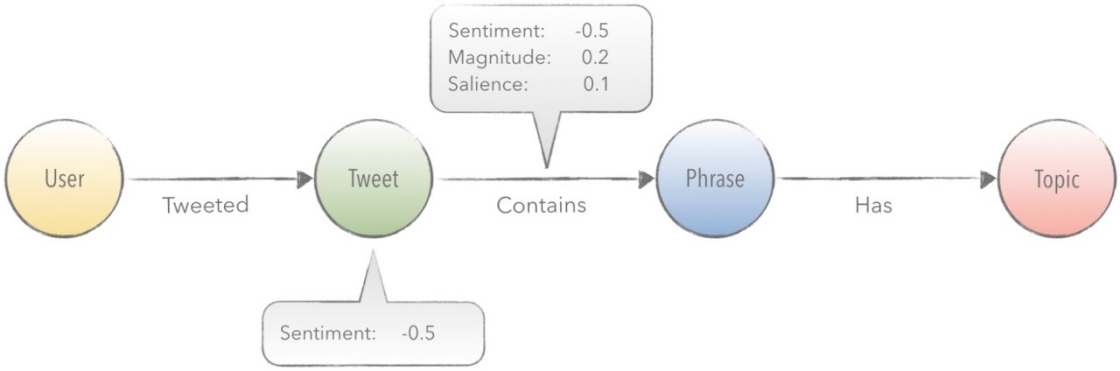


Figure 42 Twitter Use Case Overview

```
def retrieving_tweets_polarity(symbol):
    auth = tweepy.OAuthHandler(ct.consumer_key, ct.consumer_secret)
    auth.set_access_token(ct.access_token, ct.access_token_secret)
    user = tweepy.API(auth)

    tweets = tweepy.Cursor(user.search, q=str(symbol), tweet_mode='extended', lang='en').items(ct.num_of_tweets)

    tweet_list = []
    global_polarity = 0
    for tweet in tweets:
        tw = tweet.full_text
        blob = TextBlob(tw)
        polarity = 0
        for sentence in blob.sentences:
            polarity += sentence.sentiment.polarity
            global_polarity += sentence.sentiment.polarity
        tweet_list.append(Tweet(tw, polarity))

    global_polarity = global_polarity / len(tweet_list)
    return global_polarity
```

Figure 43 Sample Code to read tweet data

4.7.1.3 News Data:

- News Dataset from Kaggle News Category Dataset and Kaggle US Financial News Articles.
- Kaggle News Category Dataset contains approximately 200k news articles of various categories
- The fields in this dataset include date, authors, category, headline, short description and link.
- The news articles are from the following news publishers: Bloomberg.com, CNBC.com, reuters.com, wsj.com, fortune.com.

4.7.1.4 Stock Data

Generally, adjusted prices for stock market prediction are used because normal prices might have other influences such as stock splitting which may lead to a poorly performing model.

Historical data can be extracted by using investpy, which is mainly intended for historical data extraction.

```
import investpy

df = investpy.get_stock_recent_data(stock='BBVA',
                                    country='spain')
print(df.head())
```

```
             Open   High    Low  Close    Volume Currency
Date
2019-08-13  4.263  4.395  4.230  4.353  27250000      EUR
2019-08-14  4.322  4.325  4.215  4.244  36890000      EUR
2019-08-15  4.281  4.298  4.187  4.234  21340000      EUR
2019-08-16  4.234  4.375  4.208  4.365  46080000      EUR
2019-08-19  4.396  4.425  4.269  4.269  18950000      EUR
```

```
import investpy

df = investpy.get_stock_historical_data(stock='BBVA',
                                        country='spain',
                                        from_date='01/01/2010',
                                        to_date='01/01/2019')
print(df.head())
```

```
             Open   High    Low  Close  Volume Currency
Date
2010-01-04  12.73  12.96  12.73  12.96       0      EUR
2010-01-05  13.00  13.11  12.97  13.09       0      EUR
2010-01-06  13.03  13.17  13.02  13.12       0      EUR
2010-01-07  13.02  13.11  12.93  13.05       0      EUR
2010-01-08  13.12  13.22  13.04  13.18       0      EUR
```

Figure 44 Investpy Historical Data

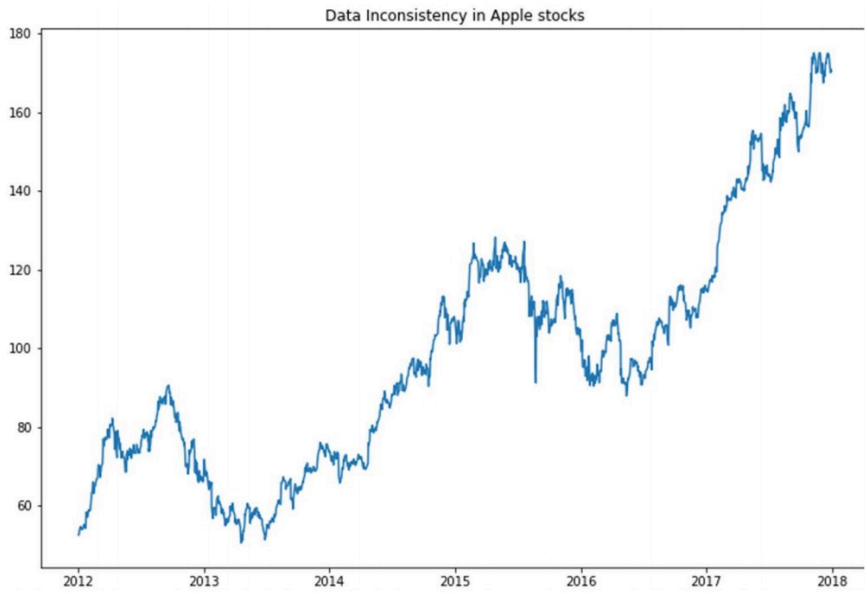


Figure 45 Apple Stock Yearly chart

4.7.1.5 News Data (Count of articles based on News Categories)

The Bar Chart below shows the count of articles by each category for the Kaggle News Category Dataset.

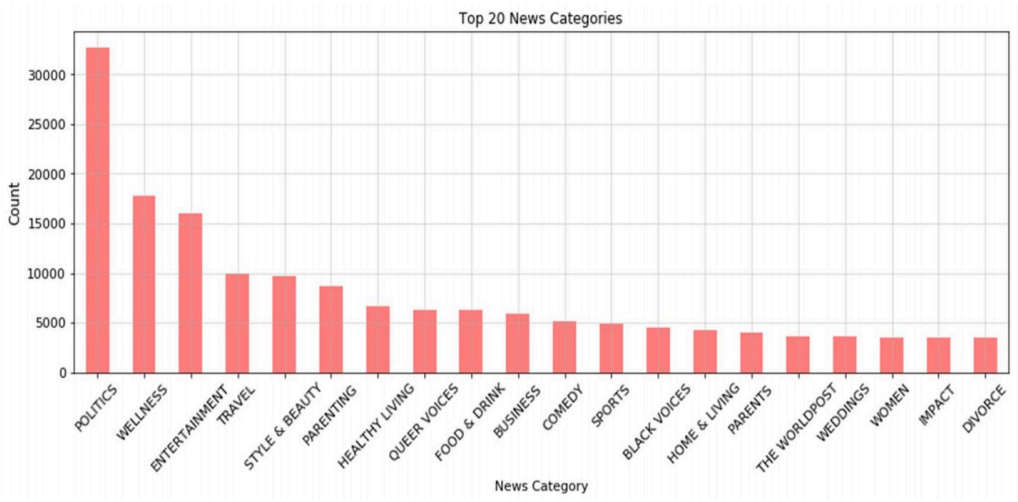


Figure 46 Top 20 News Catergories

4.7.2 Exploratory Data Analysis (EDA)

4.7.2.1 Data Pre-Processing:

- Text data is unstructured data.
- Tokenize the document into words to operate on word level, followed by removal of noisy words and numbers stop words and punctuation.
- Using polarity words dictionaries calculate the score of that document using general words with positive and negative polarity.

4.7.2.2 Algorithm:

- Tokenize the document into word vector.
- Prepare the dictionary which contains words with its polarity (positive or negative)
- Check against each word weather it matches with one of the words from positive word dictionary or negative words dictionary.

- Count number of words belongs to positive and negative polarity.
- Calculate Score of document = count (pos.matches) – count (neg.matches)
- If the Score is 0 or more, we consider the document is positive or else, negative.

4.7.2.3 Twitter Data – Pre-Processing

- Processed the twitter data to extract sentiments for stocks (for ex Apple and Facebook).
- **The data cleaning process is as follows:**
    - Remove unwanted fields, **Fields Used :** 'Date', 'screen_name', 'followers_count', 'text'.
    - Convert the timestamp (ms) in to Date and Time.
    - Replace the null values with 0 or empty string values so that the data can be processed further.
    - Clean tweets to remove unwanted texts, characters, symbols and emojis using regular expressions.
    - Filter tweets to retain only those tweets containing only words from APPLE_Word_List and joint_words and tweets not containing words in the APPLE_unwanted_list. This step ensures that we only get tweets that are related to Apple technologies and not apple the fruit.

APPLE_Word_List = ['apple','aapl', 'airpod', 'app store', 'earpods', 'homepod', 'imac', 'itablet', 'magic trackpad', 'magic mouse', 'iphone', 'ipad', 'ipod', 'macbook', 'ios','itunes', 'iwatch'] joint_words = ['steve jobs', 'tim cook']

APPLE_unwanted_list = ['fruit', 'cherries', 'berries', 'berry', 'food', 'pie', 'nutrition', 'pear', 'citrus', 'mango', grape', 'peach', 'apricot', 'mango', 'plum', 'avocado', 'ripe', 'melon', 'cherimoya', 'tropical', 'pineapple']

1. The tweets' text was then passed to a function to extract the sentiments polarity (-1 to 1)

2. We then performed multiple aggregations by Date to generate the following features:

    a. Median_sentiment

    b. Count_tweet

    c. Avg_sentiment

    d. Sum_followers_count

    e. Max_followers_count

    f. Sum_favourite_count

    g. Sum_retweet_count

    h. Avg_weighted_sentiment

        i. Rather than using a plain average sentiment we put more weights on tweets with a higher follower count to put more importance on tweets that had a higher audience exposure.

        ii. Sum_weighted_sentiment.

4.7.2.4 News Data – Pre-Processing:

- Data clean-up process (as similar as above), identify the text and check for words in the word list and assign 'AAPL' for Apple related and 'FB' for Facebook related articles.
- Pass the body text into a function to extract sentiments.
- Retain only the relevant news articles i.e. 'AAPL' for Apple and 'FB' for Facebook
- Carry out aggregations by Date to generate useful features:

a. Median_sentiment b. Count_news c. Avg_sentiment d. Sum_company_frequency e. Max_company frequency f. Sum_weighted_sentiment g. Avg_weighted_sentiment The output data was then saved as csv files.

4.7.2.5 Stock Data – Pre-Processing

Since the data was already pre-filtered with important features, there was no need for any pre-processing. Both Apple and Facebook's stock data was stored in separate .csv files.

Sentiment Analysis :

- sentiment analysis on our news data sets (Huffington Post, Wall Street Journal, Bloomberg, Reuters, CNBC) as well as the twitter dataset.

- sentiment polarity function is used to obtain a sentiment score by passing the whole body of a news article or the whole text of a tweet through TextBlob's API. The resulting score lies within a range from –1 to +1 and has the following meaning:

Score Range	Sentiment	Comment
- 1 to 0	negative	a lower score means a more negative sentiment
0	neutral	neither negative nor positive sentiment
0 to 1	positive	a higher score means a more positive sentiment

Table 7 Sentiment Scores based on comment

4.7.3 Feature Engineering: Three datasets with the following features as shown in the table:

Identify the feature which can be added to the list , by correlated with the closing prices, keep the feature list in the final vector if that correlates.

Dataset	Number of Features	Features List
News	6	date, authors, category, headline, short description and link.
Twitter	37	contributors, coordinates, created_at, delete, display_text_range, entities, extended_entities, extended_tweet, favorite_count, favorited, filter_level, geo, id, id_str, in_reply_to_screen_name, in_reply_to_status_id, in_reply_to_status_id_str, in_reply_to_user_id, in_reply_to_user_id_str, is_quote_status, lang, place, possibly_sensitive, quote_count, quoted_status, quoted_status_id,

		quoted_status_id_str, quoted_status_permalink, etc
Stock	5	Open, High, Low, Close, Volume

Table 8 Feature Engineering with Number of features

4.7.4 Correlation Analysis

4.7.4.1 Historic Stock Data

1. **VWAP (Volume Weighted Average Price):** Calculated by adding up the dollars traded for every transaction (price multiplied by the number of shares traded) and dividing by the total shares traded.

$$VWAP = \Sigma \text{ Price x Volume} / \Sigma \text{ Volume}$$

2. **DOW (Day of Week):** The last trading day of week may have certain behavior that is specific to that particular day.

3. **US/CAD Exchange Rate :** Since NASDAQ listed company stocks showed positive correlation between US dollar price and closing price.

4. **Running Difference** : Running difference is simply high price of a stock on a date minus the lowest price of that stock on that day. This feature captures the variation of price per day.

Spearson's correlation with Close Price	**Correlation score**	**Correlation meaning / direction**
DOW (Day of Week)	-0.001226	very weak - negative
High	0.999722	very strong - positive
Low	0.999744	very strong - positive
Open	0.999425	very strong - positive
Volume	0.204484	weak - positive
Running Difference	0.343693	weak - positive
US/CAD exchange rate	0.720408	strong - positive

VWAP(Volume Weighted Average Price)	0.856878	very strong - positive

Table 9 Spearson's correlation for Historic Stock Data

4.7.4.2 News Data

5. **Count News** : count of the number of news for a company (AAPL) grouped by date.

6. **Avg_Sentiment:** Average of the body sentiments for 1 company (AAPL) grouped by day.

Spearson's correlation (Close Price)	**Correlation score**	**Correlation meaning**
news_avg_sentiment	-0.003421	very weak - negative
count_news	-0.199476	very weak - negative

Table 10 Spearson's correlation for News Data

4.7.4.3 Twitter Data

7. **Tweet_Count :** Count of the number of tweets for 1 company grouped by date.

8. **Tweet_Avg_Sentiment :** Average sentiment of tweets for AAPL grouped by date.

Spearson's correlation (Close Price)	**Correlation score**	**Correlation meaning**
tweet_count	-0.424871	moderate - negative
twitter_avg_sentiment	-0.086189	very weak - negative

Table 11 Spearson's correlation for Twitter Data

Finally, the following fields as features are used for the model: Using the above features, we predict the Closing Stock Price.

Dataset	**Number of Features**	**Features**
News	2	news_avg_sentiment', 'count_news'.

Twitter	3	'tweet_count', 'twitter_sum_followers_count', 'twitter_avg_sentiment'.
Stock	9	'DOW', 'Date', 'High', 'Low', 'Open', 'Running Difference', 'US/CAD', 'VWAP', 'Volume'.

Table 12 Dataset with Features Details

4.7.4.4 Classifier Learning:

Hybrid models which comprises of various text classification and text summarization methodologies using NB Classifier, Random Forrest which helps in improving accuracy in text classification and text summarization. The factors like accuracy, precision, recall and other model evaluation methods are used to analyse to identify the best possible text classification and text summarization.

4.7.4.5 Graph Analysis

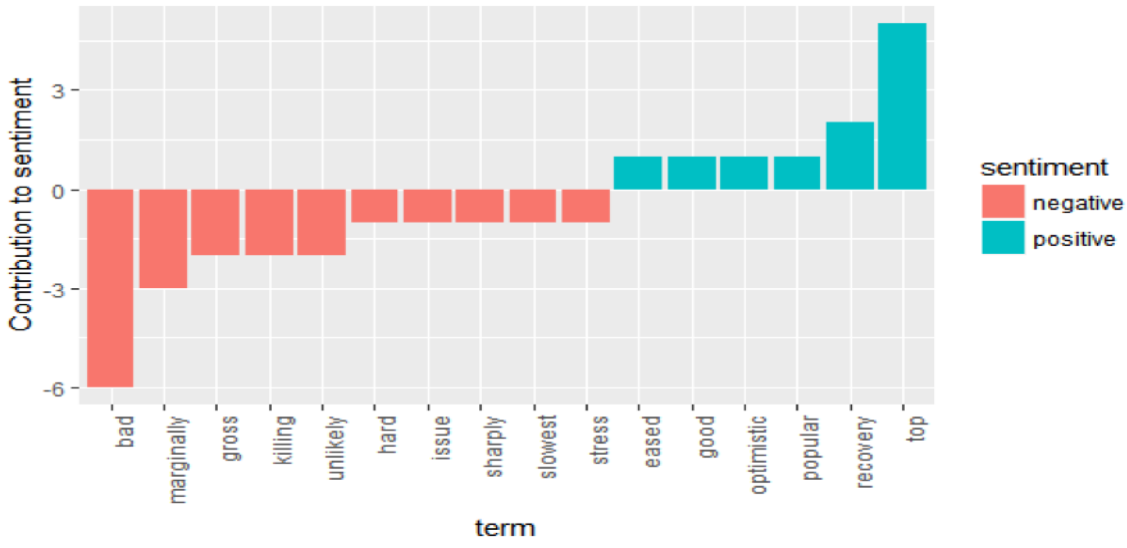


Figure 47 Graph Analysis of sentiments

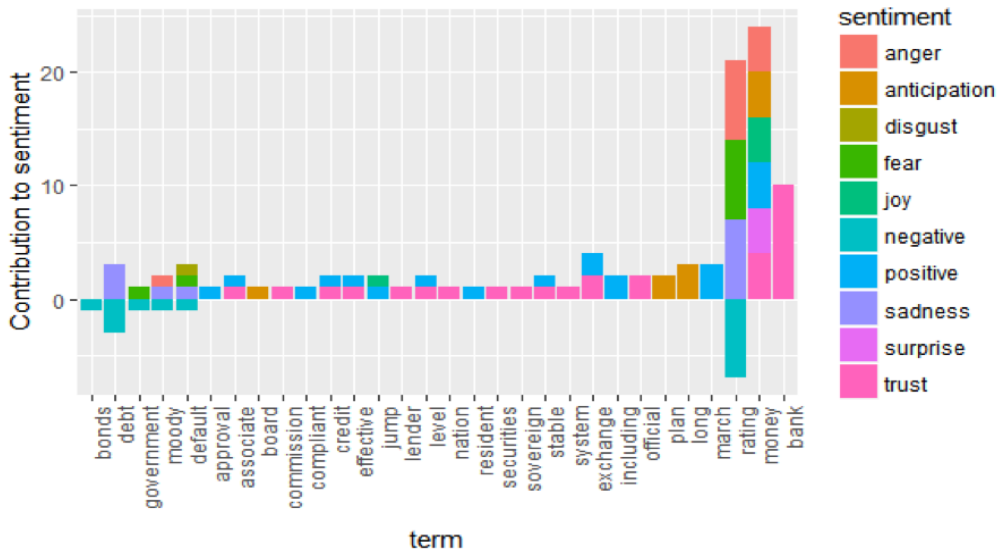


Figure 48 Sentiment Classification and subjective measure

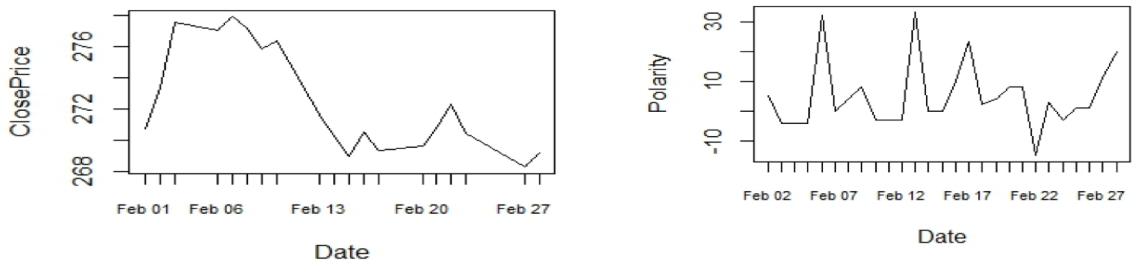


Figure 49 Polarity Analysis

5 Models and Strategies

For each of the subsequent sections there are two subsections each for the two projects we aim to document, namely

- Kaggle-NYS-Estock-prediction
- Research Paper Implementation: Baek and Ha Young Kim.

5.1 Data

- The financial data is an indicator of business performance and the efficiency of strategies/tactics applied in the business.
- It consists of sets of information related to the financial status of a business such as the price of shares of the company.
- "New York Stock Exchange" kaggle dataset which consists of historical prices and fundamental data about the S& P 500 companies.

Contents:

prices.csv: original, unaltered daily prices.

prices-split-adjusted.csv: same as prices with added adjustment for stock splits.

securities.csv: general information about each company with division on its sectors

fundamentals.csv: metrics extracted from annual SEC 10K filings (2012- 2016).

```
df = pd.read_csv("../input/prices-split-adjusted.csv", index_col = 0)
df["adj close"] = df.close # Moving close to the last column
df.drop(['close'], 1, inplace=True) # Moving close to the last column
df.head()
```

	symbol	open	low	high	volume	adj close
date						
2016-01-05	WLTW	123.430000	122.309998	126.250000	2163600.0	125.839996
2016-01-06	WLTW	125.239998	119.940002	125.540001	2386400.0	119.980003
2016-01-07	WLTW	116.379997	114.930000	119.739998	2489500.0	114.949997
2016-01-08	WLTW	115.480003	113.500000	117.440002	2006300.0	116.620003
2016-01-11	WLTW	117.010002	114.089996	117.330002	1408600.0	114.970001

Figure 50 Price Adjusted Dataframe

5.2 The SingleNet model:

```
snp_data.head()
```

	Adj Close
Date	
2000-01-04	1399.420044
2000-01-05	1402.109985
2000-01-06	1403.449951
2000-01-07	1441.469971
2000-01-10	1457.599976

```
d
```

Ticker	AAPL	ADBE	AMGN	AMZN	ASML
Date					
2000-01-04	2.440975	14.791295	48.689911	81.937500	26.826624
2000-01-05	2.476697	15.083735	50.365246	69.750000	26.085730
2000-01-06	2.262367	15.206868	51.202930	65.562500	24.480459
2000-01-07	2.369532	15.945663	56.961960	69.562500	25.036133
2000-01-10	2.327857	16.561329	60.417358	69.187500	27.413177
2000-01-11	2.208785	15.422350	55.653084	66.750000	26.517927

Figure 51SingleNet Data Model Smple Output

```
d.corrwith(snp_data['Adj Close'], axis=0, drop=False).sort_values(ascending = False)
```

```
Ticker
ASML    0.939346
ADBE    0.931655
AMGN    0.920196
MSFT    0.920161
FISV    0.915146
SBUX    0.913389
CELG    0.913010
INTU    0.909671
AAPL    0.899090
REGN    0.898842
```

```
from scipy.stats.stats import pearsonr
for i in d.columns:
  print(i, pearsonr(d[i], snp_data['Adj Close']))
```

```
ASML (0.9393456110059639, 0.0)
ADBE (0.9316554196208873, 0.0)
AMGN (0.9201959633264232, 0.0)
MSFT (0.9201605242557374, 0.0)
FISV (0.9151462492481346, 0.0)
SBUX (0.9133890277443993, 0.0)
CELG (0.9130104913548708, 0.0)
INTU (0.9096707807385662, 0.0)
AAPL (0.8990895194220256, 0.0)
REGN (0.8988415016799999, 0.0)
```

Figure 52 Correlation Index Output

Uses only the historical closing prices of S& P index. The closing prices of these companies are all highly correlated with the S& P500 stock market index. greater than 0.8 were selected (under 5 percent significance level and all p-values less than 0.05).

5.3 Model Architecture

- After normalization the data is divided into consecutive windows of 22 days and the total data is then split between train and test sets by allocating 90% to the train and 10% to test data.

- The training data is further split 10:90 to give 10% validation data and the remaining 90 for training.

- The model consists of two LSTM layers with 256 units each with the first layer returning outputs at each time step (return_sequence = True) and the second returning only the final output (return_sequence = False). Other parameters used with the model include Batch size, Epochs, Dropout rate, Validation split, Optimizer, Loss (Mean Square Error Metrics). Here X is the original value at data point, Xmax is minimum value of the feature (closing price) that X belongs to and Xmin is the minimum of that feature correspondingly. Consequently, every feature (closing price) can be translated in the range [0,1]. The normalized data is then fed into a model. The paper specifies three models with following configurations:

$$\bar{X} = \frac{X - X_{min}}{X_{max} - X_{min}}.$$

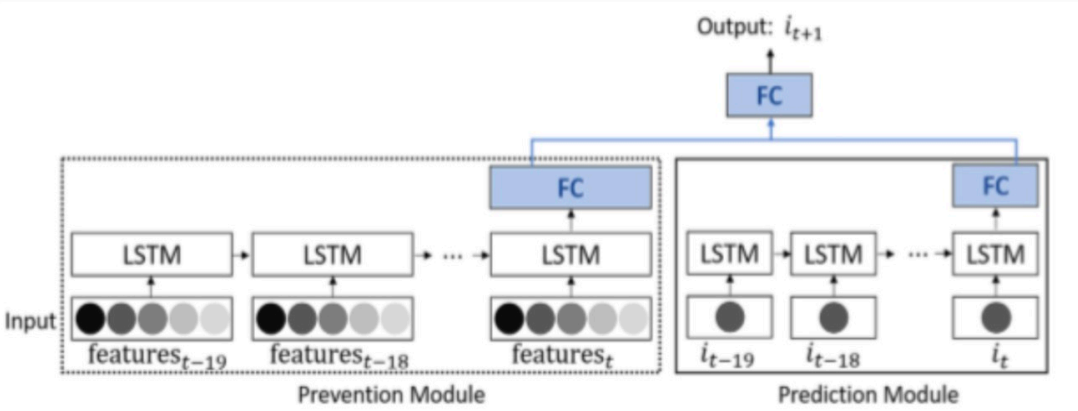


Figure 53 LSTM Prevention. And Prediction Module Diagram

Mean squared error	$\mathrm{MSE} = \frac{1}{n}\sum_{t=1}^{n} e_t^2$
Root mean squared error	$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{t=1}^{n} e_t^2}$
Mean absolute error	$\mathrm{MAE} = \frac{1}{n}\sum_{t=1}^{n} \lvert e_t \rvert$
Mean absolute percentage error	$\mathrm{MAPE} = \frac{100\%}{n}\sum_{t=1}^{n} \left\lvert \frac{e_t}{y_t} \right\rvert$

Figure 54 MSE, RMSE, MAE, MAPE Formulas

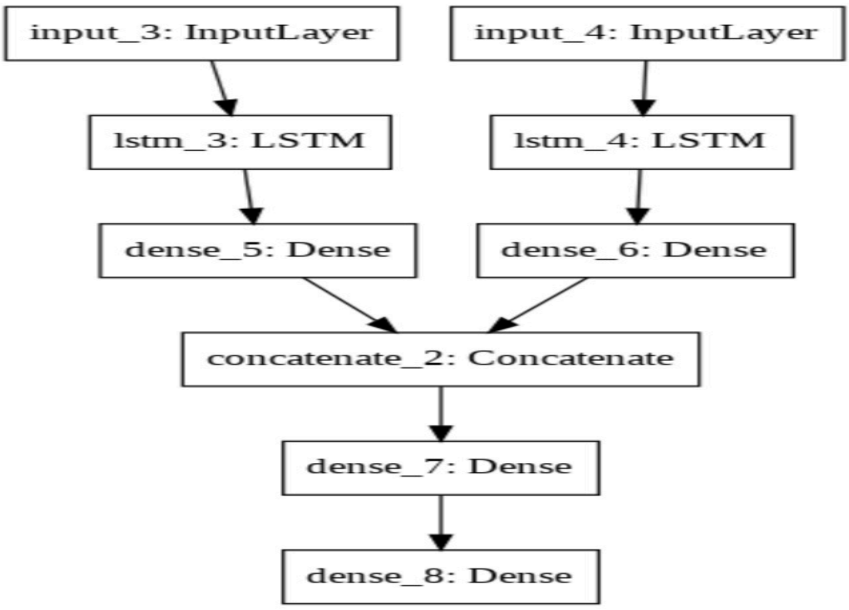


Figure 55 LSTM Model for Dense Processing

6 Software Tools

Below list of software libraries are tools used as part of this thesis are.

S.No	Library	Description
1	Keras	Open Source Neural Network library on top of Theano or Tensorflow. Keras is high-level API wrapper for the low-level API, capable of running on top of TensorFlow, CNTK, or Theano. Keras High-Level API handles the way we make models, defining layers, or set up multiple input-output models.
2	TensorFlow	It's a deep learning library. Tensorflow library incorporates different API to built at scale deep learning architecture like CNN or RNN. TensorFlow is based on graph computation
3	Python	
4	Twitter API	
5	Investpy API	To extract historical stock data
6	Quandl API	For downloading historical stock prices
7	NLTK	NLTK is a leading platform for building Python programs to work with human language data. It provides easy-to-use interfaces along with lexical resources such as WordNet, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning, wrappers for industrial-strength NLP libraries.

7 Implementation Details:

7.1 LSTM Network for Sentiment Analysis:

1. LSTM networks are proven to be effective for NLP applications for text data processing (Sundermeyer et Al., 2012)[71] and machine translation (Cho et Al., 2014)[72].
2. News data are always text based rather than numerical
3. Convolution layer has to be applied in order to make the decision whether the processed data will impact stock price positively or negatively based on Polarity and Subjectivity of the particular texts
4. Neural network that outputs a fully connected layer with a sigmoid or SoftMax activation function.
5. Usually the data is in textual form, hence splitting the text into split words is the first step to be done.
6. Comparing against pre-defined dictionary of lexicon to identify the words as positive, negative or neutral
7. Since here it's all about text, hence LSTM input will be sentence vectors.
    a. Split data into several parts per company.
    b. Put them into LSTM layer.
    c. Add a dense layer.
    d. Output layer will output sentiment analysis result, value range from 0.0~1.0.
    e. Then a 4-quantile value is used. If value < value <= 75%quantile as 0(even); value > 75%quantile as 1(positive).
8. Polarity score (from sentiment analysis) will be added as new column in the data frame for further reference during processing.
9. Polarity score will be considered as part of feature while making LSTM processing.

Date	High	Low	Close	Volume	Polarity
14-0	226.4	222.8	223.09	32226700	0
15	220.7	217.09	218.75	39763300	0.0166
16	220.1	217.56	217.78	21154324	0
17	220.82	219.43	220.7	18443215	-0.1665

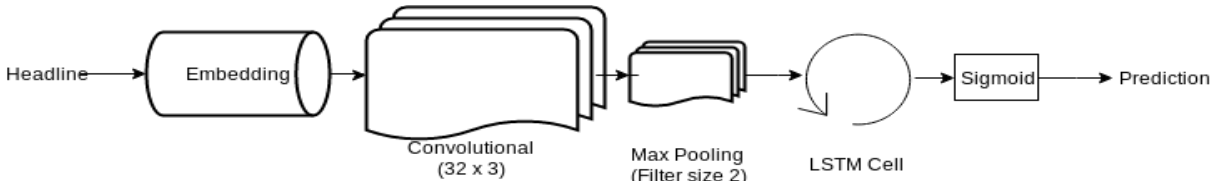


Figure 56 LSTM Network model for prediction [73]

7.2 LSTM Network Input Processing :

1. First step is to pre-process the news data, thus removing all non-alphabetical characters
2. Construct the split vector.
3. This input data will be used for further text processing to find polarity and subjectivity of text.
4. Training

    - 80% of the dataset would be used for training
    - 20% of the dataset would be used for testing.

7.3 LSTM Sentiment Analysis Implementation Algorithm:

- Training and Testing data segregation (80%-20% respectively)
- Data Normalization by applicating scalar co-efficients.
- Identifying the hidden layers of LSTM (along with loss function and hyper-parameters)
- Construct LSTM network by combining all LSTM functions and hidden layers together
- Apply prediction methodology against LSTM out to obtain the final output.

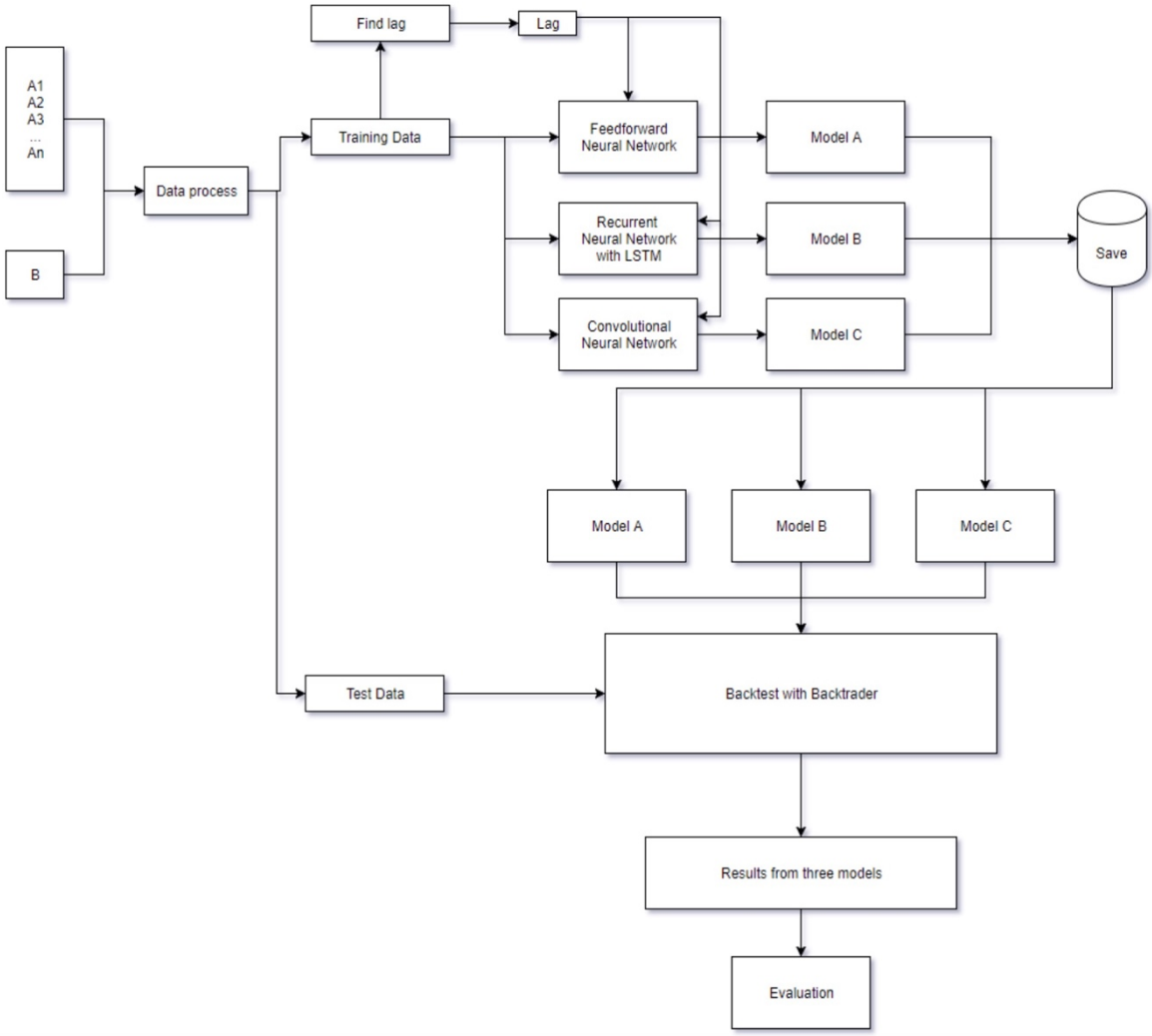


Figure 57 LSTM Network Data Processing with Backtest

7.4 End to End Use Case Diagram:

A typical workflow for this project is represented with following use case diagram.

The user now has ability to visualize stock predictions using pre-trained neural network models. The visualization for predicted prices could help a buyer to perform Buy or Sell on a stock under consideration.

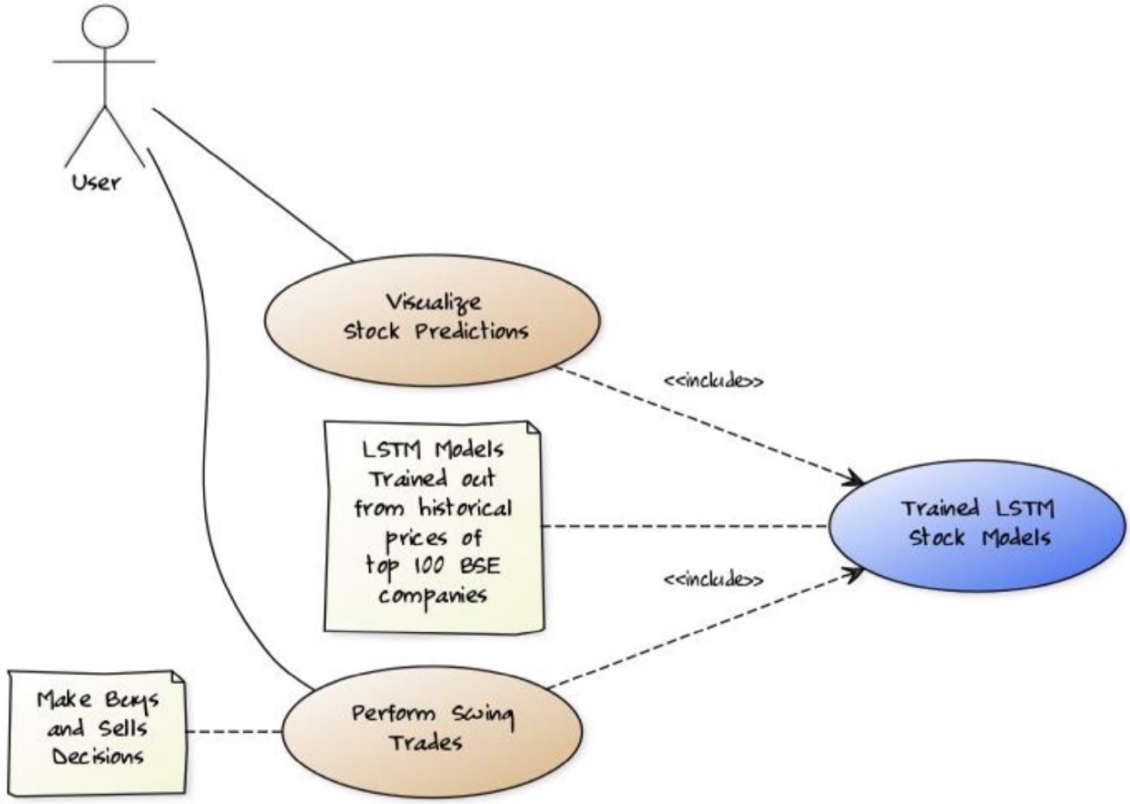


Figure 58 Use Case Diagram

7.5 Dataset Preparation:

The dataset that will be used to train the neural network would be the historical stock values of a company's share. The historical values would be ranging from 5 years to 10 years of data. The data would be available freely from website like: https://in.investing.com; https://www.quandl.com.

The dataset in figure 5 has lot many columns in it, however for the training purpose, only 'Date' and 'Close' Price columns will be taken into considerations.

The dataset will be split into two parts – training data (80%) testing data (20%). Since we will pass on this data to the LSTM neural network, we would have to preprocess the data set in such a way such that LSTM can tell for given say past N days stock price the next day's value was XX.

Date	Open	High	Low	Close	WAP	No. of Shares	No. of Trades	Total Turnover	Deliverable Quantity	% Deli. Qty to Traded Qty	Spread H-L	Spread C-O
9/11/2019	821.65	827.95	814.7	820.1	819.4	265029	3642	217165538	182625	68.91	13.25	-1.55
9/9/2019	838.05	840	827.3	829.2	831.14	118172	2988	98217182	39672	33.57	12.7	-8.85
9/6/2019	838	847.4	835.15	840.15	841.19	145166	3732	122112116	53722	37.01	12.25	2.15
9/5/2019	827.1	837	826.4	834.2	832.62	252694	8205	210397897	114057	45.14	10.6	7.1
9/4/2019	813.7	822.7	810.25	821.05	818.21	138123	3340	113013583	50373	36.47	12.45	7.35
9/3/2019	815	822.3	812.05	814.3	817.62	132158	3368	108055543	44348	33.56	10.25	-0.7
8/30/2019	808	817.5	802	814.6	809.1	156965	4282	126999897	66135	42.13	15.5	6.6
8/29/2019	798.9	809.65	796	806.85	804.54	177460	4038	142773169	57733	32.53	13.65	7.95
8/28/2019	785.1	804.9	785.1	802.1	796.47	164188	4669	130770827	60709	36.98	19.8	17
8/27/2019	791	795	781.2	785	786.03	248061	6542	194983829	101999	41.12	13.8	-6
8/26/2019	800	806.75	787.5	802.9	799.58	174352	4662	139408209	48812	28	19.25	2.9
8/23/2019	798	809.95	796	801.9	801.94	355792	8624	285325149	132721	37.3	13.95	3.9
8/22/2019	800.8	800.8	792.5	795.9	797.22	199130	9671	158750258	76127	38.23	8.3	-4.9
8/21/2019	792.8	803	792.75	799.55	799.36	339416	13955	271316701	173728	51.18	10.25	6.75
8/20/2019	781.05	797.9	781.05	792.9	793.16	513651	16144	407406376	249585	48.59	16.85	11.85
8/19/2019	777	783.3	773.8	777.8	779.71	262801	12926	204908043	154392	58.75	9.5	0.8
8/16/2019	779	779.8	762.25	774.55	772.37	226017	8379	174567971	119816	53.01	17.55	-4.45
8/14/2019	774	778.5	768.45	774.9	773.45	341438	13388	264084683	242760	71.1	10.05	0.9
8/13/2019	788.35	788.35	760.5	764.1	772.52	600910	14270	464215245	349534	58.17	27.85	-24.25
8/9/2019	791.5	796.4	785.5	790.15	790.21	483333	11043	381932759	311905	64.53	10.9	-1.35

Figure 59 Sample Historical Stock Data

7.5.1 Downloading the stock data from quandl.com:

Python's qunadl API are used to fetch the data for given stock. Data fetched once, is then saved in a csv format that will be utilized for training and testing dataset preparations. Only the 'Date' and 'Close' column is taken into consideration while parsing the data.

7.5.2 Pre-processing the data:

The data once downloaded from quandl API, will be preprocessed to select the desired columns and discard the unused data.

7.5.3 LSTM Data Preparation

Before making LSTM fit, data has to be pre-processed in below mentioned steps

- The approach is to apply supervised learning problem for the given dataset, in-order to do that, first step is to transform the time series data.
- Since time series data is always stationary and has trends, transformation of time series data is very much critical pre-processing step.
- Second step is to transform the observations from previous steps to specific scale.

7.5.3.1 Transformation Process:

- Use the observation from the last time step (t-1) as the input.
- Use the observation at the current time step (t) as the output.
- Using library functions from Pandas (like Shift ()) to push the values from one position to another position.
- Using the approach of differencing the data, trend in the time series can be removed. Using Pandas library function (like Diff()), the difference from the previous observation and current observation will be found, which results in difference series (without trend data).

- Using LSTM activation function (like hyperbolic tangent (tanh)), scaling on the output value will be applied, which makes outputs values between -1 and 1.

- Applying scaling coefficients (like min and max) on the training dataset and applied to scale the test dataset and any forecasts.

7.5.4 Normalizing the Stock Data:

For getting better accuracy and allowing the neural network to converge easily, the stock prices will be first normalized before it can be prepared for training (Normalize the close price between [0, 1])

```
def normalize_data(df):
    min_max_scaler = preprocessing.MinMaxScaler()
    df['open'] = min_max_scaler.fit_transform(df.open.values.reshape(-1,1))
    df['high'] = min_max_scaler.fit_transform(df.high.values.reshape(-1,1))
    df['low'] = min_max_scaler.fit_transform(df.low.values.reshape(-1,1))
    df['volume'] = min_max_scaler.fit_transform(df.volume.values.reshape(-1,1))
    df['adj close'] = min_max_scaler.fit_transform(df['adj close'].values.reshape(-1,1))
    return df
df = normalize_data(df)
df.plot(figsize=(23,10))
plt.show()
plt.subplot(411)
plt.plot(df.open, label='Original')
plt.legend(loc='best')
plt.subplot(412)
plt.plot(df.low, label='Trend')
plt.legend(loc='best')
plt.subplot(413)
plt.plot(df.high,label='Seasonality')
plt.legend(loc='best')
plt.subplot(414)
plt.plot(df.volume, label='Residuals')
plt.legend(loc='best')
plt.tight_layout()
plt.show()
```

Figure 60 Data Normalization Code

7.5.5 Training Set and Testing Set

- For feeding the data to the neural network, the stock data needs to be reshaped.
- For each 5 previous stock prices we would train for the 6th Stock Price

```
def load_data(stock, seq_len):
    amount_of_features = len(stock.columns) # 5
    data = stock.as_matrix()
    sequence_length = seq_len + 1 # index starting from 0
    result = []

    for index in range(len(data) - sequence_length): # maxmimum date = lastest date - sequence length
        result.append(data[index: index + sequence_length]) # index : index + 22days

    result = np.array(result)
    row = round(0.9 * result.shape[0]) # 90% split
    train = result[:int(row), :] # 90% date, all features

    x_train = train[:, :-1]
    y_train = train[:, -1][:,-1]

    x_test = result[int(row):, :-1]
    y_test = result[int(row):, -1][:,-1]

    x_train = np.reshape(x_train, (x_train.shape[0], x_train.shape[1], amount_of_features))
    x_test = np.reshape(x_test, (x_test.shape[0], x_test.shape[1], amount_of_features))

    return [x_train, y_train, x_test, y_test]
```

Figure 61 Training and Testing Dataset

Date	Close
1/1/2019	664.65
1/2/2019	669.3
1/3/2019	667.55
1/4/2019	660.75
1/7/2019	671.15
1/8/2019	669.85

Figure 62 Stock Closing price for week data

7.5.6 UML Diagram

Following UML represent the entire design of the application.

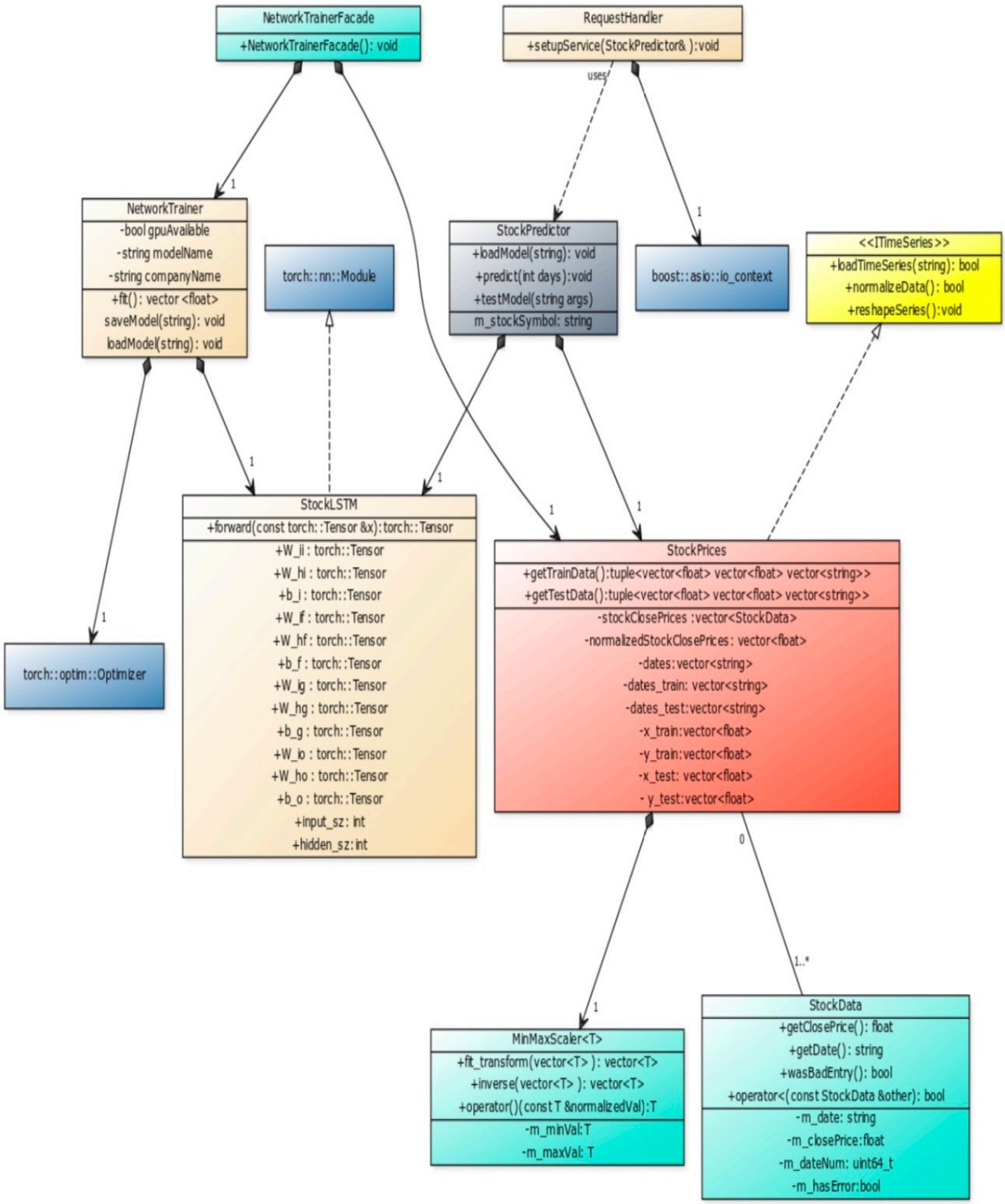


Figure 63 UML Diagram

7.5.7 Neural Network Creation

There are several neural network libraries which provide built in support for LSTM neural network, using one of the widely used open source library. is a nice open source library widely used in research areas.

LSTM Design Network Architecture visualizing using MATLAB

```
layers = [
    sequenceInputLayer(1,"Name","Input Sequence")
    lstmLayer(32,"Name","StockLSTM")
    dropoutLayer(0.1,"Name","dropout_1")
    lstmLayer(32,"Name","lstm")
    dropoutLayer(0.2,"Name","dropout_2")];
```

Number of layers: 5

Number of connections: 4

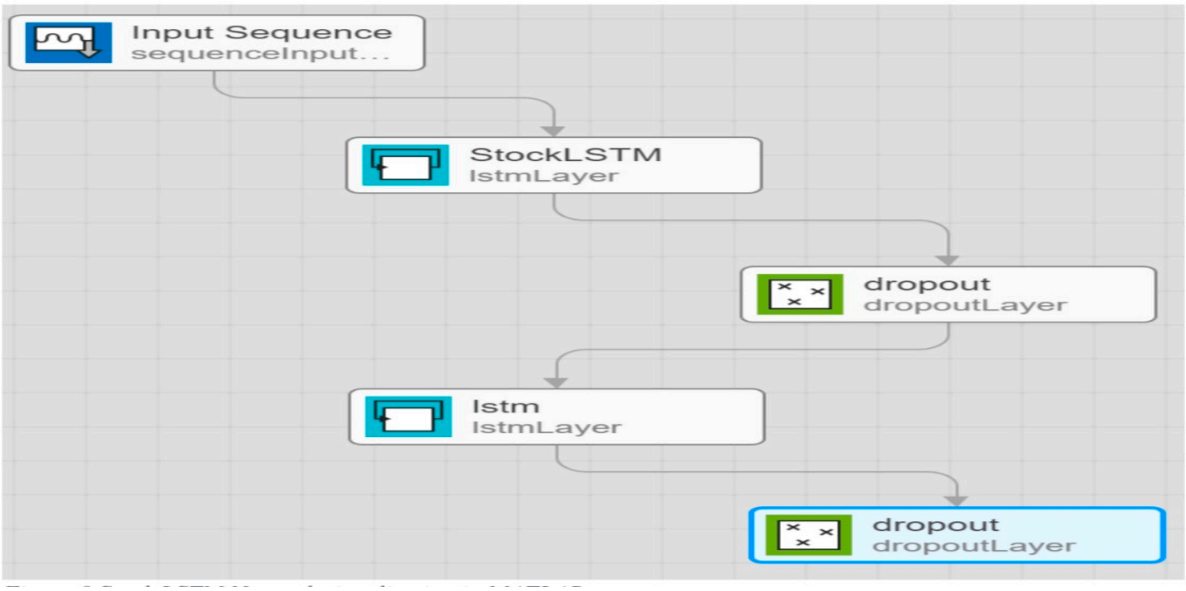


Figure 64 LSTM Model creation

7.5.8 The Stock Price Neural Network Design:

Incorporating the LSTM Neural Network created above, another neural network wrapper is created which would be finally used for training on stock prices. The LSTM stacking is represented as follows, which includes a Dropout layer followed by a Linear Layer.

A feed forward pass to this neural network would proceed as follows:

1. Input tensor is of size (previousSamples, totalBatch, 1) and is feedforward to LSTM Layer -1 (All states here are initialized to 0).
2. Output of Layer-1 is feedforward to LSTM Layer -2 with the states captures.
3. Adjust the output tensor which is (totalBatch, 1) and make it to (totalBatch) and feedforward to the Drop Out Layer with probability of 20 %
4. The final output of dropout layer is feedforward on Linear Layer.
5. Linear Layer's output the final output for one epoch, the size is the (totalBatch)

7.5.9 LSTM Network Model (with Hidden Layers)

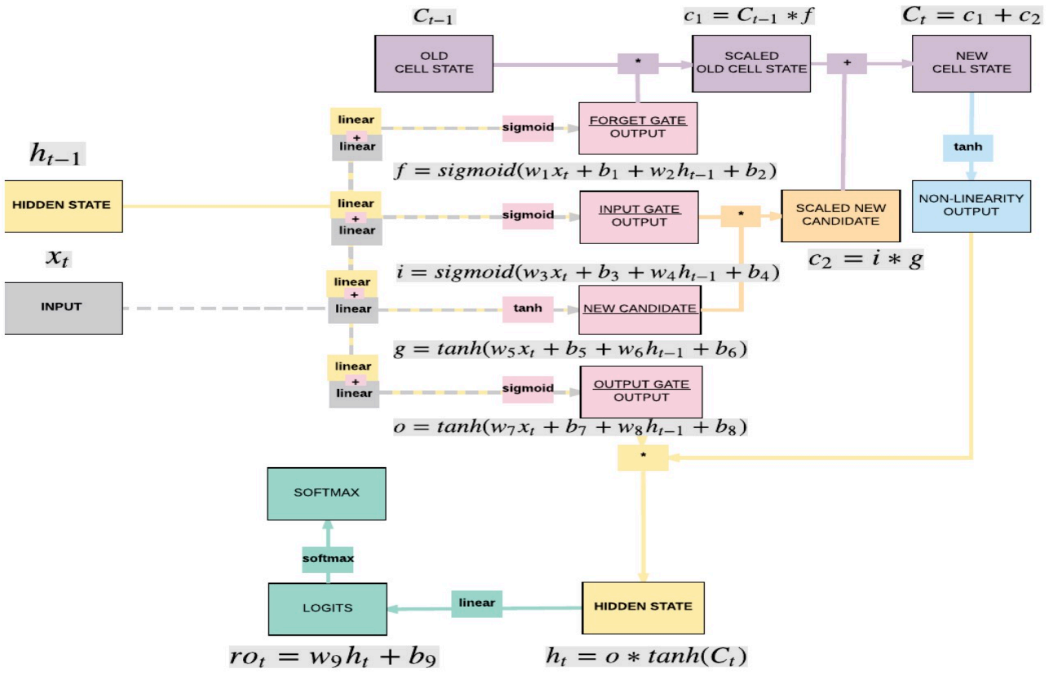


Figure 65 LSTM with Hidden Layer Diagram [74]

7.5.9.1 LSTM Network Model Implementation Considerations:

After successful Data-Preprocessing , LSTM model has to be created along with Loss and Optimizer class as shown in below figures.

1. Create Model Class
2. **Instantiate Model Class**
3. Instantiate Loss Class
4. Long Short-Term Memory Neural Network: **Cross Entropy Loss**
    - *Recurrent Neural Network*: **Cross Entropy Loss**
    - *Convolutional Neural Network*: **Cross Entropy Loss**
    - *Feedforward Neural Network*: **Cross Entropy Loss**
    - *Logistic Regression*: **Cross Entropy Loss**
    - *Linear Regression*: **MSE**
5. Instantiate Optimizer Class
    - Simplified equation
        - $\theta = \theta - \eta \cdot \nabla\theta$ $\theta = \theta - \eta \cdot \nabla\theta$
            - $\theta\theta$: parameters (our variables)
            - $\eta\eta$: learning rate (how fast we want to learn)
            - $\nabla\theta\nabla\theta$: parameters' gradients
    - Even simplier equation
        - parameters = parameters - learning_rate * parameters_gradients

7.5.10 LSTM Network Model Training Process:

Training the neural network would be done with some constraints. The training pass will be stopped when either of the following criteria is met.

- A maximum number of epochs is exhausted.
- A minimum required loss is achieved, and network did not converge.
- A maximum threshold time is elapsed, and network did not converge.

a. **Convert inputs/labels to variables :** LSTM Network which is of input size [1, 28]

b. Identify if there is any gradient buffets, if so clear them.

c. Process input and generate output for the same input.

d. Calculate the loss using Loss Model

e. Identify the gradient using hyper-parameters which are configured while creating model.

f. Gradient will be getting updated (repeatedly) using the given hyper parameters.

- "parameters = parameters - learning_rate * parameters_gradients"[77]

g. Re-iterate the process again and again.

```
'''
STEP 1: LOADING DATASET
'''
train_dataset = dsets.MNIST(root='./data',
                            train=True,
                            transform=transforms.ToTensor(),
                            download=True)

test_dataset = dsets.MNIST(root='./data',
                           train=False,
                           transform=transforms.ToTensor())

'''
STEP 2: MAKING DATASET ITERABLE
'''

batch_size = 100
n_iters = 3000
num_epochs = n_iters / (len(train_dataset) / batch_size)
num_epochs = int(num_epochs)
```

```
'''
STEP 3: CREATE MODEL CLASS
'''

class LSTMModel(nn.Module):
    def __init__(self, input_dim, hidden_dim, layer_dim, output_dim):
        super(LSTMModel, self).__init__()
        # Hidden dimensions
        self.hidden_dim = hidden_dim

        # Number of hidden layers
        self.layer_dim = layer_dim

        # Building your LSTM
        # batch_first=True causes input/output tensors to be of shape
        # (batch_dim, seq_dim, feature_dim)
        self.lstm = nn.LSTM(input_dim, hidden_dim, layer_dim, batch_first=True)

        # Readout layer
        self.fc = nn.Linear(hidden_dim, output_dim)
```

```
'''
STEP 4: INSTANTIATE MODEL CLASS
'''
input_dim = 28
hidden_dim = 100
layer_dim = 2  # ONLY CHANGE IS HERE FROM ONE LAYER TO TWO LAYER
output_dim = 10

model = LSTMModel(input_dim, hidden_dim, layer_dim, output_dim)

# JUST PRINTING MODEL & PARAMETERS
print(model)
print(len(list(model.parameters())))
for i in range(len(list(model.parameters()))):
    print(list(model.parameters())[i].size())

'''
STEP 5: INSTANTIATE LOSS CLASS
'''
criterion = nn.CrossEntropyLoss()

'''
STEP 6: INSTANTIATE OPTIMIZER CLASS
'''
learning_rate = 0.1

optimizer = torch.optim.SGD(model.parameters(), lr=learning_rate)
```

Figure 66 LSTM Model Creation Code

```
STEP 7: TRAIN THE MODEL
'''

# Number of steps to unroll
seq_dim = 28

iter = 0
for epoch in range(num_epochs):
    for i, (images, labels) in enumerate(train_loader):
        # Load images as Variable
        images = images.view(-1, seq_dim, input_dim).requires_grad_()

        # Clear gradients w.r.t. parameters
        optimizer.zero_grad()

        # Forward pass to get output/logits
        # outputs.size() --> 100, 10
        outputs = model(images)

        # Calculate Loss: softmax --> cross entropy loss
        loss = criterion(outputs, labels)

        # Getting gradients w.r.t. parameters
        loss.backward()

        # Updating parameters
        optimizer.step()

        iter += 1

        if iter % 500 == 0:
            # Calculate Accuracy
            correct = 0
            total = 0
            # Iterate through test dataset
            for images, labels in test_loader:
                # Load images to a Torch Variable
                images = images.view(-1, seq_dim, input_dim).requires_grad_()

                # Forward pass only to get logits/output
                outputs = model(images)

                # Get predictions from the maximum value
                _, predicted = torch.max(outputs.data, 1)

                # Total number of labels
                total += labels.size(0)

                # Total correct predictions
                correct += (predicted == labels).sum()

            accuracy = 100 * correct / total

            # Print Loss
            print('Iteration: {}. Loss: {}. Accuracy: {}'.format(iter, loss.item(), accuracy))
```

Figure 67 LSTM Model Training

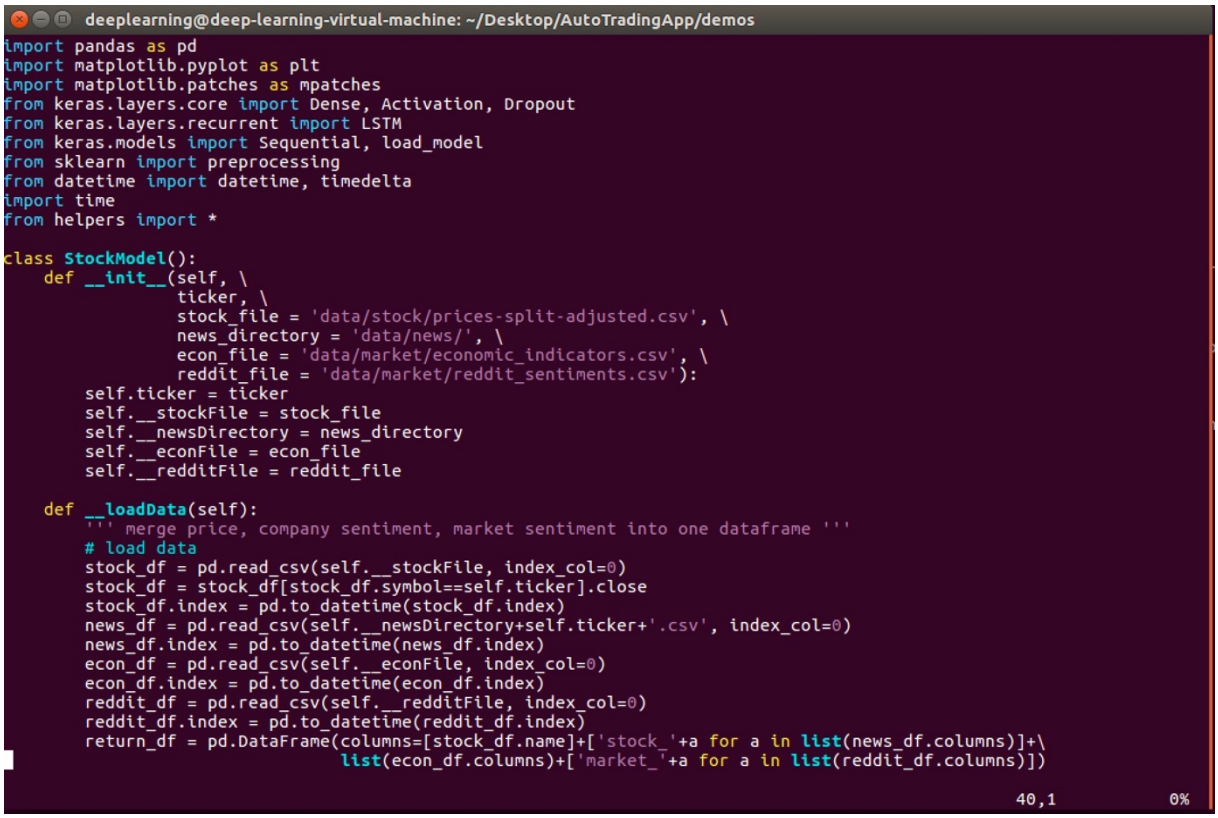


Figure 68 Demo Code Screenshots 1

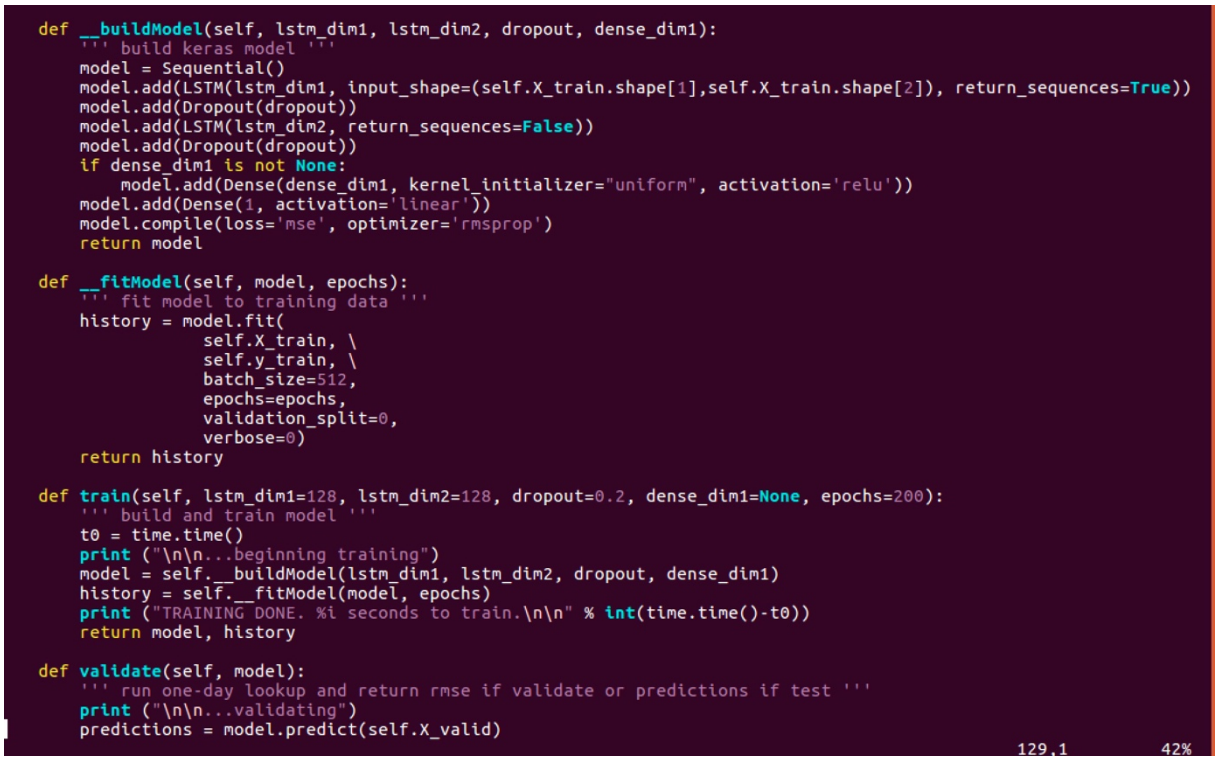


Figure 69 Demo Code Screenshots 2

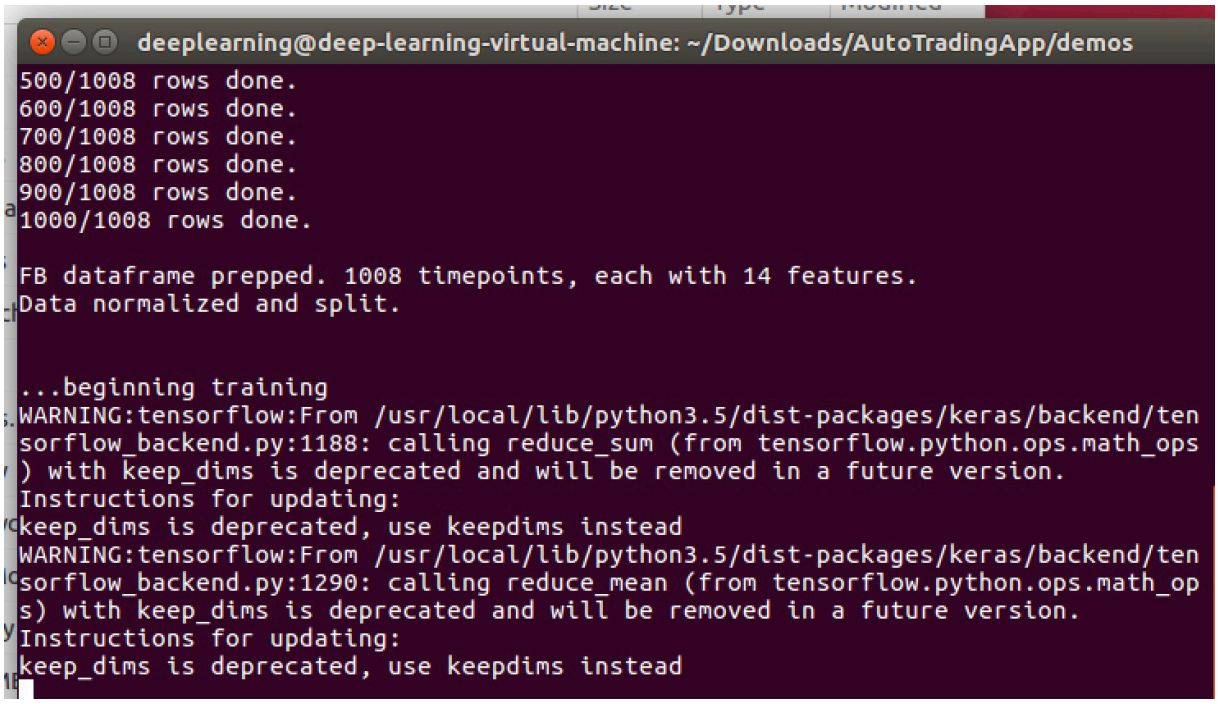


Figure 70 Demo Code Screenshots 3

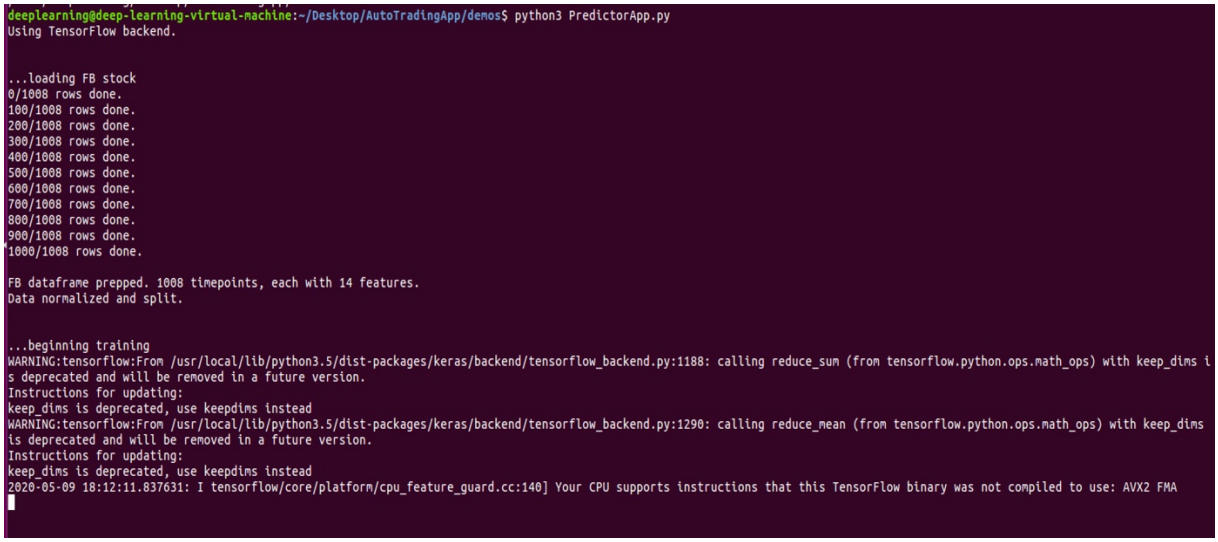


Figure 71 Demo Code Screenshots 4

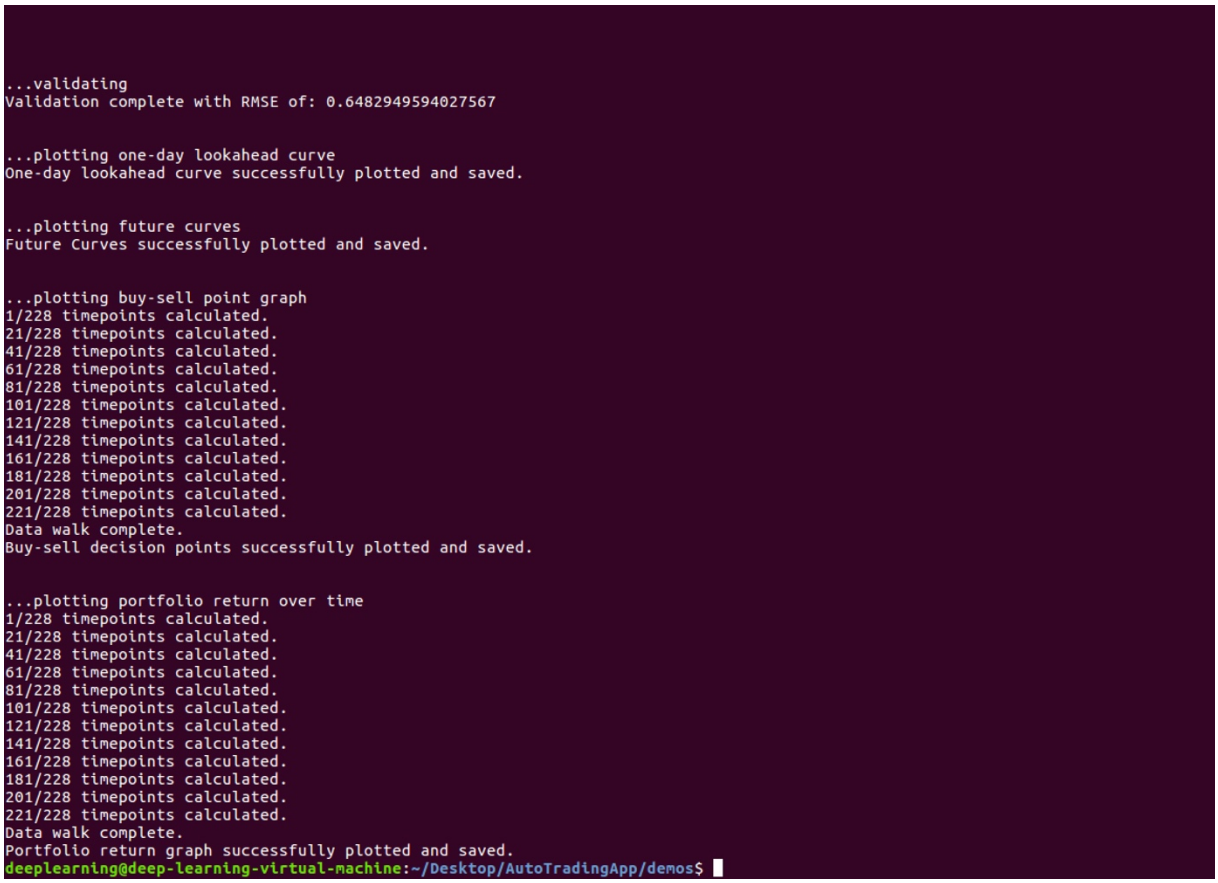


Figure 72 Demo Code Screenshots 5

7.5.11 One Epoch

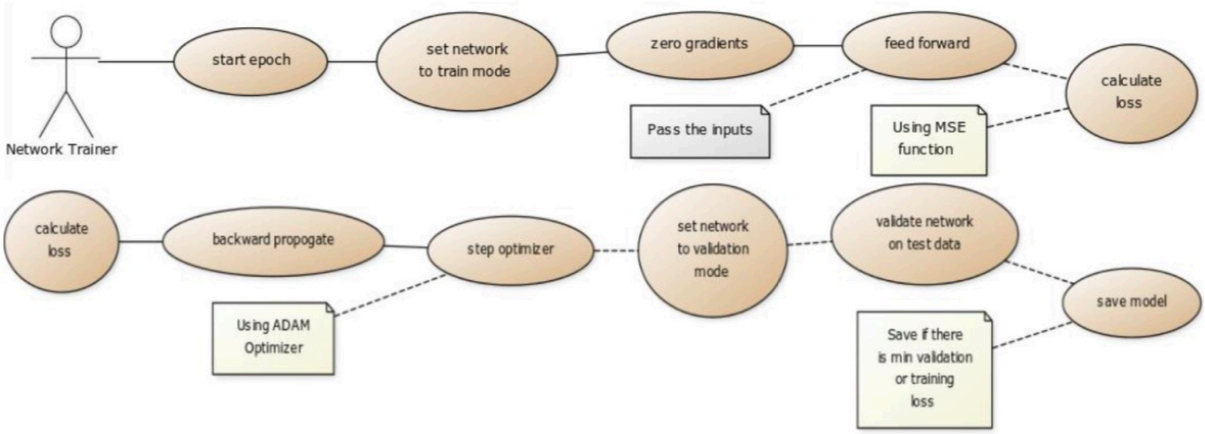


Figure 73 One Epoch Diagram

One epoch of training would do following:

1. Set network to training mode.
2. Zero out the gradients
3. Predict the output using the neural network
4. Calculate loss using mean square error function
5. Backward propagation
6. Step the optimizer
7. Calculate the training loss
8. Validate and save model
    - Set network to evaluation mode
    - Feed forward the test data
    - Calculate the validation loss

7.5.12 Testing the Neural Network

For testing the neural network, the save network is reloaded into RAM:

- Reload a previous trained neural network (For required future N days, for predicting 1 sample we would need last N but 5 samples)
- Get the test data for the most recent prices
- Erase all but 5 samples
- Normalize samples are predicted and added to the test vector
    - Predict the 6th closing price from 5 previous testSamples
    - Use the last price Prepare for next training set

7.5.13 Training Workflow

The application provides command line option to either train network for a specific stock or perform training for a list of stock provided through a csv file. For this project for sake of simplicity a list of top 100 company stocks, based on market capitalization, is taken out of companies listed on Bombay Stock Exchange and stored in a filename BSE100.csv. The file simple has stock symbols and companies name listed. For training on BSE100.csv file once stock is trained (when either of the training criteria is met), the trained neural network is stored in another csv file named full_trained.csv

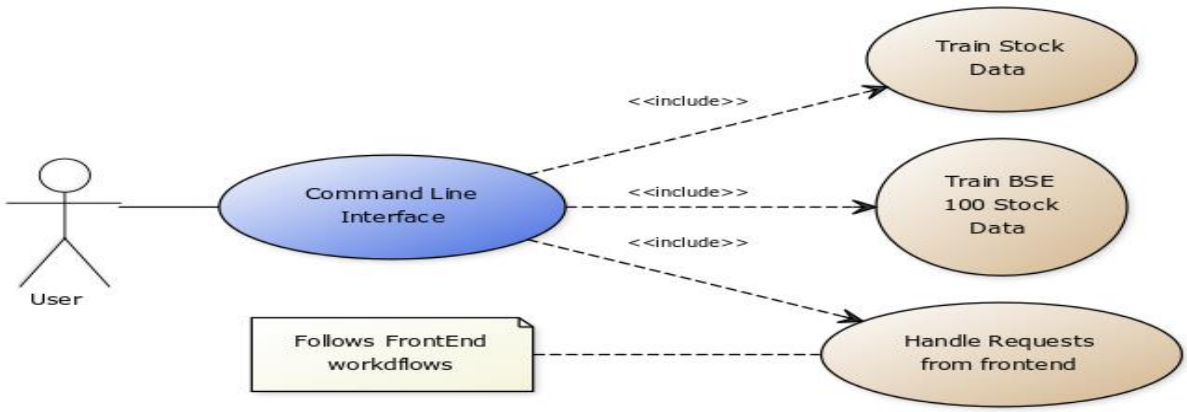


Figure 74 Data Training Workflow

7.5.14 UI Workflow:

The following use case diagram shows the workflow for usage of front-end application. The following figures shows the different sections created to visualize the trained neural network models for different stock.

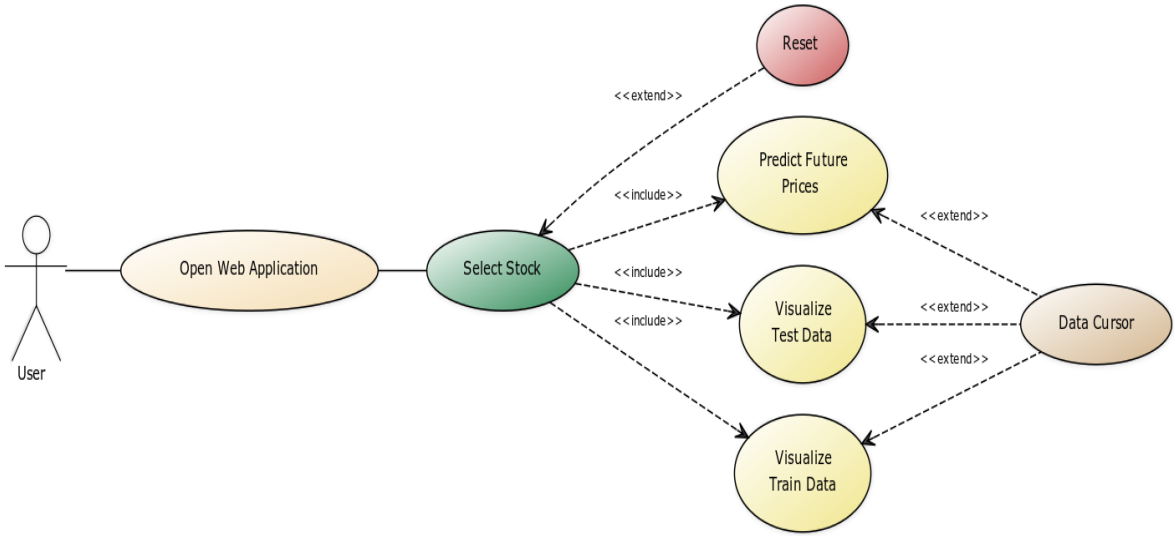


Figure 75 UI Workflow

8 Evaluation and Test Results

- The trend of the predicted stock prices is close to the trend of actual stock prices.
- Average weighted sentiment adds more importance to the data that has a wider audience exposure than just the sentiment. Combined three datasets (historic stock data, news data, twitter data) with an inner join. Using this joint table, four different test cases are created as seen in the table below.
- Using Feature Selection, elected only the features which sources have the greatest impact on a closing stock price.
- Below table indicates that, a combination of historic stock data with news data has the lowest RMSE, which means that a closing price can most accurately be predicted using these two sources. Having found out that using historic stock data with news data has the lowest RMSE, we proceeded with training our model with as much data as we have for each test case / combination of data sources. The results are shown in the tables below.
- **Case 1: Stock Data (LSTM)**

Case	Stock Data	Twitter Data	News Data	Look Back	Learning Rate	EPOCHs	Min RMSE
1	✓			12	0.002	100	5.64
2	✓	✓		15	0.002	100	5.78
3	✓		✓	15	0.002	100	4.72
4	✓	✓	✓	12	0.002	100	6.25

Table 13 Case 1 LSTM for Stock Data

Case 2: Stock Data and News Data

SNo	Epochs	Batch Size	Hidden Layers	Neurons	Loop Back	Dropout	Learning Rate	Train RMSE	Test RSME
1	100	2	1	8× 32 × 1	15	0	0.001	2.51	13.17
2	100	1	1	8× 32 × 1	30	0.5	0.002	3.94	33.78
3	100	2	2	8 × 32 × 32 × 1	30	0	0.001	2.14	22.48

Table 14 Case 2 LSTM for Stock Data News Data

SNo	Epochs	Batch Size	Hidden Layers	Neurons	Loop Back	Dropout	Learning Rate	Train RMSE	Test RSME
1	100	2	1	8× 32 × 1	15	0	0.001	3.26	14.39
2	100	1	1	8× 32 × 1	30	0.5	0.002	4.38	27.80
3	100	2	2	8 × 32 × 32 × 1	30	0	0.001	2.34	16.31

Case 3: Stock Data, Twitter Data

SNo	Epochs	Batch Size	Hidden Layers	Neurons	Loop Back	Dropout	Learning Rate	Train RMSE	Test RSME

1	100	2	1	8× 32 × 1	15	0	0.001	3.22	7.68
2	100	1	1	8× 32 × 1	30	0.5	0.002	4.25	12.41
3	100	2	2	8 × 32 × 32 × 1	30	0	0.001	3.52	10.52

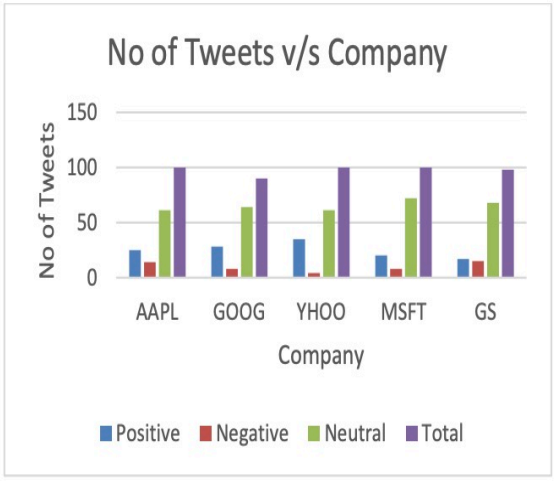
Table 15 Case 3 Stock Data, Twitter Data, News Data

Case 4: Stock Data, Twitter Data and News Data

SNo	**Epochs**	**Batch Size**	**Hidden Layers**	**Neurons**	**Loop Back**	**Dropout**	**Learning Rate**	**Train RMSE**	**Test RSME**
1	100	2	1	8× 32 × 1	15	0	0.001	4.11	12.48
2	100	1	1	8× 32 × 1	30	0.5	0.002	4.47	16.49
3	100	2	2	8 × 32 × 32 × 1	30	0	0.001	4.20	13.75

Table 16 Case 4 Stock Data, Twitter Data, News Data

LSTM model which takes as an input the generated features from the Stock, Twitter and News dataset and predicts the stock prices for the next day by using the past n days information. As the model is trained for the past years, the model can predict the future stock prices.



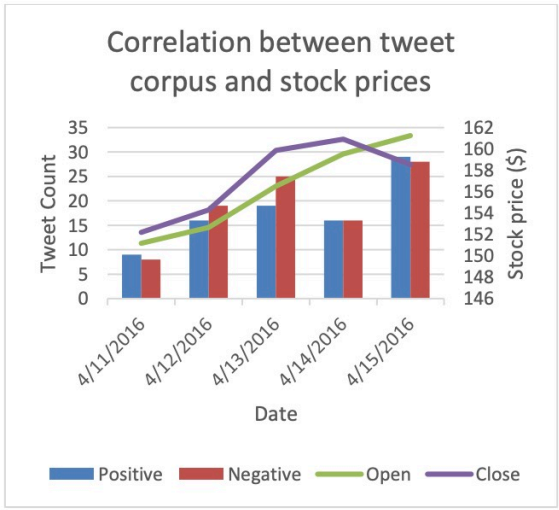


Figure 76 Comparison of Tweets and Corpus Stocks

SNo	Feature Combination	Dimension	Accuracy
1	Stock Data	9	64.86
2	Stock Data + Technical Index Data	20	66.75
3	**Stock Data + Tweet Data + News Data**	**22**	**75.78**
4	Stock Data + Tweet Data	19	65.38

Table 17 Prediction Accuracy vs Feature Combination

Finally, closing price predictions using combination of different features can be seen in the below figures.

Case 1: Historical Stock Data

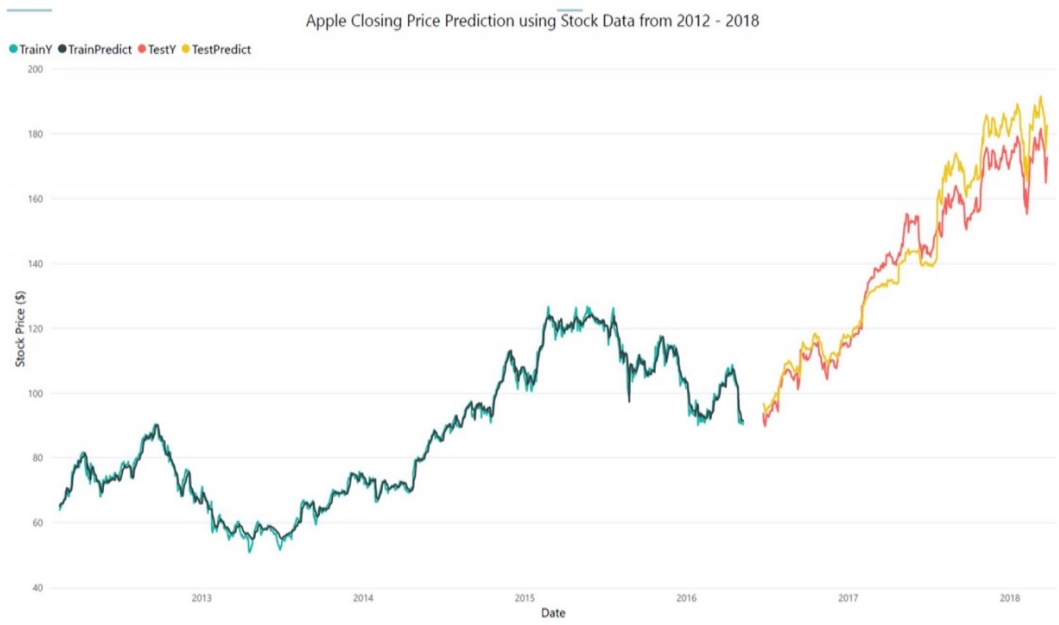


Figure 77 Apple Historical Data

Case 2: Historical Stock Data & Twitter Data

Figure 78 Apple Historical Data and Twitter Data

Case 3: Historical Stock Data & News Data

Figure 79 Apple Historical Data and News Data

Case 4: Historical Stock Data, Twitter Data & News Data

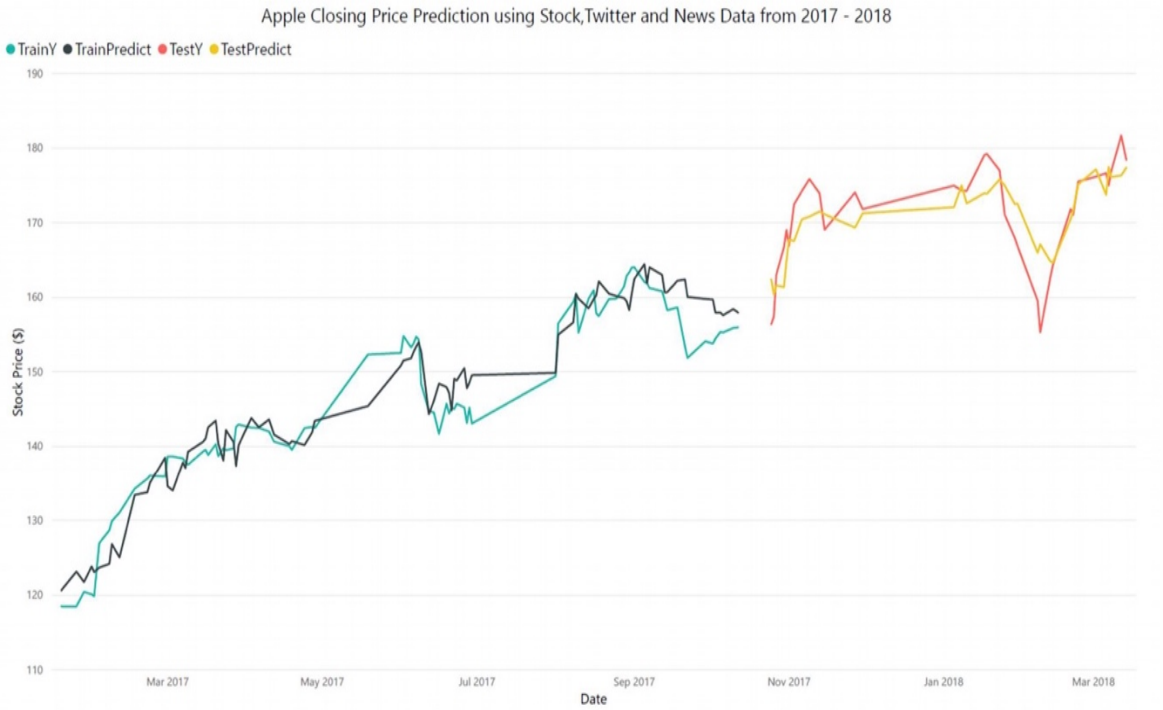


Figure 80 Apple Historical Data, News Data and Twitter Data

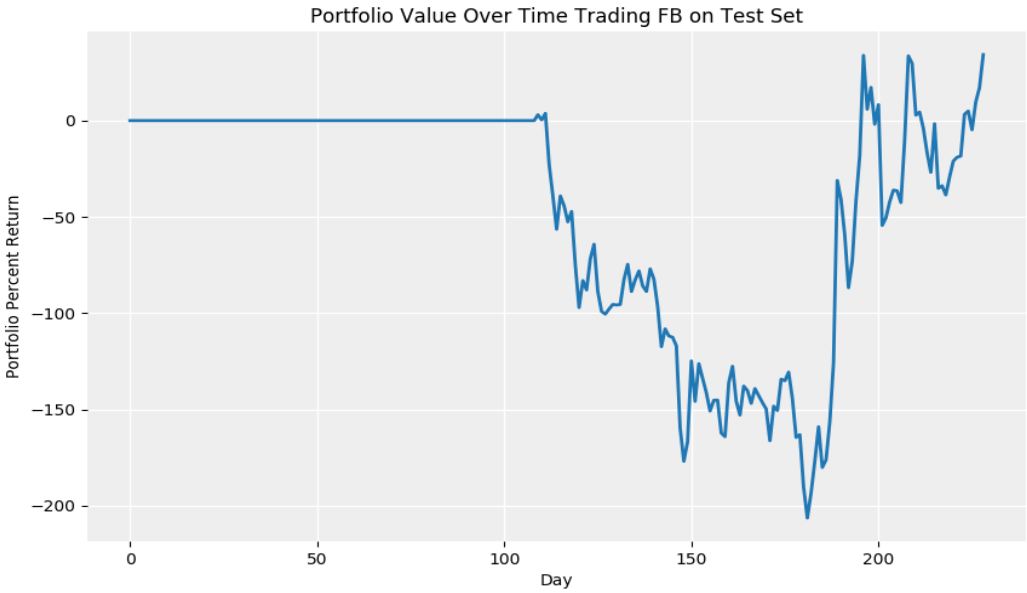


Figure 81 Portfolio Value Over Time Trading FB on Test Set

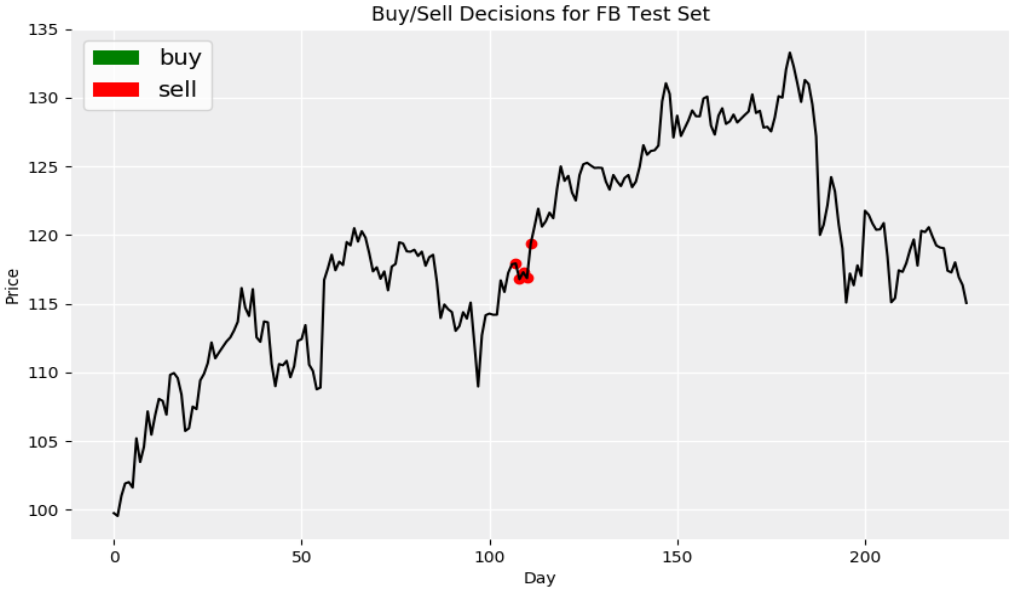


Figure 82 Buy/Sell Decisions for FB Test Set

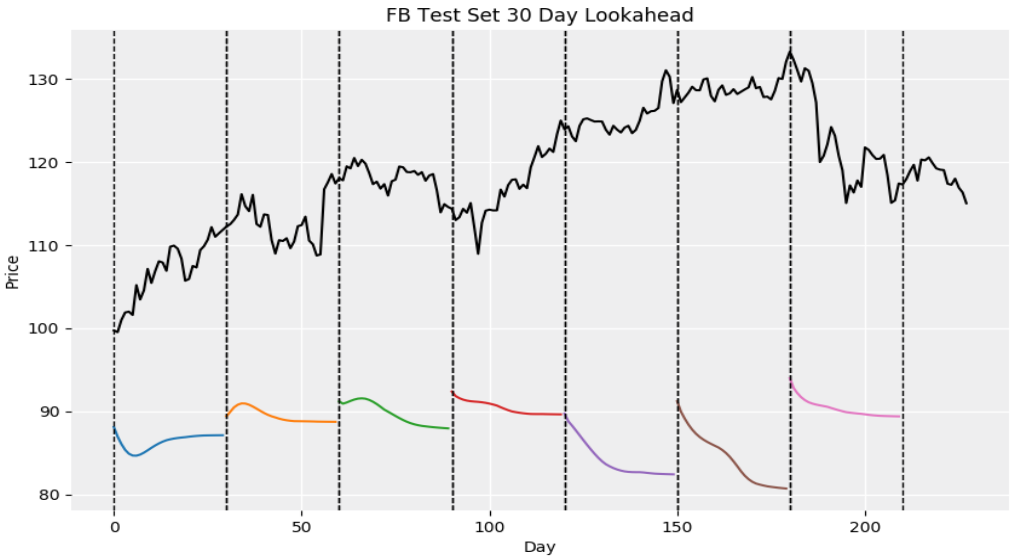


Figure 83 FB Test Set 30 Day Lookahead

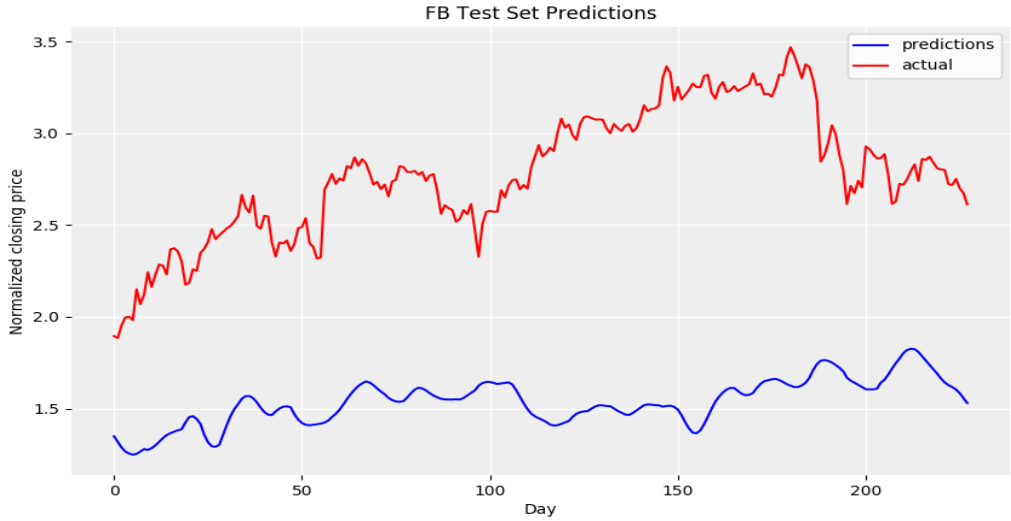


Figure 84 FB Test Set Predictions

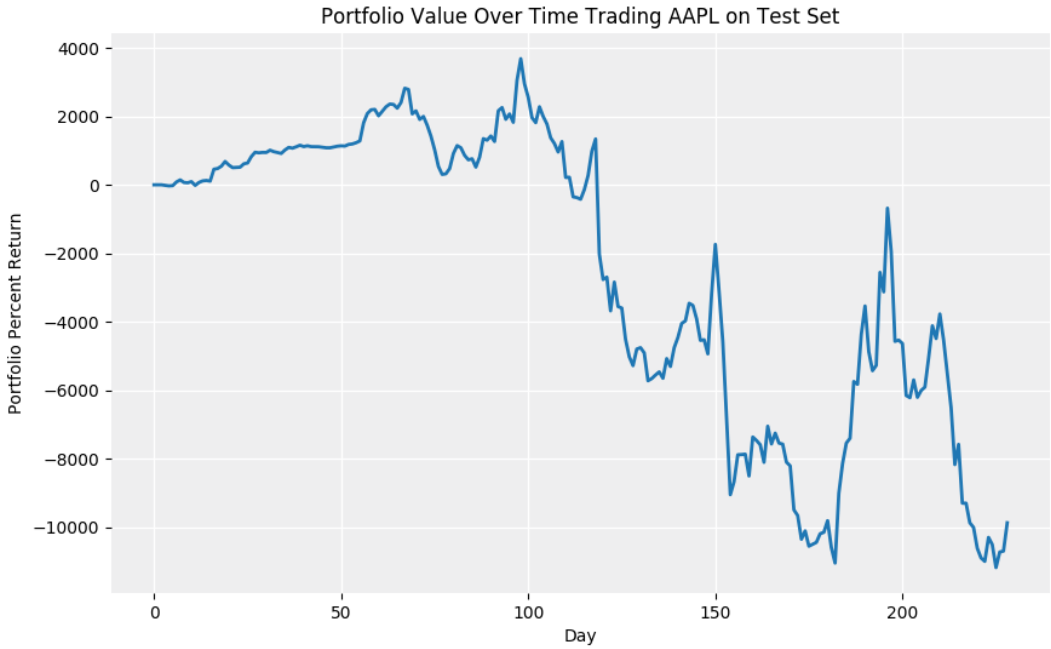


Figure 85 Portfolio Value Over Time trading for AAPL

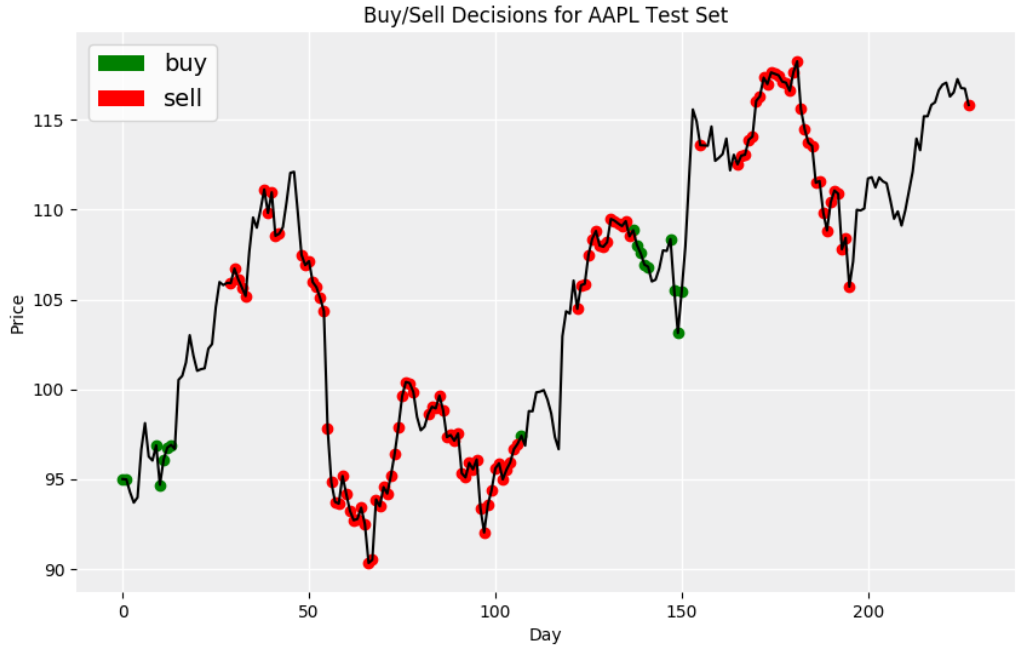


Figure 86 Buy/Sell Decisions for APPL Test Set

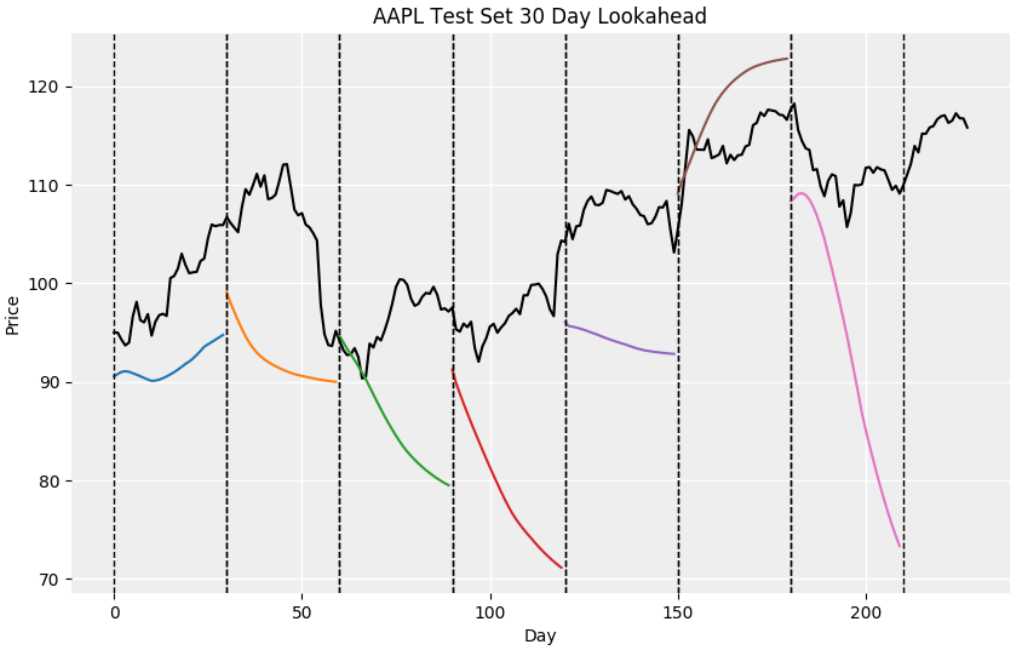


Figure 87 AAPL Test Set 30 Day Lookahead

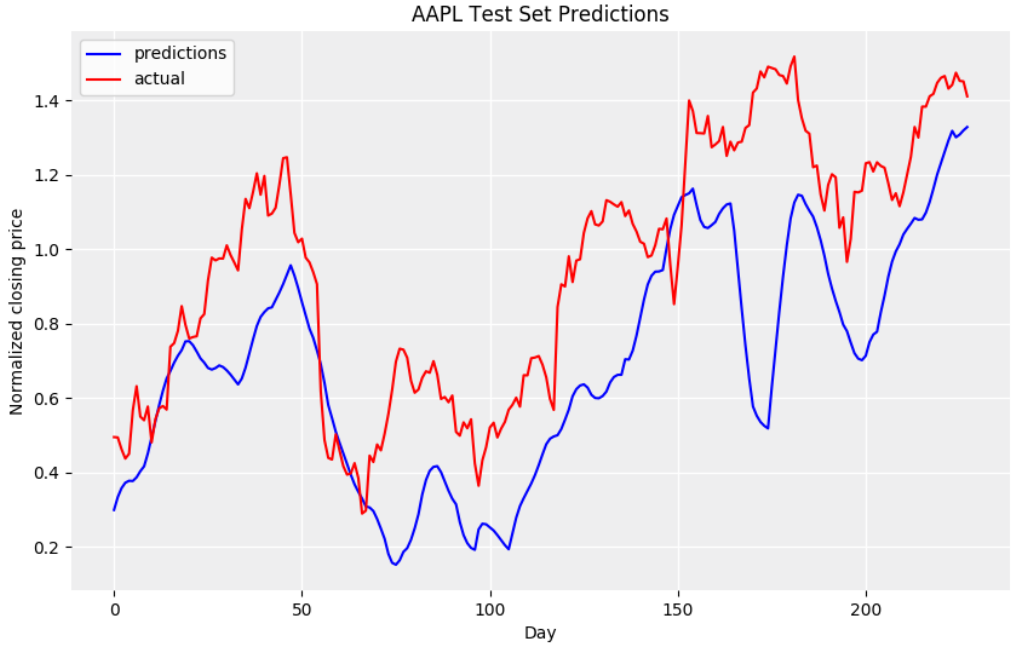


Figure 88 AAPL Test Set Predictions

9 Conclusions and Learnings

Forecasting the price of a financial asset is of great importance as it reduces the risk of decision making for investors. The prices of financial assets are dynamic and respond are dependent on a lot of factors which decide its movement. Being able to account every single one of these in hope of making precise predictions is a challenging act. A large number of investment experts and firms have taken advantage of this scenario and gained large profits from inexperienced investors. These studies have indicated that LSTMs have been the best player in this field and deep learning is the way ahead.

This project was able to demonstrate the trends in financial data prediction and explore further application of these techniques. Hyper-tuning of parameters, applying different normalization techniques, adding Regularization and dropout to eliminate Over-fitting and other optimization techniques along with these improved networks helped in achieving lower errors and better predictions.

Key Learnings:

- Extracting features and aggregating large datasets which is of in some GBs using various data wrangling techniques.
- Understanding various feature engineering mechanisms to create features that are important for predictions for a model in the stock market domain. Performing feature engineering to increase the value of a feature by combining multiple features. SDA (Stacked Denoising Auto Encoder) is applied to reduce the dimension of features which is not sensitive to the noise. The aim of an auto encoder is to learn higher-level representation for a set of data, typically for dimension reduction.
- NLP techniques to extract features like sentiments from text data.
- Creating LSTM model over time series data for future price prediction.
- How to decide which feature has the most impact on the prediction.
- Understanding finance domain and stock market field and how the data of this domain is dependent on various features such as VWAP and sentiments.
- LSTM models were then plugged in to Algorithmic Trading strategies leveraged portfolios for back-testing. This gave great insights about financial analysis along with financial predictions.

- Also, other financial factors like trading signals, investor sentiment, news abstracts etc. should be added to the model input variables in order to further the generalization and reduce overfitting in model in the future.
- This lead to have better understanding of Machine Learning and Data Science to increase by leaps and strides. Also got acquainted with the financial domain of knowledge and expanded the knowledge to new fields.

10 Expected Outcomes

Expected outcome for a given stock is to find the target selling/buying price to maximize the profit. Time series forecasting is a very intriguing field to work with. Particularly when it comes to stock predictions eagerness to learn and put more thoughts which eventually could be used for making better financial investments. The model was evaluated to predict few stocks using this model. Predications made here is riskier for performing direct trading in share market based on the values predicted, however this would be surely helping to do some swing trading with least risk. The project work explores the LSTM Neural Network, which itself is a very vast topic for research work.

Few assumptions based on which this portfolio strategy methodology is applied.

- Since the impact of news (good/bad) on stock data is always short lived, first assumption is that trade (buy and sell or sell and buy) will be completed with-in 30 minutes to make maximum profit.

- Returns were always calculated based on based on the price at which trade has happened minus the closing price for that day.

- Sentiment Score based model suggested trade is considered right prediction if that stocks gains 0.5 or more percentage.

```
# Getting the real stock price of 2017
dataset_test = pd.read_csv('Google_Stock_Price_Test.csv')
real_stock_price = dataset_test.iloc[:, 1:2].values

# Getting the predicted stock price of 2017
dataset_total = pd.concat((dataset_train['Open'], dataset_test['Open']), axis = 0)
inputs = dataset_total[len(dataset_total) - len(dataset_test) - INPUT_SIZE:].values
inputs = inputs.reshape(-1,1)
inputs = sc.transform(inputs)
X_test = []
for i in range(INPUT_SIZE, 80):
    X_test.append(inputs[i-INPUT_SIZE:i, 0])
X_test = np.array(X_test)
X_test = np.reshape(X_test, (X_test.shape[0], 1, X_test.shape[1]))
```

```
X_train_X_test = np.concatenate((X_train, X_test),axis=0)
hidden_state = None
test_inputs = Variable(torch.from_numpy(X_train_X_test).float())
predicted_stock_price, b = rnn(test_inputs, hidden_state)
predicted_stock_price = np.reshape(predicted_stock_price.detach().numpy(), (test_inputs.shape[0],
1))
predicted_stock_price = sc.inverse_transform(predicted_stock_price)

real_stock_price_all = np.concatenate((training_set[INPUT_SIZE:], real_stock_price))
```

```
# Visualising the results
plt.figure(1, figsize=(12, 5))
plt.plot(real_stock_price_all, color = 'red', label = 'Real')
plt.plot(predicted_stock_price, color = 'blue', label = 'Pred')
plt.title('Google Stock Price Prediction')
plt.xlabel('Time')
plt.ylabel('Google Stock Price')
plt.legend()
plt.show()
```

Figure 89 Code Snippet for Visualizing the results

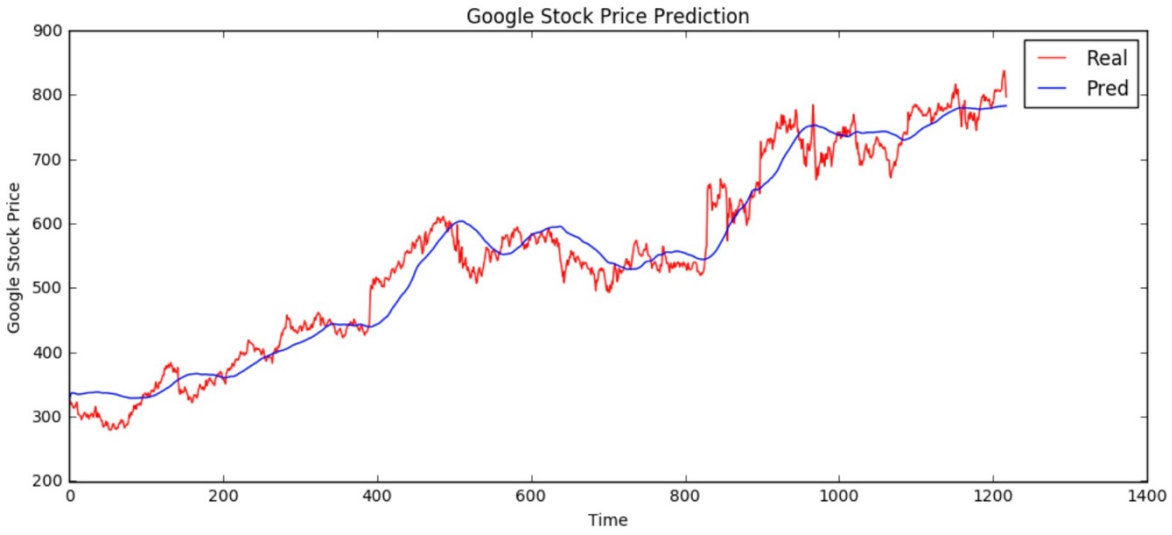


Figure 90 Google Price Prediction. Vs. Actual

```
import matplotlib.pyplot as plt2
plt.figure(figsize=(25,8))
plt2.plot(newp,color='red', label='Prediction')
plt2.plot(newy_test,color='blue', label='Actual')
plt2.legend(loc='best')
plt2.show()
```

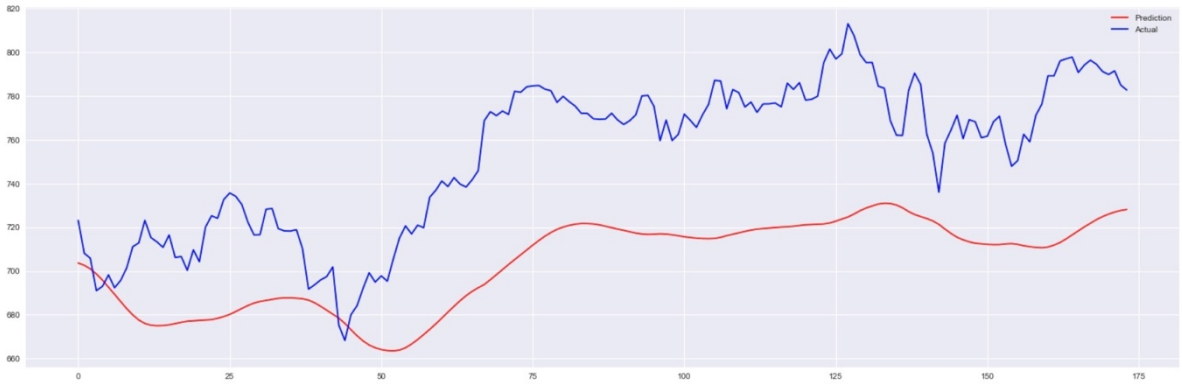


Figure 91 Sample Plot with Precition

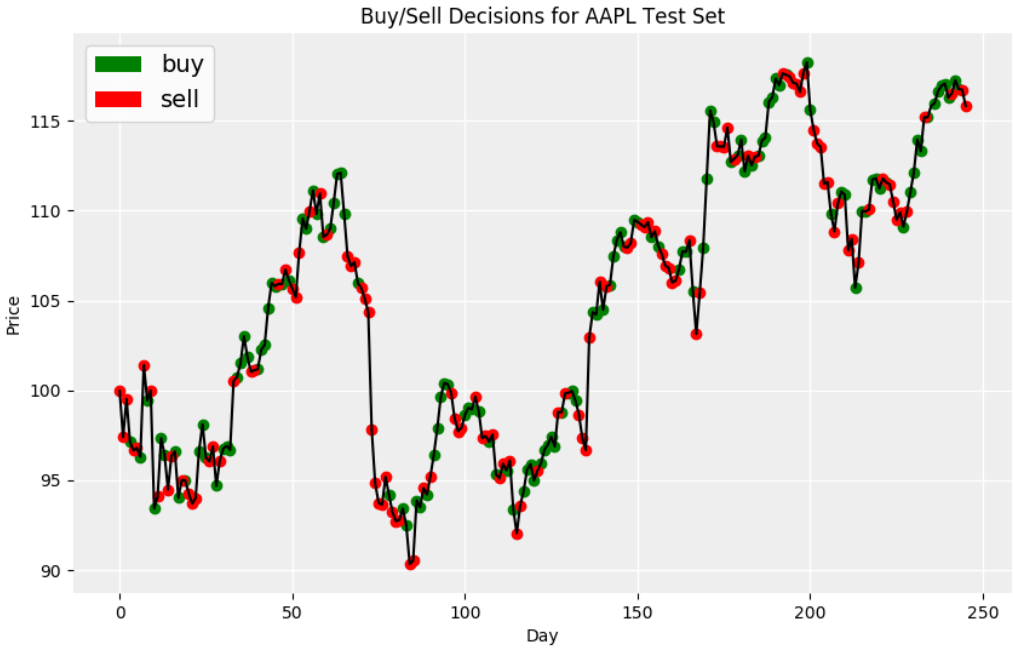


Figure 92 Buy/Sell Decisions for AAPL Test Set

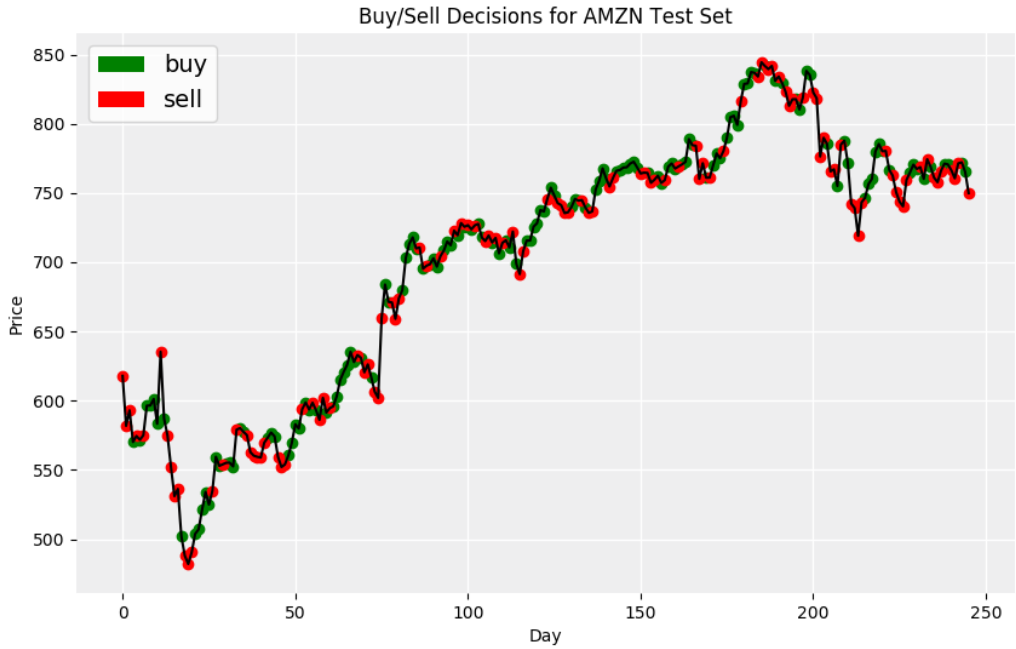


Figure 93 Buy/Sell Decisions for AMZN Test Set

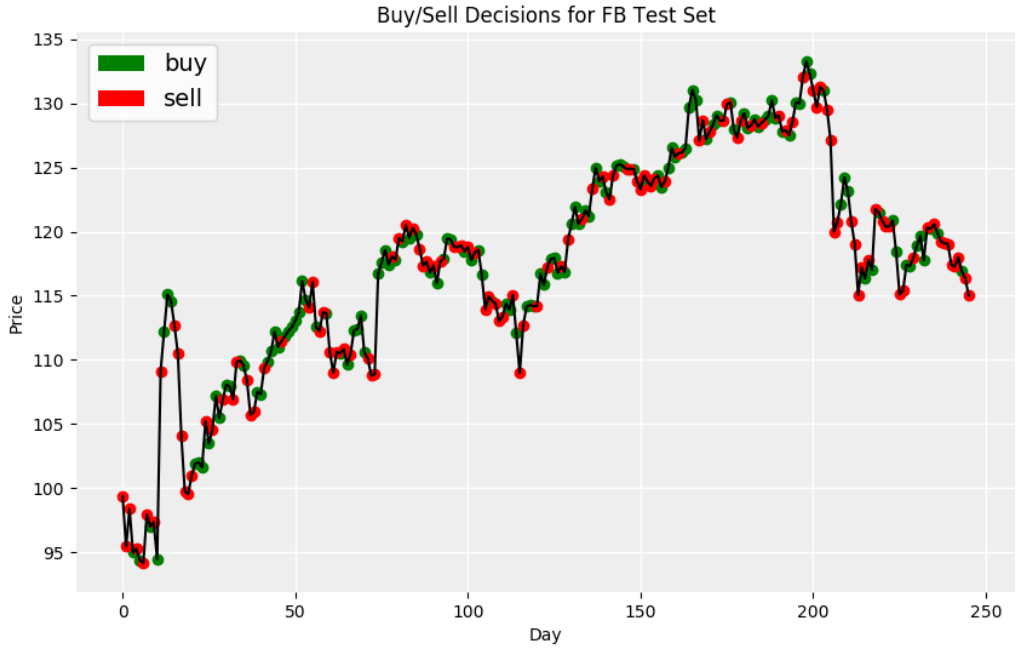


Figure 94 Buy/Sell Decisions for FFB Test Set

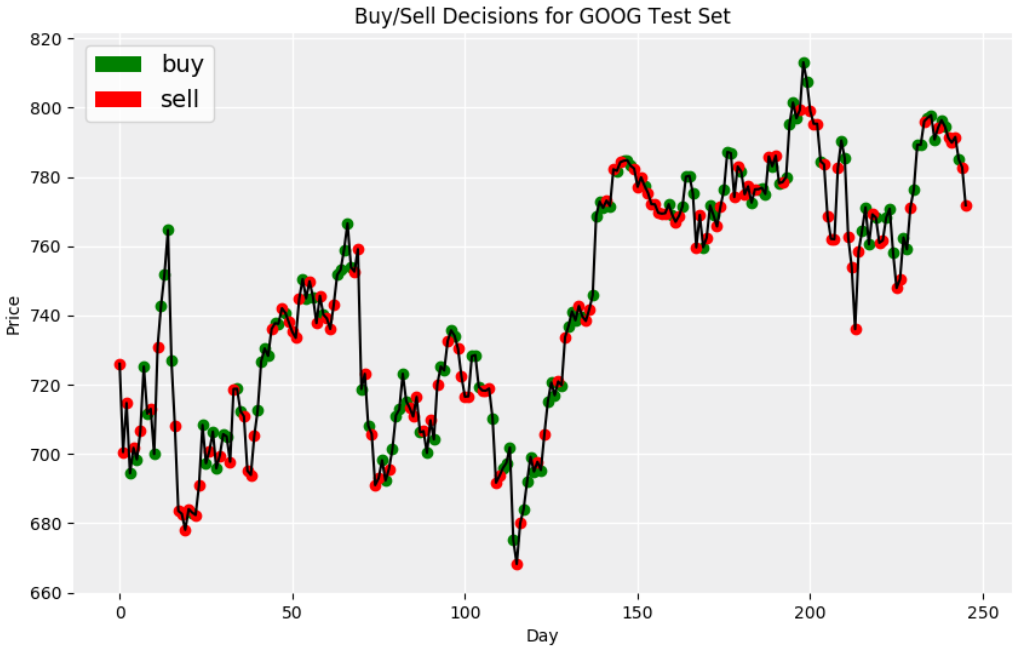


Figure 95 Buy/Sell Decisions for Google Test Set

11 Summary

Time series forecasting is a very intriguing field to work with, particularly when it comes to stock predictions eagerness to learn and put more thoughts which eventually could be used for making better financial investments. Predications made here is riskier for performing direct trading in share market based on the values predicted, however this would be surely helping to do some swing trading with least risk.

The project work explores the LSTM Neural Network which itself is a very vast topic for research work. In this project predicted the future closing stock price using historical stock data in combination with the sentiments of news articles and twitter data. Started with data collection of the historical stock price, twitter and news data by web scraping and through various data sources. Data Pre-processing helped in removing unwanted records and carry out aggregations to extract useful features.

Sentiment analysis has been performed on the news and twitter data to generate weighted average sentiments. The usefulness of the features is validated by performing correlation analysis. Based on feature engineering, identified critical are used as an input to our LSTM model and predict the future closing stock prices. The best results were obtained by using the stock data along with the new data. On the other hand, when including twitter sentiments, the error was higher which indicates that the vast number of tweets were not directly related to Apple's success which can interfere with predictions.

To summarize our results:

- Best result is obtained when combination of Stock, Twitter and News Data are used.
- Twitter Data alone did not provide better stock predictions.
- Including Twitter data with Stock and News data combined had a negative effect on RMSE.
- A Learning Rate of 0.002 was found to be most effective.
- A lookback of 15 was found to be the best given our data sources.
- Advanced NLP techniques can be used to improve the model predictions using twitter data.

12 Future Work

While some of the stock prices when used for training the model, resulted in very close graph with respect to the original stock prices, some of the company's stock prices didn't result in very good trained model.

To overcome this, the network parameters needs to be tuned properly. A better benchmark would be required to precisely choose number of units, optimizer learning rate, number of previous stock prices, dropout probability, etc.

The tuning parameters needs to be provided through a configuration file, enabling us to avoid compilation of code, when any of the network parameters changes.

The front-end UI needs to be updated for better look and feel and enhance user experience. The response received from back-end needs to be done using existing popular framework models.

Bibliography

1. Makeshwar, M.S., Rajgure, N.K. & Pund, M. (2010). Object Identification using Neural Network. International Journal of Computer Science and Application , pp.29–32.
2. Hertz, J., Krogh, A. and Palmer, R.G. (1991), Introduction to the Theory of Neural Computation, Redwood City, CA: AddisonWesley.
3. Kohavi, R. (1995). A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection. In International Joint Conference on Artificial Intelligence. San Francisco, CA: Morgan Kaufmann Publishers.
4. Herrera, J.L. (1999). Time Series Prediction Using Inductive Reasoning Techniques. Instituto de Organizacion y Control de Sistemas Industriales.
5. Bachelier (1900), Cootner (1964) and Fama(1965), Theory of Speculation
6. www.finace.yahoo.com
7. J.-H. Wang and J.-Y. Leu, "Stock Market Trend Prediction Using ARIMA-Based Neural Networks," The 1996 IEEE International Conference on Neural Networks, Washington DC, 3-6 June 1996, pp. 2160-2165.
8. "Predicting stock market index using fusion of machine learning techniques" by Jigar Patel, Sahil Shah, Priyank Thakkar , K Kotecha Computer Science & Engineering Department, Institute of Technology, Nirma University, Ahmedabad, Gujarat, India, Article history: Available online 25 October 2014, http://dx.doi.org/10.1016/j.eswa.2014.10.031 0957-4174/ 2014 .
9. Jiahong Li, Hui Bu and Junjie Wu, "Sentiment-aware stock market prediction: A deep learning method," *2017 International Conference on Service Systems and Service Management*, Dalian, 2017, pp. 1-6.
10. Novel Approaches to Sentiment Analysis for Stock Prediction by Chris Wang, Yilun Xu, Qingyang Wang Stanford University chrwang, ylxu, iriswang @ stanford.edu
11. Sun, Haonan, et al. "Stacked Denoising Autoencoder Based Stock Market Trend Prediction via K-Nearest Neighbour Data Selection." International Conference on Neural Information Processing. Springer, Cham, 2017.
12. Predicting Stock Prices using Social Media Team: Stock Brokers | CMPT 733 - Programming for Big Data 2 | April 08, 2018 | By Mihir Gajjar, Gaurav Prachchhak, Tommy, Betz, Veekesh Dhununjoy

13. International Journal of Computer Science & Information Technology (IJCSIT) Vol 8, No 3, June 2016 , STOCK TREND PREDICTION USING NEWS SENTIMENT ANALYSIS Kalyani Joshi[1], Prof. Bharathi H. N.[2], Prof. Jyothi Rao[3] .
14. https://github.com/omerbsezer/LSTM_RNN_Tutorials_with_Demo
15. Proceedings of the International MultiConference of Engineers and Computer Scientists 2019, IMECS 2019, March 13-15, 2019, Hong Kong, Stock Market Trend Prediction with Sentiment Analysis based on LSTM Neural Network, Xu Jiawei, Tomohiro Murata.
16. Avramov, Doron. (2001). Stock Return Predictability and Model Uncertainty. Journal of Financial Economics. 64. 423-458. 10.1016/S0304-405X(02)00131-9.
17. Journal of Multinational Financial Management,Volume 13, Issues 4–5, December 2003, Pages 443-463,Statistical and economic significance of stock return predictability: a mean–variance analysis, Steven X.Wei1ChuZhang
18. International Review of Economics & Finance,Volume 10, Issue 4, December 2001, Pages 353-368,Nonlinear predictability of stock market returns: Evidence from nonparametric and threshold models, David GMcMillan
19. Knowledge-Based Systems,Volume 50, September 2013, Pages 151-158 ,Posterior probability model for stock return prediction based on analyst's recommendation behavior, Jiang jia Duana Hong Zhong, Liua Jian ping, Zengb.
20. Journal of Banking & Finance, Volume 34, Issue 3, March 2010, Pages 509-521, The degree of financial liberalization and aggregated stock-return volatility in emerging markets,MehmetUmutluaLeventAkdenizbAslihanAltay-Salihb
21. International Review of Financial Analysis ,Volume 18, Issues 1–2, March 2009, Pages 1-11.
22. The Externalities of High Frequency Trading, Chen Yao, The Chinese University of Hong Kong (CUHK) - CUHK Business School, JH Hill, University of Illinois at Urbana-Champaign, Date Written: August 7, 2013.
23. Ticknor J.L.,A Bayesian regularized artificial neural network for stock market forecasting, Expert Syst. Appl., 40 (14) (2013), pp. 5501-5506.
24. Rout A.K., Biswal B., Dash P.K.,,A hybrid FLANN and adaptive differential evolution model for forecasting of stock market indices Int. J. Knowl.-Based Intell. Eng. Syst., 18 (1) (2014), pp. 23-41
25. Zhong X., Enke D, Forecasting daily stock market return using dimensionality reduction, Expert Syst. Appl., 67 (2017), pp. 126-139.

26. Vargas M.R., de Lima B.S., Evsukoff A.G.,Deep learning for stock market prediction from financial news articles,Proceedings of IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (2017), pp. 60-65.
27. Gupta A., Dhingra B.,Stock market prediction using hidden markov models,Proceedings of Students Conference on Engineering and Systems (2012), pp. 1-4.
28. Chang P.C.,A novel model by evolving partially connected neural network for stock price trend forecasting,Expert Syst. Appl., 39 (1) (2012), pp. 611-620.
29. Pang X., Zhou Y., Wang P., Lin W., Chang V.,An innovative neural network approach for stock market prediction,J. Supercomput. (2018), pp. 1-21.
30. Atsalakis G.S., Dimitrakakis E.M., Zopounidis C.D, Elliott Wave Theory and neuro-fuzzy systems, in stock market prediction: The WASP system Expert Syst. Appl., 38 (8) (2011), pp. 9196-9206.
31. Chatzis S.P., Siakoulis V., Petropoulos A., Stavroulakis E., Vlachogiannakis N. Forecasting stock market crisis events using deep and statistical machine learning techniques, Expert Syst. Appl., 112 (2018), pp. 353-371
32. Shen W., Guo X., Wu C., Wu D.Forecasting stock indices using radial basis function neural networks optimized by artificial fish swarm algorithm, Knowledge Based Syst., 24 (3) (2011), pp. 378-385.
33. Asadi S., Hadavandi E., Mehmanpazir F., Nakhostin M.M.Hybridization of evolutionary Levenberg–Marquardt neural networks and data pre-processing for stock market prediction, Knowledge Based Syst., 35 (2012), pp. 245-258.
34. Adebiyi A.A., Ayo C.K., Otokiti S.O.Fuzzy-neural model with hybrid market indicators for stock forecasting, Int. J. Electron. Finance, 5 (3) (2011), pp. 286-297
35. Hsieh T.J., Hsiao H.F., Yeh W.C.Forecasting stock markets using wavelet transforms and recurrent neural networks: An integrated system based on artificial bee colony algorithm,Appl. Soft Comput., 11 (2) (2011), pp. 2510-2525
36. Xie X.K., Wang H.Recurrent neural network for forecasting stock market trend, Proceedings of International Conference on Computer Science, Technology and Application (2017), pp. 397-402
37. Chen W., Zhang Y., Yeo C.K., Lau C.T., Lee B.S, Stock market prediction using neural network through news on online social networks Proceedings of International Conference on Smart Cities (2017), pp. 1-6

38. Oztekin A., Kizilaslan R., Freund S., Iseri A. A data analytic approach to forecasting daily stock returns in an emerging market, European J. Oper. Res., 253 (3) (2016), pp. 697-710
39. Zhang X., Qu S., Huang J., Fang B., Yu P. Stock market prediction via multi-source multiple instance learning IEEE Access (2018)
40. Arévalo R., García J., Guijarro F., Peris A. A dynamic trading rule based on filtered flag pattern recognition for stock market price forecasting Expert Syst. Appl., 81 (2017), pp. 177-192
41. Ariyo A.A., Adewumi A.O., Ayo F.Stock price prediction using the ARIMA model, Procedings of Computer Modelling and Simulation (2014), pp. 106-112
42. Srinivasan P., Ibrahim P. Forecasting stock market volatility of BSE-30 index using GARCH models Asia Pac. Bus. Rev., 6 (3) (2010), pp. 47-60
43. Nanda S.R., Mahanty B., Tiwari M.K.Clustering Indian stock market data for portfolio management, Expert Syst. Appl., 37 (12) (2010), pp. 8793-8798
44. Systematic analysis and review of stock market prediction techniques, Author links open overlay panelDattatray P.GandhmalK.Kumar, Computer Science Review, Volume 34, November 2019, 10019, https://doi.org/10.1016/j.cosrev.2019.08.001
45. Ramon Lawrence. "Using Neural Networks to Forecast Stock MarketPrices". Neural Networks in the Capital Markets, chapter 10, pages 149–162. John Wiley and Sons, 1995.
46. Vivek Rajput, Sarika Bobade . "Stock Market Prediction Using Hybrid Approach". International Conference on Computing, Communication and Automation (ICCCA2016).
47. Li Xiong, Yeu Lu (2017). "Hybrid ARIMA-BPNN Model for Time Series Prediction of the Chinese Stock Market". 2017 3rd International Conference on Information Management.
48. Manuel R. Vargas,Carlos E.M. dos Anjos,Gustavo L.G. Bichara,Alexandre G. Evsukoff (2018). "Deep Learning for Stock Market Prediction Using Technical Indicators and Financial News Articles". 2018 International Joint Conference on Neural Networks (IJCNN).
49. Marios Mourelatos, Thomas Amorgianiotis, Christos Alexakos, Spiridon Likothanassis (2018). "Financial Indices Modelling and Trading Utilizing Deep Learning Techniques". 2018 Innovations in Intelligent Systems and Applications (INISTA)
50. Mohammed Asiful, Hossain, Rezaul Karim, Ruppa THulasiram, Neil D.B Bruce, Yang Wang (2018). "Hybrid Deep Learning Model for Stock Price Prediction". 2018 IEEE Symposium Series on Computational Intelligence (SSCI).
51. J.J. Wang, J. Z. Wang, Z. G. Zhang, and S. P Guo (2012)."Stock index forecasting based on a hybrid model". Omega, vol. 40, pp. 758-766.

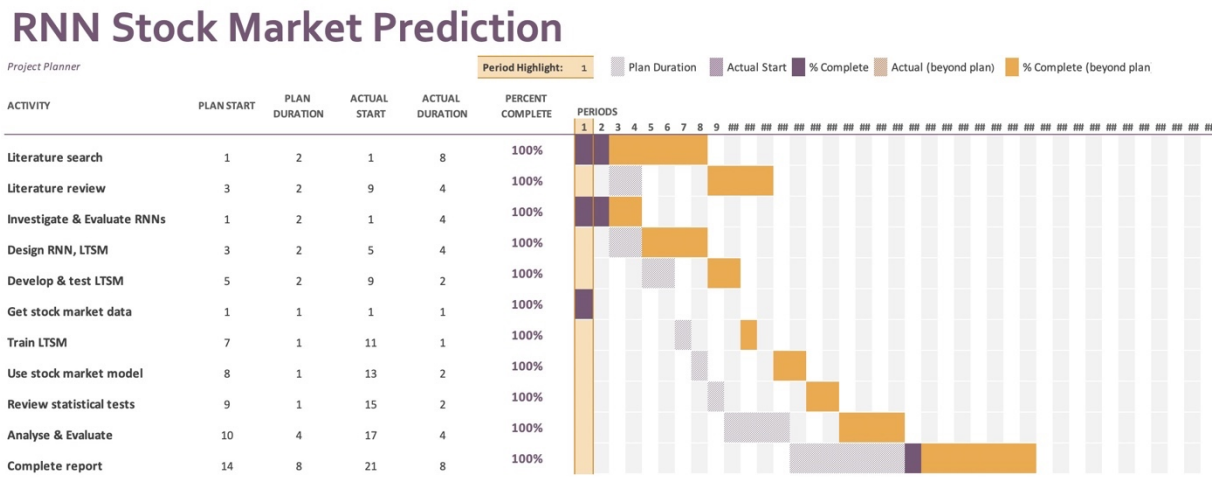
52. C. Narendra Babu and B. Eswara Reddy (2014). "A moving-average filter based hybrid ARIMA-ANN model for forecasting time series data". Applied Soft Computing, vol. 23, pp. 27-38.
53. Graves (2012). “Supervised Sequence Labelling with Recurrent”. Studies in Computational Intelligence, Springer.
54. Warbos, Paul J (1990). “Backpropagation through time: what it does and how to do it”. Proceedings of the IEEE 78.10, pp.1550-1560.
55. C. Narendra Babu and B. Eswara Reddy, “Prediction of selected Indian stock using a partitioning-interpolation based ARIMAGARCH model,” Applied Computing and Informatics, vol.11, pp. 130-143, July 2015.
56. R. K. Nayak, D. Mishra, and A. K. Rath, “A Naïve SVM-KNN based stock market trend reversal analysis for Indian benchmark indices,”Applied Soft Computing, vol. 35, pp. 670-680, October 2015.
57. C. H. Su and C. H. Cheng, “A hybrid fuzzy time series model based on ANFIS and integrated nonlinear feature selection method for forecasting stock,” Neurocompting, vol. 205, pp. 264-273, September 2016.
58. C. M. Anish and B. Majhi, “Hybrid nonlinear adaptive scheme for stock market prediction using feedback FLANN and factor analysis,” Journal of the Korean Statistical Society, vol. 45, pp.64-76, March 2016.
59. L. Y. Wei, "A hybrid ANFIS model based on empirical mode decomposition for stock time series forecasting," Applied Soft Computing, vol.42, pp. 368-376, May 2016.
60. K. Cho, B. van Merrienboer, D. Bahdanau, and Y. Bengio , “On the properties of neural machine translation: Encoder-decoder approaches”. arXiv preprint arXiv: 1409.1259,2014.
61. G. P. Zhang,"Time series forecasting using a hybrid ARIMA and neural network model". Neurocomputing, vol. 50, pp. 159-175,2003.
62. Mohammad Obaidur Rahman, Md. Sabir Hossain, Ta-Seen Junaid, Md. Shafiul Alam Forhad, Muhammad Kamal Hossen , “Predicting Prices of Stock Market using Gated Recurrent Units (GRUs) Neural Networks”.,IJCSNS International Journal of Computer Science and Network Security, VOL.19 No.1,2019.
63. Bengio, Yoshua, S. Patrice, F. Paolo, “Learning long-term dependences with gradient descent is difficult”. Neural Networks, IEEE Transactions on 5.2, pp.157-166,1994.
64. Cheng, Li-Chen, Yu-Hsiang Huang, and Mu-En Wu. "Applied attention-based LSTM neural networks in stock prediction." 2018 IEEE International Conference on Big Data (Big Data). IEEE, 2018.

65. Khare, Kaustubh, et al. "Short term stock price prediction using deep learning." 2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT). IEEE, 2017.
66. L. Di Persio and O. Honchar, “Artificial neural networks architectures for stock price prediction: Comparisons and applications,” International Journal of Circuits, Systems and Signal Processing, vol. 10, pp. 403– 413, 2016.
67. M. F. Dixon, D. Klabjan, and J. H. Bang, "Classification-Based Financial Markets Prediction Using Deep Neural Networks," arXiv preprint, arXiv:1603.08604v2 , June 2017.
68. Wenping Zhang,Chunping Li,Yunming Ye,Wenjie Li and Eric W.T. Ngai , “Dynamic Business Network Analysis for Correlated Stock Price Movement Prediction”, IEEE Intelligent Systems Volume: 30 , Issue: 2 , Mar.-Apr. 2015.
69. Ouahilal, M., El Mohajir, M., Chahhou, M., & El Mohajir, B. E. “Optimizing stock market price prediction using a hybrid approach based on HP filter and support vector regression”. 2016 4th IEEE International Colloquium on Information Science and Technology(CiSt).
70. Price Prediction of Share Market using Artificial Neural Network (ANN), [www.ijcaonline.org], by Zabir Haider Khan , Tasnim Sharmin Alin , Md. Akter Hussain
71. Rout A.K., Biswal B., Dash P.K. A hybrid FLANN and adaptive differential evolution model for forecasting of stock market indices Int. J. Knowl.-Based Intell. Eng. Syst., 18 (1) (2014), pp. 23-41.
72. Chakravarty S., Dash P.K. A PSO based integrated functional link net and interval type-2 fuzzy logic system for predicting stock market indices, Appl. Soft Comput., 12 (2) (2012), pp. 931-941
73. Xu B., Zhang D., Zhang S., Li H., Lin H. Stock market trend prediction using recurrent convolutional neural networks
74. Badge J.Forecasting of indian stock market by effective macro-economic factors and stochastic model J. Stat. Econom. Methods, 1 (2) (2012), pp. 39-51
75. Chen M.Y., Chen B.T. A hybrid fuzzy time series model based on granular computing for stock price forecasting
76. Vatsal H. Shah - Foundations of Machine Learning| Spring, 2007 - bigquant.com
77. Pang, Xiongwen, et al. "An innovative neural network approach for stock market prediction." The Journal of Supercomputing (2018): Pages 1-21.
78. On the Properties of Neural Machine Translation: Encoder-Decoder Approaches, September 2014 , Kyunghyun Cho

79. Stock Market prediction using news headlines, Joshua van Kleef, Valerie Scholten and Emiel Stoelinga, Cognitive Computational Modeling of Language and Web Interaction June 19, 2017
80. LSTM Neural Networks for Language Modeling, Conference Paper, September 2012, Martin Sundermeyer
81. https://www.deeplearningwizard.com/deep_learning/practical_pytorch/pytorch_lstm_neuralnetwork/
82. "Analyzing Stock Market Movements Using Twitter Sentiment Analysis",
    Nirali Patel, Computer Science Illinois Institute of Technology Chicago npate104@hawk.iit.edu
    Jaimin Sanghvi, Computer Science Illinois Institute of Technology Chicago jsanghv2@hawk.iit.edu
83. https://www.deeplearningwizard.com/deep_learning/practical_pytorch/pytorch_lstm_neuralnetwork/
84. https://cs231n.github.io/convolutional-networks/
85. http://colah.github.io/posts/2015-08-Understanding-LSTM

Appendix A

Research Plan



Appendix B

Research Proposal Document

Automated Algorithmic Trading
(automated algorithmic stock trading)

by

Arun Prasath S
M.Sc (Data Science)

Mentored by : Vibhor

Abstract

Stock market prediction is an act of trying to determine the future value of a stock other financial instrument traded on a financial exchange. Recent days, several researchers are using machine learning models to predict the stock price from various financial instruments. Usually the stock markets have high uncertainties and are affected by interrelated economic, political factors at both geographical levels. The key to successful stock market forecasting is achieving best results with minimum required input data against all the factors influencing the stock price.

The prediction of the share market values is of great importance to help in maximizing the profit of stock purchase while keeping the risk low. Using features like the latest announcement about an organization, their quarterly revenue results etc., machine learning techniques have the potential to unearth patterns and insights, that we dint not see before and these can be used to make accurate predictions.

High level of accuracy and precision is the key factor in predicting a stock market. The technical, fundamental or the time series analysis is used by most of the stockbrokers while making the predictions. Nevertheless, these methods cannot be trusted fully, so there is a necessity to provide the supportive method for stock market prediction. Artificial Neural Network (ANN) was found to be the most practical consideration. Neural network models having the features and customisable parameters makes it possible to implement wide number of features along with the cross-validation sets.

The analysis of historical stock data sets and extracting certain trends would help to predict the future value of the stock. For prediction, a recurrent neural network will be fed with a pre-processed historical value of a stock value for getting trained on the time series. Once trained the neural network layer would be used for making some predictions for next trading day(s). Based on these predictions a potential Buys and Sells for given stock can be generated for a potential swing trade.

Table of Contents

5 Introduction

Accurate prediction of prices of financial instruments is essential to take better investment decisions with minimum risk. In view of the complexity of the financial time series data, resulting from a huge number of factors which could be economic or political [1-2]. Machine learning and soft computing methods have been used by several authors in the last two decades for financial time series forecasting. The nonlinearity of financial time series motivated many authors to use back-propagation (BP) neural network due to its simple architecture design yet powerful problem-solving ability. However, this method has following drawbacks: presence large number of controlling parameters, over-fitting problem, slow convergence and struck to local minima (as the back-propagation algorithm is obtained by minimizing a nonlinear error function) [3].

People who trade and make investment in stock market directly by purchasing shares or indirectly by making investments through Mutual Funds AMCs have a very staunch desire to make large money. As the technology is advancing, the opportunity to gain a steady fortune from the share market is increased.

Unfortunately, predicting how the stock market will perform is one of the most challenging things to do. The fluctuation of the stock market is highly volatile because several factors are involved in the prediction – physical factors vs. physiological, rational and irrational behavior, etc. All these factors combine to make share prices very difficult to predict with a high degree of accuracy.

Several technical analysis tools have been used by data and finance analysts. These tools and techniques need lot of technical details and news flows for individual stock. Even after gathering lot of technical details and analyzing several data points, they still would not give reliable predictions and thereby, causing more confusions among beginner.

The analysis of historical stock data sets and extracting certain trends would help to predict the future value of the stock. Based on these predictions a potential Buys and Sells for given stock can be generated for a potential swing trade.

6 Background and related research

4.1 Introduction to Stock Market Prediction

Recent research in application of machine learning is growing in the area of Machine Learning Algorithms for analyzing price patterns and predicting stock prices and index changes. Nowadays algorithmic trading or software-based trading systems are getting traction due to their nature in predicting prices based on various situations and conditions, thereby helping them in making instantaneous investment decisions and sizing their trade positions accordingly.

Stock Prices are considered to be very dynamic and susceptible to quick changes because of the underlying nature of the financial domain and in part because of the mix of known parameters (**Previous Day's Closing Price, P/E Ratio etc.)** and unknown factors (like **Election Results, Rumors etc**.) An intelligent trader would predict the stock price and buy a stock before the price rises, or sell it before its value declines. Though it is very hard to replace the expertise that an experienced trader has gained, an accurate prediction algorithm can directly result into high profits for investment firms, indicating a direct relationship between the accuracy of the prediction algorithm and the profit made from using the algorithm.

Typical Stock Chart : A typical stock chart for a share value would look like:

2.2 Fundamental Analysis Based Stock Prediction

Introduction to fundamental Analysis :The fundamental analysis involves the in-depth analysis of a company's performance and the profitability to measures it's intrinsic value by studying the company physically in terms of its product sales, man power quality, infrastructure, profitability on investment. It uses revenues, earnings, future growth, return on equity, profit margins, and other data to determine a company's underlying value and potential for future growth. To a fundamentalist, the market price of a stock tends to move towards its "real value" or "intrinsic value".

The advantages of fundamental analysis :

- The advantages of fundamental analysis are its systematic approach and its ability to predict changes before they show up on the charts.
- Fundamental analysis is a superior method for long-term stability and growth.

Disadvantages of fundamental analysis :

- It becomes harder to formalize all this knowledge for purposes of automation (with a neural network for example), and interpretation of this knowledge may be subjective.
- It is hard to time the market using fundamental analysis.

Important Ratios for fundamental analysis :

The concept of stock valuation can be understood mainly by knowing some ratios. None of the valuation discussed below are superior to each other. Each of them has their own importance. Any share which does not reflect all these parameters as healthy means it is not a good buy.

- **The Price-to-Book Ratio (P/B):** Book value is that value of a company, which the owner is likely to gather if they decide to liquidate (sell off in dire straits) the company.
- **Price-to-Earnings Ratio (P/E):** P/E Ratio compares the market price with the EPS. Where, EPS (Earning per share) is the company's net earnings by the numbers of outstanding shares of the stock. Higher the P/E ratio, more people are convinced to pay high for that share expecting higher growth in coming future.

- **The PEG Ratio:** The PEG ratio is the P/E ratio of a company by the year-on-year growth rate of its earnings. The lower the value of your PEG ratio, the better the deal you're getting for the stock's future estimated earnings.

- **Dividend Yield:** The dividend yield shows how much payout you're getting for your money. It is the stock's annual dividend payout by the stock's price.

- **Debt to Equity Ratio:** This should not be more than 1, and less than 1 indicates company has very less debt. This is very important during market down trend as company has to pay lots of interest beside low profitability. So, it is a good sign, if company has less debt to Equity Ratio.

- **Returns on Equity (ROE):** It is used as a general indication of the company's efficiency, in other words, how much profit it is able to generate given the resources provided by its stockholders. Investors usually look for companies with ROE that are high and growing. No Element Stands Alone. P/B, P/E, PEG, Dividend Yield, Debt to Equity Ratio is too narrowly focused to stand alone as a single measure of a stock. By combining these methods of valuation, you can get a better view of a stock's worth.

2.3 Technical Analysis based Stock Prediction

Introduction to Technical Analysis : Technical analysis is a method of evaluating stocks by analyzing statistics generated by market activity, past prices, and volume. It looks for peaks, bottoms, trends, patterns, and other factors affecting a stock's price movement. Future values of stock prices often depend on their past values and the past values of other correlated variables. Technical analysis looks for patterns and indicators on stock charts that will determine a stocks future performance [4]. However, it is used by approximately 90% of the major stock traders. Despite its widespread use, technical analysis is criticized because it is highly subjective. Different individuals can interpret charts in different manners.

Recently, neural networks have been successfully applied in time-series problems to improve multivariate prediction ability. Neural networks have good generalization capabilities by mapping input values and output values of given patterns. Neural networks are usually robust against noisy or missing data, all of which are highly desirable properties in time series prediction problems. Various neural network models have already been developed for the stock market analysis.

The advantages of technical analysis :

- It is used by approximately 90% of the major stock traders.
- It is also used to analyze the stock for shorter period.

Disadvantages of technical analysis

- Despite its widespread use, technical analysis is criticized because it is highly subjective.
- Different individuals can interpret charts in different manners.

Important Parameters for technical analysis :

In technical analysis of stock market data 52 different parameters, indicators and oscillators have been defined. Even though each indicator provides some additional information about the stock, using each one of them will make the system complex and slow. This would require at least 2^52 rules for the ANFIS structure. Hence there is a need to identify the parameters (feature vectors of the financial data) that most closely predict the nature of the movement without increasing the system complexity [5], [6].

Moving Average (MA): This is perhaps the oldest and the most widely used technical indicator. It shows the average value of stock price over time. The shorter the time period, the more reactionary a moving average becomes. A typical short term moving average ranges from 5 to 25 days, an intermediate-term from 5 to 100, and long-term 100 to 250 days.

Exponential Moving Average (EMA): An exponential moving average gives more weight to recent prices and is calculated by applying a percentage of today's closing price to yesterday's moving average. The longer the period of the exponential moving average, the less total weight is applied to the most recent price. The advantage to an exponential average is its ability to pick up on price changes more.

Moving Average Convergence/Divergence (MACD): It is the difference between two exponential moving averages, normally one short moving average and one long moving average.

Relative Strength Index (RSI): An oscillator, introduced by J. Welles Wilder, Jr., is based upon the difference between the average gains vs. the average loss over a given period. The RSI compares the magnitude of a stock's recent gains to the magnitude of its recent losses.

2.4 Sentiment analysis for Stock Prediction :

The task of mining subjective feelings expressed in the text [7], has been found to play a significant role in many applications such as product recommendations, healthcare, politics, and in surveillance [8]. The expression of moods and emotions in a large amount of social media data is very important in gauging the opinions of investors [9]. Twitter data has recently gained traction in predicting stock prices based on public sentiments [10]. This is because Twitter data is already public and thus automatically and very quickly influences stock prices. However, quality measures must be put in place in order to use Twitter data because of its widespread use by malicious users to promote or devalue products, services, ideas, and ideologies [11]. The reliability of the models for stock market prediction is important as it can directly affect economy and lead to financial loss. News data are sources of information that can be relatively relied upon.

For stock market prediction selection of input features is important task. Most of the machine learning techniques are use technical indicators as input. In the following section we see some of the techniques used by researchers and input features used by them for prediction.

2.5 Literature Review:

For stock market prediction selection of input features is important task. Most of the machine learning techniques are use technical indicators as input. In the following section we see some of the techniques used by researchers and input features used by them for prediction.

Zabir Haider Khan, Tasnim Sharmin Alin, Md. Akter Hussain[12] have used ANN for stock market prediction and have used 5 fundamental input variable which are general index(GI), Net Asset Value(NAV), profit per earning(P/E) ratio, Earnings per share(EPS) and share volume. They have applied these parameters to NN and compared their outcome.

Ramin Rajabioun and Ashkan Rahimi-Kian[13] created a genetic programming (GP) based prediction model which considers interaction between companies and results generated by five agents each created with different strategy like buy/sell maximum allowed stock, use

of Mean Variance analysis(MVA), Random walk theory, MVA with less risk, MVA with risk taking resp.to predict future price. They compared results of GP with MLP and Neuro-Fuzzy network to show that GP performs better. They have used daily opening price, daily closing price, daily highest price, daily lowest price and daily exchange volume as input features which comes under technical indicators.

Tsai, C.-F. and Wang, S.-P.[13] have shown how Hybrid machine learning methods outperforms that techniques alone. They combined ANN with Decision Tree(DT) to improve accuracy of ANN in the task of prediction. They created four models ANN, DT, ANN+DT, DT+DT and verified these models against data collected from TEJ database. After collecting all related variables, they applied Principal Component Analysis(PCA) for filtering out unwanted or unrelated variables and finally 53 variables are selected. After apply all four models on data, Hybrid system gives better results than other models. In particular they have found out twelve different decision rules for predicting rise or fall. From decision tree we can see most of the indicators used in rules are microeconomic indicators like Export growth rate, product export, import amount from USA, export PI increase rate, import growth rate, CPI, etc. **Nguyen Lu Dang Khoa1, Kazutoshi Sakakibara2 and Ikuko Nishikawa2[14]** applied back propagation algorithm with time and profit base adjusted weight factors. They have used Feed-forward neural network and simple Recurrent neural network for modified training algorithm. Results show that recurrent neural network with modified back propagation performs better. In this paper they tried to select as few independent inputs as possible. They have used input indicators as inflation rate, GDP, relative strength index, directional index, daily high, daily low, closing price and moving averages.

Y.R.Ramesh Kumar and Prof.A.Govardhan, [15] try to use user experience which he gains through market along with the technical analysis and precious methods to enhance the profit. They claim user forget his previous experience and make same mistakes again and again because his/ her judgments are purely instinctive. In this task they first use neural network with technical indicator like opening price, highest price, closing and lowest price and supportive data warehouse as inputs, to generate a buy/sell signal. Supportive data ware house contains previous experiences and information about market movement which will help in prediction. After generating signal transaction is made its effect is recorded as profit or loss along with required information. They have compared this to other techniques to find out better method.

Wei Huang, Kin Keung Lai, Yoshiteru Nakamori, Shouyang Wang, Lean Yu[16] have given a detail description of which input variables can be used for predicting stock market index price. They have also applied different ANN models for prediction and compared their results. According to them there are two approaches for prediction first one is relationship between market price and microeconomic indicators. Second approach takes into account non-linear relation between stock price, trading volume and dividends. They have surveyed on application of ANN to stock market index forecasting where they tried to find different variables that have impact on stock index.

Chen[17] has provided information about how variables like default spread, term spread, one-month T-bill rate, lagged industrial production growth rate and dividend–price ratio affects stock price index. He also has shown their ability to predict future stock price.

Fama and French[18] identified three common risk factors: the overall market factor, factors related to firm size and book-to-market equity, which seem to explain average returns on stocks and bonds.

Kohara[13] have used five microeconomic indicators for prediction.

TOPIX(Tokyo stock exchange price index) he has also used US dollar to Yen exchange rate, three-month interest rate, crude oil price and New York Dow Jones average of closing price of 30 industrial stocks because according to him TOPIX stock prices are often influenced by New York stock prices. There is a set of macroeconomic indicators that can be used for stock index prediction term structure of interest rates (TS), short term interest rate (ST), long term interest rat (LT), consumer price index (CPI), industrial production (IP), government consumption (GC), private consumption (PC), gross national product (GNP), gross domestic product (GDP) and they are easily available also[10].

Adebiyi Ayodele A., Ayo Charles K., Adebiyi Marion O., and Otokiti Sunday O.[19] have suggested use of hybridized parameters i.e. variables of both technical and fundamental analysis. They have selected total 18 inputs and applied them to different neural networks (such as 18-24-1, 18-18-1, 18-22-1) and compared results with those networks with only technical variables. Out of 18 input variables 10 are technical variables which are opening price, closing price, highest price, lowest price and trading volume of last two days. Remaining 8 input variables are fundamental variables which are price per annum of last two years, news/rumors to buy/sell stock of last two days, book value of last two years and

financial status of company for last two years. In this paper they have shown how hybridized parameters outperforms technical parameters.

Robert P. Schumaker, Hsinchun Chen[20] tried to find out influence of news article on stock price. They have analyzed news articles related to the stock and find out the meaning of article and supplied it as input parameter along with stock quotes to Support Vector Machine(SVM) and error is calculated by Mean Square Error(MSE).

Leonardo C. Martinez, Diego N. da Hora, Joao R. de M. Palotti, Wagner Meira Jr. and Gisele L. Pappa [21] have given another approach for increasing profit by predicting highest and lowest price instead of closing price. According to them we can make more than one transaction per day by observing current stock price and comparing with highest/lowest price. They have used 33 input variables which are applied to ANN. Out of 33 input variables 10 are lowest and highest price of last 5 days, 10 are opening and closing price of last 5 days, 2 are Exponential Moving Average(EMA) of highest and lowest price of last 5 days, 2 are EMA of opening and closing price of last 5 days , 4 are Bollinger band(BB) of highest and lowest price of last 5 days, 4 are BB of opening and closing price of last 5 days and 1 for opening price of current day. They have compared their system with other existing systems and have shown how this technique can maximize use profit.

Buys and Sells

Since shares are traded everyday (except over National Holidays and weekends), there values change daily. It's always beneficial to buy a stock at lower values and sell when its value is above the purchase price. Like in above Figure, a buy at around Nov 16th, 2018 at a price around 650 Rs a stock and selling same stock today at 815 Rs. / a stock would have resulted in a profit of 165 Rs. /- a stock, resulting in some 23 % gain in just 9 months! While trading or investment normally people buy several quantities of stock. Like buying then selling, there's almost a reverse strategy, i.e. sell then buy the stock. However, this only little expert traders do but the idea here is same, i.e. here a trader would sell at higher rise and buy same quantity when the stock value decrease then the selling Price.

3. Research Questions (If any)

4. Aim and Objectives

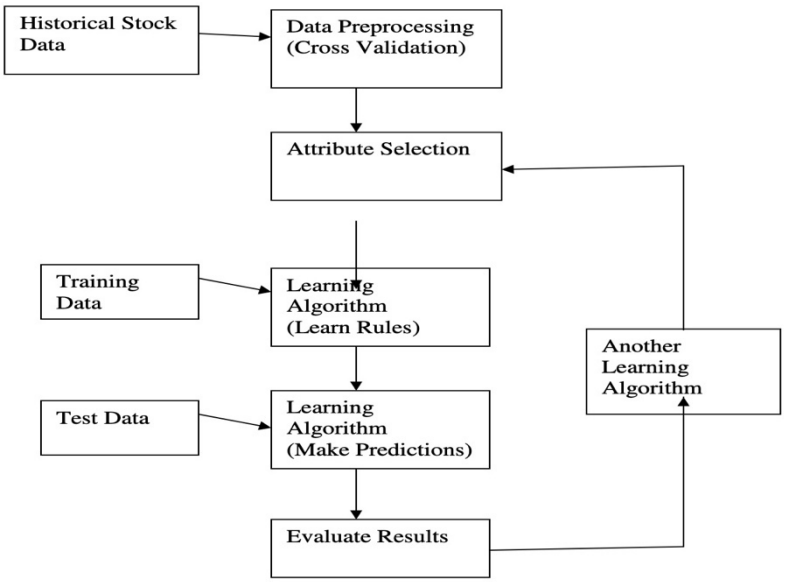
The main aim of this research is to propose an algorithm-based approach for stock trading or prediction for sizing of their trading positions.

The research objectives are formulated based on the aim of this study which are as follows:

- Analyze the data-set available for learning process
- To identify the learning methodology for prediction using LTSM
- Predict the stock price using the LTSM based RNN model
- Evaluate the performance of the model in terms of accuracy in terms of square off the trading position with maximum profit (sell at high, but at low)

5. Research Methodology

The Learning Environment:
The Weka and YALE Data Mining Environments were used for carrying out the experiments. The general setup used is as follows:

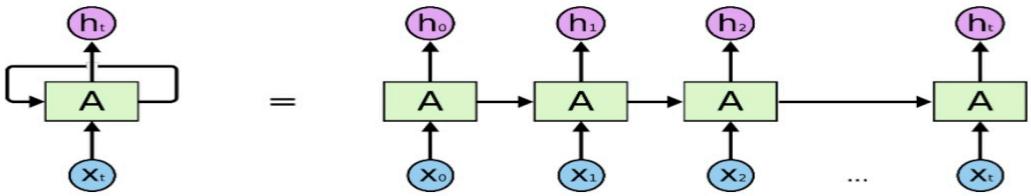


Neural Network: Machine learning has many applications, one of which is to forecast time series. One of the most interesting (or perhaps most profitable) time series to predict are the stock prices.

Need for Recurrent Neural Network (RNN) : Recurrent neural networks allow data to get persisted. Traditional neural networks cannot do this, and it seems like a major shortcoming, when it comes to forecasting a time series wherein in previous data values are put into considerations. A basic feed forward networks just “remember” things too, but they remember things they learnt during training, so is of no use in prediction of share value of stock.

For example, imagine we want to predict a stock price of a share, given a news good thing as happened few days back, say, the company has acquired a major online retail store. It's unclear how a traditional neural network could use its reasoning about previous events happened to inform later ones. Recurrent neural networks address this issue.

They are networks with loops in them, allowing information to persist within them.

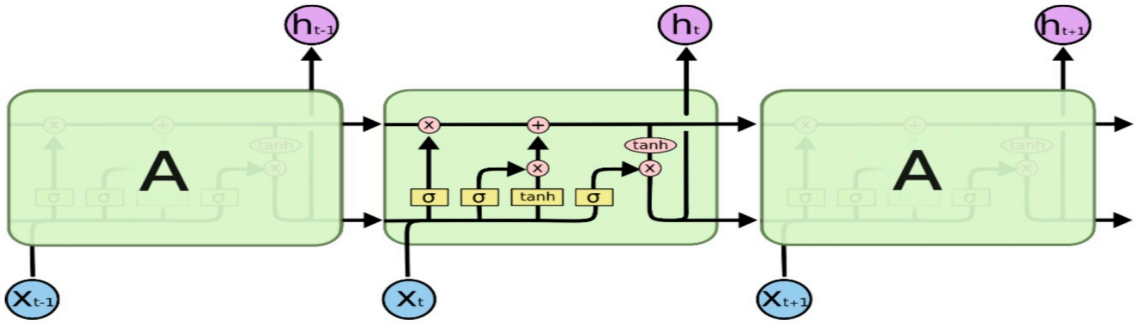


The Long Short Term Memory (LSTM) Neural Network

Long Short Term Memory networks – usually just called "LSTMs" – are a special kind of RNN, capable of learning long-term dependencies.

They were first introduced by Hochreiter & Schmidhuber in the year 1997, and they were refined and popularized by many people. They work tremendously well on a large variety of problems and are now widely used in several areas of machine learning.

LSTMs, like RNNs, also have this chain like structure, but the repeating module has a different structure. Instead of having a single neural network layer, there are four, interacting in a very special way.



LSTMs are widely used for sequence prediction problems and have proven to be extremely effective. The reason they work so well is because LSTM can store past information that is important and forget the information that is not.

LSTMs would learn the historical stock price of a share in a desired way using the following gates:

- *The input gate*: The input gate will add relevant share price information to the internal state.
- *The forget gate*: The forget gate will removes the extra/unnecessary information about stock trend that is no longer required by the model.
- *The output gate*: At Output Gate will selects the information to be shown as predicted values.

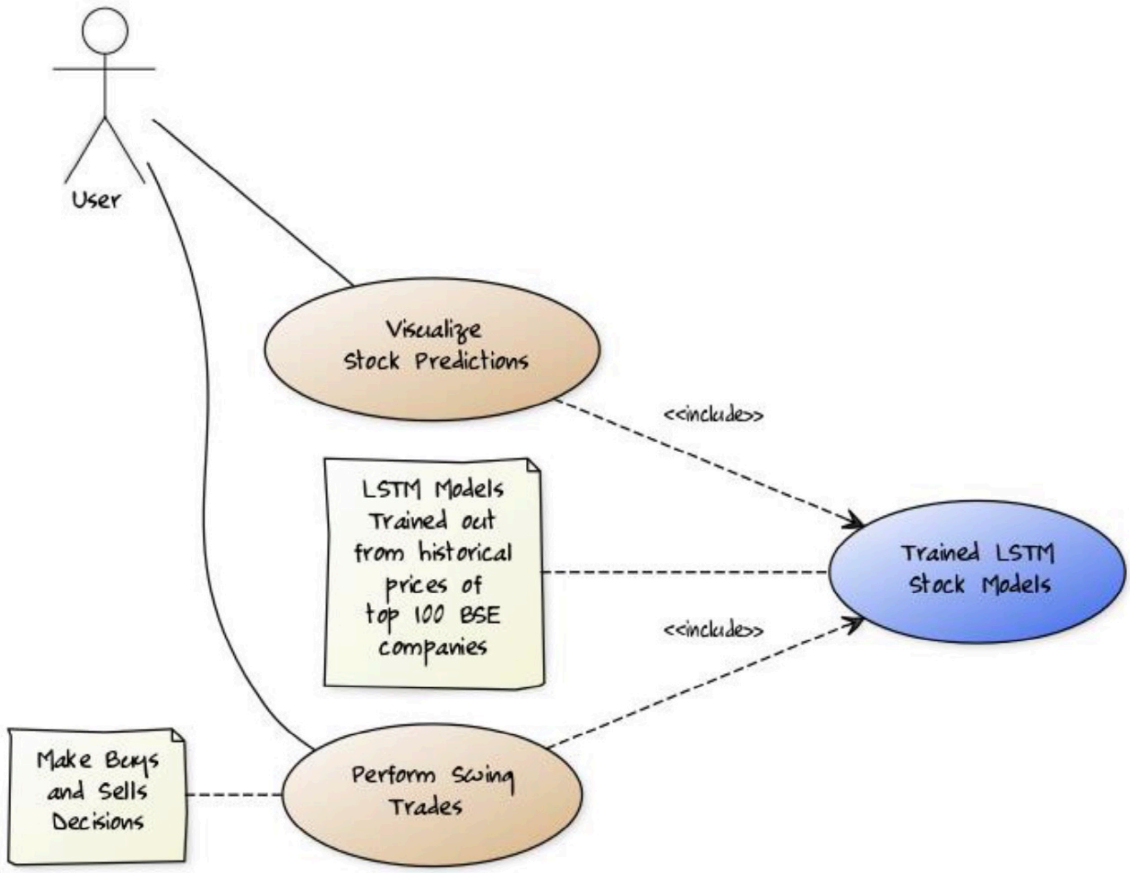
Training and Testing Dataset

A data set used to fit the model for purpose of training the neural network is the training data set. The entire data set will be the historical stock prices of a share.

The proportion of this data set would be used to form a testing dataset used for testing the trained model. About 70 - 80% of the entire dataset would be used for training and rest for testing.

The End to End Use Case Diagram

A typical work flow for this project is represented with following use case diagram. The user now has ability to visualize stock predictions using pre-trained neural network models. The visualization for predicted prices could help a buyer to perform Buy or Sell on a stock under consideration.



Dataset Preparation

The dataset that will be used to train the neural network would be the historical stock values of a company's share. The historical values would be ranging from 5 years to 10 years of data. The data would be available freely from website like:

• https://in.investing.com/

• https://www.quandl.com/

Date	Open	High	Low	Close	WAP	No. of Shares	No. of Trades	Total Turnover	Deliverable Quantity	% Deli. Qty to Traded Qty	Spread H-L	Spread C-O
9/11/2019	821.65	827.95	814.7	820.1	819.4	265029	3642	217165538	182625	68.91	13.25	-1.55
9/9/2019	838.05	840	827.3	829.2	831.14	118172	2988	98217182	39672	33.57	12.7	-8.85
9/6/2019	838	847.4	835.15	840.15	841.19	145166	3732	122112116	53722	37.01	12.25	2.15
9/5/2019	827.1	837	826.4	834.2	832.62	252694	8205	210397897	114057	45.14	10.6	7.1
9/4/2019	813.7	822.7	810.25	821.05	818.21	138123	3340	113013583	50373	36.47	12.45	7.35
9/3/2019	815	822.3	812.05	814.3	817.62	132158	3368	108055543	44348	33.56	10.25	-0.7
8/30/2019	808	817.5	802	814.6	809.1	156965	4282	126999897	66135	42.13	15.5	6.6
8/29/2019	798.9	809.65	796	806.85	804.54	177460	4038	142773169	57733	32.53	13.65	7.95
8/28/2019	785.1	804.9	785.1	802.1	796.47	164188	4669	130770827	60709	36.98	19.8	17
8/27/2019	791	795	781.2	785	786.03	248061	6542	194983829	101999	41.12	13.8	-6
8/26/2019	800	806.75	787.5	802.9	799.58	174352	4662	139408209	48812	28	19.25	2.9
8/23/2019	798	809.95	796	801.9	801.94	355792	8624	285325149	132721	37.3	13.95	3.9
8/22/2019	800.8	800.8	792.5	795.9	797.22	199130	9671	158750258	76127	38.23	8.3	-4.9
8/21/2019	792.8	803	792.75	799.55	799.36	339416	13955	271316701	173728	51.18	10.25	6.75
8/20/2019	781.05	797.9	781.05	792.9	793.16	513651	16144	407406376	249585	48.59	16.85	11.85
8/19/2019	777	783.3	773.8	777.8	779.71	262801	12926	204908043	154392	58.75	9.5	0.8
8/16/2019	779	779.8	762.25	774.55	772.37	226017	8379	174567971	119816	53.01	17.55	-4.45
8/14/2019	774	778.5	768.45	774.9	773.45	341438	13388	264084683	242760	71.1	10.05	0.9
8/13/2019	788.35	788.35	760.5	764.1	772.52	600910	14270	464215245	349534	58.17	27.85	-24.25
8/9/2019	791.5	796.4	785.5	790.15	790.21	483333	11043	381932759	311905	64.53	10.9	-1.35

The dataset in figure 5 has lot many columns in it, however for the training purpose, only ’Date’ and ‘Close’ Price columns will be taken into considerations.

The dataset will be split into two parts – *training data* (80%) *testing data (20%)*.

Since we will pass on this data to the LSTM neural network, we would have to preprocess the data set in such a way such that LSTM can tell for given say past N days stock price the next day’s value was XX.

6. Expected Outcomes

Expected outcome for a given stock is to find the target selling/buying price to maximize the profit. Time series forecasting is a very intriguing field to work with. Particularly when it comes to stock predictions eagerness to learn and put more thoughts which eventually could be used for making better financial investments.

We evaluate our model using a few selected pharma stocks. Several assumptions were made while constructing this portfolio strategy. First, the stock was assumed to be bought or sold within thirty minutes of the release of a news. Second, the returns were calculated based on the buy or sell price minus the closing price for that day. Furthermore, if the model suggests a buy decision based on the sentiment score, then it is considered a correct prediction only if the stock gains more than 0.5%. Hence a threshold of 0.5% was implemented for ’buy’ and ’sell’ decisions and 1% for ’hold’ decisions. For example, if the model suggests to ’hold’ a stock then it is essentially saying that the stock will not move either way for more than 1%, if it does, then the prediction is incorrect.

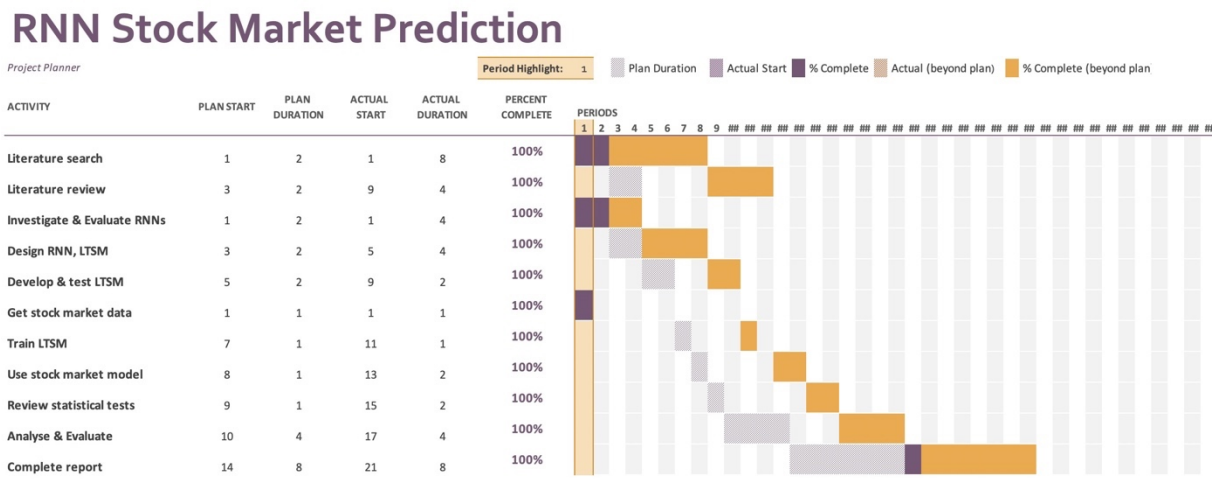
Predications made here is riskier for performing direct trading in share market based on the values predicted, however this would be surely helping to do some swing trading with least risk.

The project work explores the LSTM Neural Network gem, which itself is a very vast topic for research work.

7. Requirements / resources

1. PyTorch
2. GPU
3. Python
4. AWS/Cloud Infrastructure to store and process the data
5. Deep Learning

8. Research Plan



References

[1] Hall JW. Adaptive selection of U.S. stocks with neural nets. In: GJ Deboeck (Ed.), Trading on the edge: neural, genetic,and fuzzy systems for chaotic financial markets. New York: Wiley, 1994.

[2] Yaser SAM, Atiya AF. Introduction to financial forecasting. Applied Intelligence 1996; 6: 205–13. [3] Haykin, S., 1999, Neural Networks: A Comprehensive Foundation, 2nd Ed. (Englewood Cliffs, NJ: Prentice-Hall).

[3] Haykin, S., 1999, Neural Networks: A Comprehensive Foundation, 2nd Ed. (Englewood Cliffs, NJ: Prentice-Hall)

[4] Mehrara, M., Moeini, A., Ahrari, M., Ghafari, A., "Using Technical Analysis with Neural Network for Forecasting Stock Price Index in Tehran Stock Exchange" Middle Eastern Finance and Economics, Vol. 6, No. 6, pp. 50-61, 2010.

[5] Agrawal, S., Jindal, M., Pillai, G. N., "Momentum Analysis based Stock Market Prediction using Adaptive Neuro-Fuzzy Inference System (ANFIS)", Proceedings of the International MultiConference of Engineers and Computer Scientists, Hong Kong, Vol. 1, March 17 -19, 2010.

[6] Majumder, M., Hussian, A., "Forecasting of Indian Stock Market Index Using Artificial Neural Network", .

[7] B. Pang and L. Lee, "Opinion mining and sentiment analysis," Foundations and Trends® in Information Retrieval, vol. 2, no. 1–2, pp. 1-135, 2008.

[8] H. Isah, "Social Data Mining for Crime Intelligence: Contributions to Social Data Quality Assessment and Prediction Methods," University of Bradford, 2017.

[9] A. Mittal and A. Goel, "Stock prediction using twitter sentiment analysis," Standford University, CS229 (2011 http://cs229. stanford. edu/proj2011/GoelMittalStockMarketPredictionUsingTwitterSentimentAnalysis. pdf), vol. 15, 2012.

[10] H. Li, A. Mukherjee, B. Liu, R. Kornfield, and S. Emery, "Detecting campaign promoters on twitter using markov random fields," in Data Mining (ICDM), 2014 IEEE International Conference on, 2014, pp. 290-299: IEEE.

[11] T. H. Nguyen, K. Shirai, and J. Velcin, "Sentiment analysis on social media for stock movement prediction," Expert Systems with Applications, vol. 42, no. 24, pp. 9603-9611, 2015.

[12] Zabir Haider Khan, Tasnim Sharmin Alin, Md. Akter Hussain,‖ Price Prediction of Share Market using ArtificiaNeural Network (ANN)‖, International Journal of Computer Applications (0975 – 8887) Volume 22– No.2, May 2011

[13] Ramin Rajabioun and Ashkan Rahimi-Kian,‖ A Genetic Programming Based Stock Price Predictor together with Mean-Variance Based Sell/Buy Actions", Proceedings of the World Congress on Engineering 2008 Vol II WCE 2008, July 2 - 4, 2008, London, U.K.

[14] Tsai, C.-F. and Wang, S.-P.,‖ Stock Price Forecasting by Hybrid Machine Learning Techniques‖ Proceedings of the International MultiConference of Engineers and Computer Scientists 2009 Vol I IMECS 2009, March 18 - 20, 2009, Hong Kong.

[15] Nguyen Lu Dang Khoa1, Kazutoshi Sakakibara2 and Ikuko Nishikawa2,‖ Stock Price Forecasting using Back Propagation Neural Networks with Time and Profit Based Adjusted Weight Factors‖, SICE-ICASE International Joint Conference 2006 Oct. 18-2 1, 2006 in Bexco, Busan, Korea.

[16] Y.R.RAMESH KUMAR, Prof.A.Govardhan,‖Stock market predictions integrating user perception for extracting better prediction – a Frame work‖, International Journal of Engineering Science and Technology Vol. 2(7), 2010, 3305-3310.

[17] Wei Huang, Kin Keung Lai, Yoshiteru Nakamori, Shouyang Wang, Lean Yu ―NEURAL NETWORKS IN FINANCE AND ECONOMICS FORECASTING" International Journal of Information Technology & Decision Making Vol. 6, No. 1 (2007) 113–140.

[18] N. Chen, ―Financial investment opportunities and the macroeconomy‖, Journal of Finance 46 (1991) 529–554.

[19] E. Fama and K. French,‖Common risk factors in the returns on stocks and bonds‖, Journal of Financial Economics 33 (1993) 3–56.

[20] K. Kohara, T. Ishikawa, Y. Fukuhara and Y. Nakamura, ―Stock price prediction using prior knowledge and neural networks‖, Intelligent Systems in Accounting, Finance and Management 6 (1997) 11–22.

[21] Adebiyi Ayodele A., Ayo Charles K., Adebiyi Marion O., and Otokiti Sunday O.,‖ Stock Price Prediction using Neural Network with Hybridized Market Indicators", Journal of Emerging Trends in Computing and Information Sciences VOL. 3, NO. 1, January 2012 ISSN 2079-8407.

[22] Robert P. Schumaker, Hsinchun Chen ―Textual Analysis of Stock Market Prediction Using Financial News Articles‖.

[23] Leonardo C. Martinez, Diego N. da Hora, Joao R. de M. Palotti, Wagner Meira Jr. and Gisele L. Pappa, ―From an Artificial Neural Network to a Stock Market Day-Trading System: A Case Study on the BM&F BOVESPA‖, Proceedings of International Joint Conference on Neural Networks, Atlanta, Georgia, USA, June 14-19, 2009.

[24] Joish Bosco. Stock Market Prediction and Efficiency Analysis Using Recurrent Neural Network: Grin Verlag, 2018.

[25] Simeon Kostadinov. Recurrent Neural Networks with Python Quick Start Guide: Sequential learning and language modeling with TensorFlow: Packt Publishing Limited, 2018.

[26] Zhang Yi : Convergence Analysis of Recurrent Neural Networks (Network Theory and Applications): Springer, 2014

[27] Recurrent neural network https://en.wikipedia.org/wiki/Recurrent_neural_network

[28] Understanding-LSTMs http://colah.github.io/posts/2015-08-Understanding-LSTMs/