

TRIBHUWAN UNIVERSITY

INSTITUTE OF ENGINEERING

PULCHOWK CAMPUS

DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING



MAJOR PROJECT MID TERM REPORT

On

STOCK MARKET ANALYSIS AND PREDICTION

Submitted To :

Department of Electronics and Computer Engineering
Institute of Engineering (IOE)
Pulchowk Campus

Submitted By:

Abin Shakya(070 BCT 503)
Anuj Pokhrel(070 BCT 507)
Ashuta Bhattarai(070 BCT 510)
Pinky Sitikhu(070 BCT 524)

July 17, 2017

ACKNOWLEDGMENT

We would like to express our sincere gratitude to the Department of Electronics and Computer Engineering, Pulchowk Campus for providing us the opportunity to do this project. We are extremely thankful to Dr. Nanda Bikram Adhikari for providing us with invaluable guidance and support.

We would like to express our gratitude to our Supervisor, Prof. Dr. Subarna Shakya for guiding us throughout the project and helping us correct the errors.

Lastly, a very special thanks to our colleagues whose support and suggestions have been an invaluable contribution to our project.

With warm regards,

Abin Shakya (070BCT503)

Anuj Pokhrel (070BCT507)

Ashuta Bhattarai (070BCT510)

Pinky Sitikhu (070BCT524)

TABLE OF CONTENTS

Acknowledgment	i
Table of Contents	iii
List of Figures	iv
1 Introduction	1
1.1 Background	1
1.2 Objectives	2
1.3 Problem Statement	2
1.4 Scope of project	3
2 LITERATURE REVIEW	4
2.1 Theory Details	4
2.2 Related Work	6
3 METHODOLOGY	7
3.1 Data Collection	7
3.2 Data Preprocessing	7
3.3 Data analysis and visualization	7
3.4 Feature Selection	8
3.5 Prediction	8
3.5.1 Prediction using Technical Analysis	9
3.5.2 K- nearest neighbor	12
3.6 Tools and Technique	13
4 Software Development Model	18
5 REQUIREMENTS	19
5.1 Functional Requirements	19
5.2 Non-Functional Requirements	19
6 System Model	21
6.1 Backlog	21
6.2 Dataflow Diagram	22
6.3 Sequence Diagram	23
6.4 Use Case Modeling	24
6.5 Activity Diagram	25

7 CONCLUSION**26****References****27**

LIST OF FIGURES

3.1	ANN model	9
3.2	Table showing activation function	10
3.3	Backpropagation Algorithm	11
3.4	KNN model	13
6.1	Dfd level 0	22
6.2	Dfd level 1	22
6.3	Sequence diagram for stock prediction	23
6.4	Sequence diagram for stock analysis	23
6.5	Top level usecase diagram for stock market analysis and prediction	24
6.6	Activity diagram for stock market analysis and prediction	25

1. INTRODUCTION

1.1. Background

Stock is a share in the ownership of a company. Stock represents a claim on the company's assets and earnings. Stocks are issued by the company itself so that it can raise money by selling a small fraction of it. Issuing stock is advantageous for the company because it does not require the company to pay back the money to some party/bank. The buying and selling of the stocks is done in a stock market.

Stock market and its trends are extremely volatile in nature in the finance field. It attracts researchers to capture the volatility and predicting its next moves. Investors and market analysts study the market behavior and plan their buy or sell strategies accordingly. The overall operation of stock market is based on the concept of demand and supply. If demand for a company's stock is higher in the market then the company's share prices move in upward direction. Likewise if the demand for the company's stock is very low compared to the supply then it is obvious that the company shares will be dealt in lower price. This is the basic principle behind the operation of stock market.

A stock exchange is an exchange where stock-brokers and traders can buy and/or sell stocks (or shares), bonds and other securities. Companies may want to get their stock listed on a stock exchange. Other stocks may be traded "over the counter", that is, through a dealer. A large company will usually have its stock listed on many exchanges across the world. Some exchanges are physical locations where transactions are carried out on a trading floor, by a method known as open outcry. This method is used in some stock exchanges and commodity exchanges, and involves traders entering oral bids and offers simultaneously. An example of such an exchange is the New York Stock Exchange(abbreviated as NYSE). The other type of stock exchange is a virtual kind, composed of a network of computers where trades are made electronically by traders. An example of such an exchange is the NASDAQ.

The Nepal Stock Exchange Limited (abbreviated as NEPSE) is the only Stock Exchange of Nepal. It is located in Singha Durbar Plaza, Kathmandu Nepal. The basic objective of NEPSE is to impart free marketability and liquidity to the government and corporate securities by facilitating transactions in its trading floor through member, market intermediaries, such as broker, market makers etc.

The stock market can be viewed as a particular data mining and artificial intelligence problem. The movement in the stock exchange depends on capital gains and losses and most people consider the stock market erratic and unpredictable. However, patterns that allow the prediction of some movements can be found. Stock market analysis deals with the study of these patterns. It uses different techniques and strategies, mostly automatic that trigger buying and selling orders depending on different decision making algorithms. It can be considered as an intelligent treatment of past and present financial data in order to predict the stock market future behavior. Therefore it can be viewed as an artificial intelligence problem in the data mining field.

But, stock market prediction comes with a challenging question of whether the stock price is predictable or not? So, the random walk hypothesis states that the price of stock is collocated by a random walk and hence the stock market is unpredictable. The debate about whether the stock market can be predicted or not has lasted for many years, but there has not been a consensus yet. However, many researchers have built their own stock price predicting systems, to some extent, for proving the stock market's predictability.

1.2. Objectives

The main objectives of this projects are as follows :

1. To predict the stock market trend based on technical, fundamental and news-sentimental analysis
2. To visualize the prediction results and daily trading prices in the form of interactive charts
3. To compare the results and effectiveness of each algorithm with another

1.3. Problem Statement

Theoretically, the stock market is said to be very difficult to predict, due to its dynamic and non-linear model. However, the investors and stock analysts have been trying to somehow predict the stock prices of a company, to increase the profit in buying and selling stocks. Appreciable efforts have also been made from academic researchers and enthusiasts in this field. However, identifying the pattern of such an uncertain system through simple calculations and mathematics results in poor accuracy with questionable reliability. Complex, dedicated systems and models are required which can take into consideration, the numerous factors that can affect the stock price of a company. For an instance, the intrinsic valuation of a company

and its performance in the market till now are equally important factors in determining its future price. However, it is very difficult to know for certain which factor affects the most at the given time, and by how much. Therefore, the market should be analyzed under various influencing factors, the prime of which are: Technical factors, Fundamental factors and News-sentimental factors. In technical analysis, the prediction model is built considering a company's past performance in the stock market, which includes studying the past rise and fall trends, average traded volumes, bullish trend behaviors and so on. In fundamental analysis, the worth of the company, its current profits, capital gains and the future profits play a vital role in understanding its stock price behaviors. In news-sentiment analysis, the immediate effects of political, economic and stock related news in a company's stock prices is studied and applied.

Therefore, through circumstantial application of above mentioned analysis, this project presents a general and complete solution for stock prediction, which can be employed in the real world for gaining profit in the stock market.

1.4. Scope of project

The project aims to predict the stock trend movements of trading companies based on large volume of historical data collected from various sources. The historical data constitutes of a company's fundamental valuations, past trading prices and volumes, and past news features. The basic driving factors for choosing a prediction model is its effectiveness, applicability and accuracy of results. By using multiple analysis and prediction models, the project aims to compare the usability of each such models. On completion of this project, we aim to establish a highly reliable stock prediction system, which can be used by investors to decide when to buy or sell the stocks of a company in order to gain maximum profit. It is hoped that the project will be beneficial for the stakeholders including, researchers, business analysts, stock market enthusiasts and policy makers. The project is also focused on improving the trading experience of new investors who may or may not know much about the market behaviors.

2. LITERATURE REVIEW

2.1. Theory Details

When predicting the future prices of Stock Market securities, there are several theories available. The first is Efficient Market Hypothesis (EMH). In EMH, it is assumed that the price of a security reflects all of the information available and that everyone has some degree of access to the information. Fama's theory further breaks EMH into three forms: Weak, Semi-strong, and Strong. In Weak EMH, only historical information is embedded in the current price. The Semi-Strong form goes a step further by incorporating all historical and currently public information in the price. The Strong form includes historical, public, and private information, such as insider information, in the share price. From the tenets of EMH, it is believed that the market reacts instantaneously to any given news and that it is impossible to consistently outperform the market.

A different perspective on prediction comes from Random Walk Theory. In this theory, Stock Market prediction is believed to be impossible where prices are determined randomly and outperforming the market is infeasible. Random Walk Theory has similar theoretical underpinnings to Semi-Strong EMH where all public information is assumed to be available to everyone. However, Random Walk Theory declares that even with such information, future prediction is ineffective.

It is from these theories that two distinct trading philosophies emerged; the fundamentalists and the technicians. In a fundamentalist trading philosophy, the price of a security can be determined through the nuts and bolts of financial numbers. These numbers are derived from the overall economy, the particular industry's sector, or most typically, from the company itself. Figures such as inflation, joblessness, return on equity (ROE), debt levels, and individual Price to Earnings (PE) ratios can all play a part in determining the price of a stock.

In contrast, technical analysis depends on historical and time-series data. These strategists believe that market timing is critical and opportunities can be found through the careful averaging of historical price and volume movements and comparing them against current prices. Technicians also believe that there are certain high/low psychological price barriers such as support and resistance levels where opportunities may exist. They further reason that price movements are not totally random, however, technical analysis is considered to be more of an art form rather than a science and is subject to interpretation.

Both fundamentalists and technicians have developed certain techniques to predict prices from financial news articles. In one model that tested trading philosophies, LeBaron et. al.

posited that when predicting the future prices of Stock Market securities, there are several theories available. The first is Efficient Market Hypothesis (EMH). In EMH, it is assumed that the price of a security reflects all of the information available and that everyone has some degree of access to the information.

In similar research on real stock data and financial news articles, Gidofalvi gathered over 5,000 financial news articles concerning 12 stocks, and identified this brief duration of time to be a period of twenty minutes before and twenty minutes after a financial news article was released. Within this period of time, Gidofalvi demonstrated that there exists a weak ability to predict the direction of a security before the market corrects itself to equilibrium. One reason for the weak ability to forecast is because financial news articles are typically reprinted throughout the various news wire services. Gidofalvi posits that a stronger predictive ability may exist in isolating the first release of an article. Using this twenty-minute window of opportunity and an automated textual news parsing system, the possibility exists to capitalize on stock price movements before human traders can act.

Also Ralph Nelson Elliott developed the Elliott wave theory in the late 1920s by discovering that stock markets, thought to behave in a somewhat chaotic manner, in fact traded in repetitive cycles. Elliott discovered that these market cycles resulted from investors reactions to outside influences, or predominant of psychology of the masses at the time. He found that the upward and downward swings of the mass psychology always showed up in the same repetitive patterns, which were then divided further into patterns he termed **waves**.

For the stock market prediction, Artificial Neural Network(ANN) has been considered the most efficient method. Inspired by neurosciences, ANNs have shown great potential in terms of recognizing patterns in nonlinear systems. Existing research suggests that ANN is an eminent model to predicting stock markets due to its dynamical characteristics. Even so, a common criticism of neural networks is that they require a large diversity of training for real-world operation. Moving average analysis and single exponential smoothing methods are frequently used in order to make stock analysis. The Nepal stock exchange (NEPSE) uses exponential smoothing in its website for this purpose. Moving averages work quite well in strong trending conditions, but often poorly in choppy or ranging conditions.

Under the assumption that the stock market could be predicted, there are some major categories of prediction methods: fundamental analysis, technical analysis and news analysis.

I. Fundamental Analysis

It mainly depends on statistical data of a company. It includes reports, financial status of the company, the balance sheets, dividends and policies of the companies whose stock are to be

observed. It also includes analysis of market data, strength and investment of company, the competition, import/export volume, production indices, price statistics of the company.

II. Technical Analysis

In stock analysis there are two approaches, first approach includes analysis of graphs where analysts try to find out certain patterns that are followed by stock but this approach is very difficult and complex to be used with ANN. In second approach analysts make use of quantitative parameters like trend indicators, daily ups and downs, highest and lowest values of a day, volume of stock, indices, pull/call ratios, etc. It also includes some averages which is nothing more than mean of prices for particular window size like Simple Moving Average(MA) and Exponential Moving Average(EMA). Here prices of recent days have more weight in average. Analysts try to find out some mathematical formula which can map this input in the desired output.

III. News Analysis

Some of the researchers showed that there is a strong relationship between news article about a company and its stock prices fluctuations. This analysis includes the processing of analyzing the news of the company using Natural Language Processing techniques.

Both long-term and short-term stock price can be calculated considering the above analysis. Technical and fundamental analysis could be adopted for long term prediction whereas news sentiment analysis could play a handy role for short-term predictions.

2.2. Related Work

Many algorithms of data mining have been proposed to predict stock price. Neural Network, Genetic Algorithm, Decision Tree and Fuzzy systems are widely used. In addition, pattern discovery is beneficial for stock market prediction and public sentiment is also related to predicting stock price.

There are a lot of software and web applications working with the similar concept. Nepal Sharemarket is a website that makes individual, comparative as well as in depth analysis on stock market companies and also forecasts their price on a chosen time basis.

Another website, Stock-forecasting.com also makes stock prediction using neural networks and boasts of highest accuracy among all the stock-prediction applications. It is an American company and gives minute predictions of various international companies.

Other software like InteliCharts and Addaptron also make stock predictions based on neural networks.

3. METHODOLOGY

3.1. Data Collection

Both analysis and prediction of stock market needed an extensive amount of data for better visualization and training. News data regarding stock market were required for news analysis, which were collected from sharesansar website via web crawling. Trading data of listed companies, for technical analysis were collected from merolagani website. Similarly, Sector wise data required for Nepal Stock Exchange Ltd.(NEPSE) index prediction were collected from sharesansar. Unfortunately, the data required for Fundamental Analysis were not available for crawling. As a result, required data were extracted manually from the web.

Aside from the static past data, a mechanism to update the recent changes was also required to update the database constantly. For this, a manual update module was built, which at the end of the day, updates the changes made throughout the day in the database

3.2. Data Preprocessing

The trading data crawled from Merolagani and Sharesansar had numerous missing fields, which were filled up using interpolation technique to cover up the possible setbacks. Once the data is cleaned, it is stored in the database for future retrievals.

The training data on analysis showed high fluctuation, which needed some smoothing technique in order to feed it into our model for better results. Thus, Exponential Moving Average was used to reduce the data into suitable form.

For the news analysis, the news so collected are to be divided into feature set and label. The feature set were extracted using 'Bag of words representation'. The label were given to each vector as positive or negative based on whether the corresponding price increased or decreased. All the news were labeled accordingly to get a complete training data set.

3.3. Data analysis and visualization

Data Analysis in financial market involves two basic approaches and they are: Technical-analysis and Fundamental analysis. Technical analysis, which involves detecting patterns in

security prices, goes on the assumption that the price of a stock - like the price of everything else - is a matter of supply and demand. Technical analysis generates and interprets charts of the price and volume histories of stocks to predict movement in stock prices according to perceived trends. Fundamental analysis, which examines the earning potential of the company issuing a stock, goes on the assumption that a share of ownership of a company has an intrinsic value that is a function of the underlying value of the company as a whole. Fundamental analysis reports which shares are undervalued by the investor community and which are overvalued then trust the market to make corrections.

Data visualization is a general term that describes any effort to help people understand the significance of data by placing it in a visual context. Data visualization was done with the help of charting library called AmCharts. Company wise data were shown in charts and features like comparison of stock data were also integrated.

3.4. Feature Selection

The data features that are used to train machine learning models have a huge influence on the performance that can be achieved. Irrelevant or less relevant selection of data features result in low performance during prediction. So, it is necessary to select the best possible features that best influence the result.

Benefits of performing feature selection before modeling the data are:

- Reduces Over-fitting: Less redundant data means less opportunity to make decisions based on noise.
- Improves Accuracy: Less misleading data means modeling accuracy improves.
- Reduces Training Time: Less data means that algorithms train faster.

Statistical tests can be used to select those features that have the strongest relationship with the output variable. The scikit-learn library provides the SelectKBest class that we used with a suite of different statistical tests to select a specific number of features.

3.5. Prediction

The system intends to predict the stock market using the available historical data. The prediction model is generated by manipulating the historical data by casting them through various

artificial intelligence techniques. In general, stock market prediction can be done by analyzing the past stock trends with respect to fundamental, technical and news-sentiment analysis.

3.5.1. Prediction using Technical Analysis

[1]The field of technical analysis is based on three assumptions:

- a. The market discounts everything
- b. Price moves in trends
- c. History tends to repeat itself

Technical analysis studies the trend of supply and demand within the market to determine what direction or trend will continue in the future. In other words, it attempts to understand the emotions in the market by studying the market itself, as opposed to its components. Various technical indicators, such as Relative Strength Index (RSI) and Moving Average Convergence Divergence (MACD), are used to ease the process of technical analysis.

1. Technical analysis using Artificial Neural Network

Artificial Neural Networks (ANNs) are simply inspired from biological neural networks that make up the networks of living neuron cells in animals. Computer systems excel the human brain in performing complex mathematical operations by thousands of times but, lack the human ability of logical reasoning and pattern recognition. The use of ANN allows computers to process data the same way the human brain processes a stimulus, providing them the ability to recognize patterns even in non-linear data such as that of stock market. For this process, the ANN is trained with historical data using supervised learning method. Once the training is completed, we move on to the testing phase, where the reliability, accuracy and efficiency of the training algorithm is tested. Once the ANN has passed the test, it can then be used for prediction.

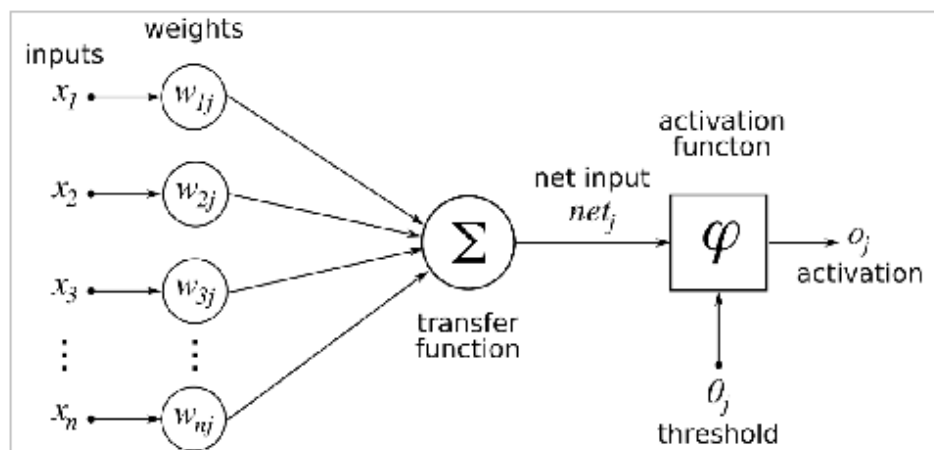


Figure 3.1: ANN model

ANN is considered one of the most effective methods in predicting the stock market. Even within ANN, the Multi-Layer Perceptron (MLP) model is widely accepted for effective pattern recognition. [2] An MLP model is a class of feed-forward Artificial Neural Network that consists of at least three layers of nodes namely: Input layer, Hidden layer(s) and Output layer. MLP utilizes supervised learning technique and Backpropagation algorithm for training.

The artificial neurons shown in 3.1 have n number of inputs:

$$x_1, x_2, \dots, x_n$$

each of which is associated with a weight onto the connection line, denoted as

$$w_{1j}, w_{2j}, \dots, w_{nj}$$

respectively. The weights can be referred as synaptic weights as in a biological neural network. " θ " represents the threshold and " α " represents the activation function given by:

$$\alpha = \sum_{k=1}^n [w_{kj} * x_k + \theta]$$

The output of the neuron, O_j , is a function of its activation given by: $O_j = f(\alpha)$

Several types of activation functions can be used, which are summarized in Figure 3.2:


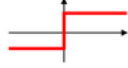


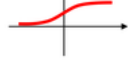

Activation function	Equation	Example	1D Graph
Unit step (Heaviside)	$\phi(z) = \begin{cases} 0, & z < 0, \\ 0.5, & z = 0, \\ 1, & z > 0, \end{cases}$	Perceptron variant	
Sign (Signum)	$\phi(z) = \begin{cases} -1, & z < 0, \\ 0, & z = 0, \\ 1, & z > 0, \end{cases}$	Perceptron variant	
Linear	$\phi(z) = z$	Adaline, linear regression	
Piece-wise linear	$\phi(z) = \begin{cases} 1, & z \geq \frac{1}{2}, \\ z + \frac{1}{2}, & -\frac{1}{2} < z < \frac{1}{2}, \\ 0, & z \leq -\frac{1}{2}, \end{cases}$	Support vector machine	
Logistic (sigmoid)	$\phi(z) = \frac{1}{1 + e^{-z}}$	Logistic regression, Multi-layer NN	
Hyperbolic tangent	$\phi(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$	Multi-layer NN	

Figure 3.2: Table showing activation function

2. Backpropagation Algorithm

The chief objective of the Backpropagation Algorithm is to reduce the error function. [3]

This algorithm falls into the general category of gradient descent algorithms, which intend to find the minima/maxima of a function by iteratively moving in the direction of the negative of the slope of the function to be minimized/maximized. This algorithm proceeds across the network, providing activation to each node until the output node is reached. Then, the weights are updated backwards, from the output layer towards the input layer until one epoch has been completed. The weights are updated according to the respective errors computed for each layer.

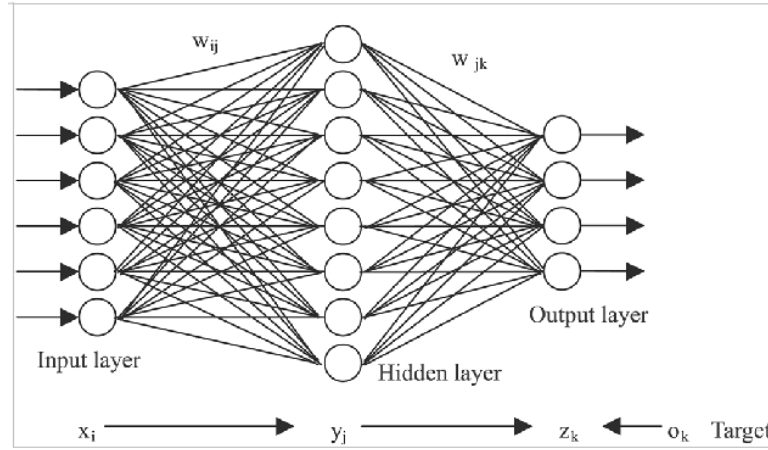


Figure 3.3: Backpropagation Algorithm

In the figure, For k output units, if t_k signifies the target value, o_k signifies the actual output, α signifies the learning rate and z_{ink} signifies the activation function for the k^{th} output node, error (δ) is given by: $\delta_k = (t_k - o_k) * f'(z_{ink})$

Now, the weight correction for each output unit is given by: $\Delta w_{jk} = \alpha * \delta_k * z_j$

Similarly, by propagating the delta term further back in the network, the input error in the hidden network is calculated $\delta_{inj} = \sum_{k=1}^m \delta_k * w_{jk}$ Where, m = number of neurons in the hidden layer

Now, error in the j th hidden unit is calculated by: $\delta_j = \delta_{inj} * f'(x_{inj})$ Where x_{inj} signifies the activation function for the j^{th} hidden layer node.

Now, the weight correction is given by:

$$\Delta w_{ij} = \alpha * \delta_j * X_i, \text{ for hidden layer nodes}$$

$$\Delta w_{jk} = \alpha * \delta_k * Y_j, \text{ for output layer nodes}$$

Then, the weights for each neuron is updated with the new ones. Once an epoch has been completed, the average error for each training data is calculated. Usually the RMS error between the target value and actual outputs is computed for convergence. If the RMS error falls

within the acceptable range, the training is completed, else, the whole process is repeated.

Hence, the above algorithm can be used to train an Artificial Neural Network. The network, in general, can have an arbitrary number of hidden layers and an arbitrary number of hidden neurons in each layer. For practical reasons, ANNs implementing the backpropagation algorithm do not have too many layers, since the time for training the networks grows exponentially. The number of input layer neurons is decided by the number of input features in each pattern, and the number of output layer neurons is decided by the number of output features in the target values.

There are a few disadvantages associated with backpropagation learning as well:

- The convergence obtained from backpropagation learning is very slow.
- The convergence in backpropagation learning is not guaranteed.
- The result may generally converge to any local minimum on the error surface, since stochastic gradient descent exists on a surface which is not flat.
- Backpropagation learning requires input scaling or normalization.
- Backpropagation requires the activation function used by the neurons to be differentiable.

3.5.2. K- nearest neighbor

The k-nearest neighbors algorithm is a non-parametric method used for classification and regression. We are using k-NN for classification of news in news analysis in our project. The input consists of the k closest training examples in the feature space. The output in k-NN classification is a class membership. An object is classified by a majority vote of its neighbors, with the object being assigned to the class most common among its k nearest neighbors. The K-Nearest Neighbor is a simple lazy learner algorithm that stores all available data points and classifies new instances based on a similarity measure.

During the training phase the algorithm simply stores the data points including their class labels and all computation is deferred until the classification process. It is based on a principle that instances that are in close proximity to another have similar properties. Thus, to classify new unclassified instances, one simply has to look at their k-nearest neighbors, to figure out the classification label. The class membership can be defined by a majority vote of the k closest neighbors or the neighbors can be ranked and weighted according to their distance to the new instance.

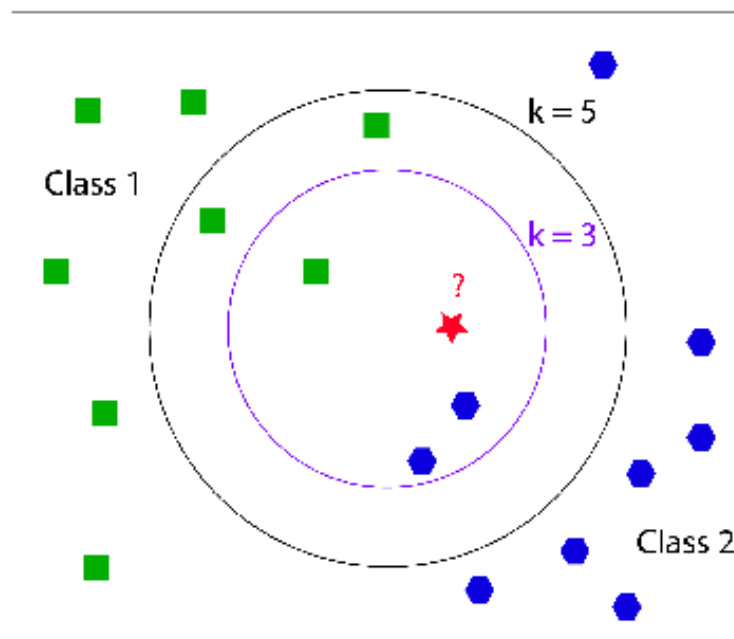


Figure 3.4: KNN model

3.6. Tools and Technique

The various tools and techniques used in this project are described below:

1. Python

The whole project is written in Python Programming Language. Various libraries of python are used in the project. Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. It was created by Guido van Rossum during 1985-1990. Like Perl, Python source code is also available under the GNU General Public License (GPL).

Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages. Python is processed at runtime by the interpreter. Python is Interactive. Users can actually sit at a Python prompt and interact with the interpreter directly to write programs. Python is Object-Oriented: Python supports Object-Oriented style or technique of programming that encapsulates code within objects. Python is a Beginner's Language: Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games. It supports functional and structured programming methods as well as OOP. It can be used as a scripting language or can be compiled to byte-code for building large applications. It provides very high-level dynamic data types and supports dynamic type checking. It supports automatic

garbage collection. It can be easily integrated with C, C++, COM, ActiveX, CORBA, and Java.

2. Pandas

Pandas is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series. Pandas is free software released under the three-clause BSD license. The name is derived from the term "panel data", an econometrics term for multidimensional structured data sets. Python has long been great for data munging and preparation, but less so for data analysis and modeling. Pandas helps fill this gap, enabling users to carry out the entire data analysis workflow in Python without having to switch to a more domain specific language like R. Pandas modules uses objects to allow for data analysis at a fairly high performance rate in comparison to typical Python procedures. With it, users can easily read and write from and to CSV files, or even databases. From there, users can manipulate the data by columns, create new columns, and even base the new columns on other column data.

3. Scikit learn

Scikit-learn (formerly scikits.learn) is a free software machine learning library for the Python programming language. It features various classification, regression and clustering algorithms including support vector machines, random forests, naive bayes, gradient boosting, k-means and DBSCAN, and is designed to inter operate with the Python numerical and scientific libraries NumPy and SciPy. Scikit-learn was initially developed by David Cournapeau as a Google summer of code project in 2007. Later Matthieu Brucher joined the project and started to use it as apart of his thesis work. In 2010 INRIA got involved and the first public release (v0.1 beta) was published in late January 2010. Scikit-learn provides a range of supervised and unsupervised learning algorithms via a consistent interface in Python. It is licensed under a permissive simplified BSD license and is distributed under many Linux distributions, encouraging academic and commercial use. Some popular groups of models provided by scikit-learn include:

- Clustering: for grouping unlabeled data such as KMeans.
- Cross Validation: for estimating the performance of supervised models on unseen data.
- Dimensionality Reduction: for reducing the number of attributes in data for summarization, visualization and feature selection such as Principal component analysis.
- Feature extraction: for defining attributes in image and text data.
- Parameter Tuning: for getting the most out of supervised models.

- Supervised Models: a vast array not limited to generalized linear models, discriminate analysis, naive bayes, lazy methods, neural networks, support vector machines and decision trees.

4.Numpy

Numpy is the core library for scientific computing in Python. NumPy is a Python package. It stands for 'Numerical Python'. It is a library consisting of multidimensional array objects and a collection of routines for processing of array.

Using NumPy, a developer can perform the operations like Mathematical and logical operations on arrays, Fourier transforms and routines for shape manipulation, Operations related to linear algebra, NumPy has in-built functions for linear algebra and random number generation.

5.NLTK

NLTK is a leading platform for building Python programs to work with human language data. It provides easy-to-use interfaces to over 50 corpora and lexical resources such as WordNet, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning. Tokenizers is used to divide strings into lists of substrings. For example, Sentence tokenizer can be used to find the list of sentences and Word tokenizer can be used to find the list of words in strings. NLTK is one of the natural language processing tool that was used in the project.

6. Django

Django was used in the project to design the web interface. Django is a free and open-source web framework, written in Python, which follows the modelcontroller (MVC) architectural pattern. It is maintained by the Django Software Foundation (DSF), an independent organization established as a 501(c)(3) non-profit. Django's primary goal is to ease the creation of complex, database-driven websites. Django emphasizes reusability and "pluggability" of components, rapid development, and the principle of don't repeat yourself.

Python is used throughout, even for settings files and data models. Django also provides an optional administrative create, read, update and delete interface that is generated dynamically through introspection and configured via admin models.

7.Git

Git was used as a version control system to collaborate among the team members. Git is a version control system that is used for software development and other version control tasks. As a distributed revision control system it is aimed at speed, data integrity, and support for distributed, non-linear workflows. Git was created by Linus Torvalds in 2005 for development of the Linux kernel, with other kernel developers contributing to its initial development. The Git feature that really makes it stand apart from nearly every other SCM out there is its branching model.

Git allows and encourages you to have multiple local branches that can be entirely independent of each other. The creation, merging, and deletion of those lines of development takes seconds.

8.Postgresql

PostgreSQL is a powerful, open source object-relational database system. It has more than 15 years of active development and a proven architecture that has earned it a strong reputation for reliability, data integrity, and correctness. PostgreSQL runs on all major operating systems, including Linux, UNIX (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64), and Windows. PostgreSQL (pronounced as "post-gress-Q-L") is an open source relational database management system (RDBMS) developed by a worldwide team of volunteers. PostgreSQL is not controlled by any corporation or other private entity and the source code is available free of charge. It supports text, images, sounds, and video, and includes programming interfaces for C / C++, Java, Perl, Python, Ruby, Tcl and Open Database Connectivity (ODBC). PostgreSQL supports a large part of the SQL standard and offers many modern features like Complex SQL queries, SQL Sub-selects, Foreign keys, Trigger, Views, Transactions, Multiversion concurrency control (MVCC), Streaming Replication (as of 9.0), Hot Standby (as of 9.0).

9.AM Charts (JavaScript Charts)

AM Charts made it easy to display complex data visualizations. Combine various graph types on a single chart. Create clusters, or stacks, or clusters of stacks. Control the widths, open and close values, apply coloring based on value thresholds or changes, recalculate the values automatically. Use various value scales, including date and time. Those are just a few examples of what we can do.

Features of AM Charts

- **Interactive**

Zoom or pan serial charts, drill-down to other data levels, select slices, toggle graphs using legend, display HTML-rich contextual info, or draw trend lines directly on chart.

- **Export options**

Annotate and export charts dynamically to various formats including static images, SVG, PDF, Excel, and CSV.

- **Load external data**

Easily setup and load external data sources in JSON or CSV formats. Enable reloads. Add custom pre-processing functions.

- **Extendable**

Enhance charting capabilities with a range of plugins built by amCharts team.

- **Responsive** Resize your browser window, rotate the phone, watch the chart not just take the new shape, but adapt its contents and controls accommodate available space. Use full-fledged responsive features transparently, or write your own responsive rules.

- **Mobile-friendly**

We made it extremely easy to control the charts using touch gestures. Zoom, pan, click the charts, without sacrificing the general responsiveness of the web page.

- **Accessible**

As of version 3.20 JavaScript Charts features extensive accessibility functionality right out-of-the-box. The product is fully compatible with standard-based screen readers as well as W3C-approved properties for easy navigation between map elements for people with impaired vision or with mobility restrictions. The screen reader content is even customizable per your requirements. Visit our Accessibility center for more information.

- **Dynamic**

Update data, size or just about any other configuration variable dynamically, without reloading the page. Add graphs, legends, titles, guides, bullets, or change colors, switch between 3D settings on the fly via well-documented API. Tap into chart's various events using custom handler functions.

- **Live-updated charts**

Update data every second to create 'live' charts. Simulate just about any interaction using API function calls.

4. SOFTWARE DEVELOPMENT MODEL

Agile Software Development Model

Agile methodology is an alternative to traditional project management, typically used in software development. It helps teams respond to unpredictability through incremental, iterative work cadences, known as sprints. Agile methodologies are an alternative to waterfall, or traditional sequential development. Agile development model is a type of Incremental model. Software is developed in incremental, rapid cycles. This results in small incremental releases with each release building on previous functionality. Each release is thoroughly tested to ensure software quality is maintained. Agile model is generally used for time critical applications. The project is started by preparing the backlog. The backlog contains the modular decomposition of the overall system. Then the requirement specification and system model diagrams are prepared. Agile is an adaptive model which allows continuous changes in the system requirements as well the system model.

5. REQUIREMENTS

5.1. Functional Requirements

- The system shall provide visualization of the stock market's data of individual companies.
- The system shall examine the stocks of different companies based on the technical indicators.
- The system shall predict the value of a stock based on past values of that particular stock.
- The system shall predict the increase or decrease of the NEPSE based on the news analysis.
- The system shall predict the worth of a company based on the fundamental indicators of the company

5.2. Non-Functional Requirements

- The prediction system should be dynamic enough to easily adapt with the daily increase in the data.
- The visualization system should be adaptive to dynamically add new data to the visualization.

6. SYSTEM MODEL

6.1. Backlog

Story	Task	Time Estimation (Days)	Time Estimation (Hours)
As a user, I should have access to stock data	Crawl data from Merolagani, Sharesansar and NEPSE website	7	35
	Clean the data obtained from the web	1	5
	Design a data model and represent the data in proper format	7	35
	CRUD on the available data set	7	35
	Design simple User Interface to view the stock data	6	30
	Perform Tests	2	10
As a user, I should be able to visualize the stock data for various companies and sector , compare and analyze them	Represent data in proper JSON format for visualization	3	15
	User Interface for data visualization	3	15
	Implementation of data visualization using AmCharts	7	35
	Design and write test cases	4	20
	Perform test	1	5
As a user, I should be able to view the fundamental background of a company	Represent the fundamental data in proper format	5	25
	User Interface for viewing fundamental data	4	20
	Integrate back-end and front-end	4	20
As a user I should be able to predict the company wise stock price	Design a model of data for prediction	7	35
	Design the prediction engine using ANN	25	125
	Implement the prediction engine	15	75
	Test the prediction model	4	20
	Refine the ANN model	5	25
	Design the prediction engine using KNN	5	25
	Implement the prediction engine	2	10
	Test the prediction engine	2	10
As a user I should be able to predict the sector wise index	Design a news data model data for prediction	10	50
	Design a prediction using KNN	5	25
	Implement the prediction engine	2	10
	Test the prediction engine	2	10

6.2. Dataflow Diagram

The level 0 dataflow diagram is shown in Figure 6.1.



Figure 6.1: Dfd level 0

The level 1 dataflow diagram is shown in Figure 6.2.

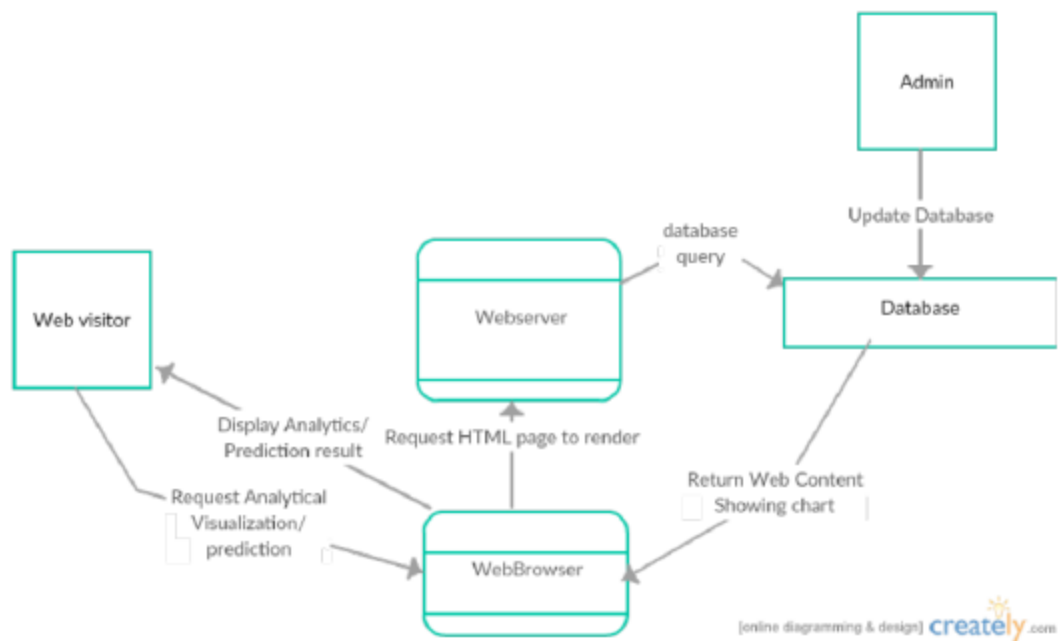


Figure 6.2: Dfd level 1

6.3. Sequence Diagram

The sequence diagram for the stock prediction is shown below in Figure 6.3.

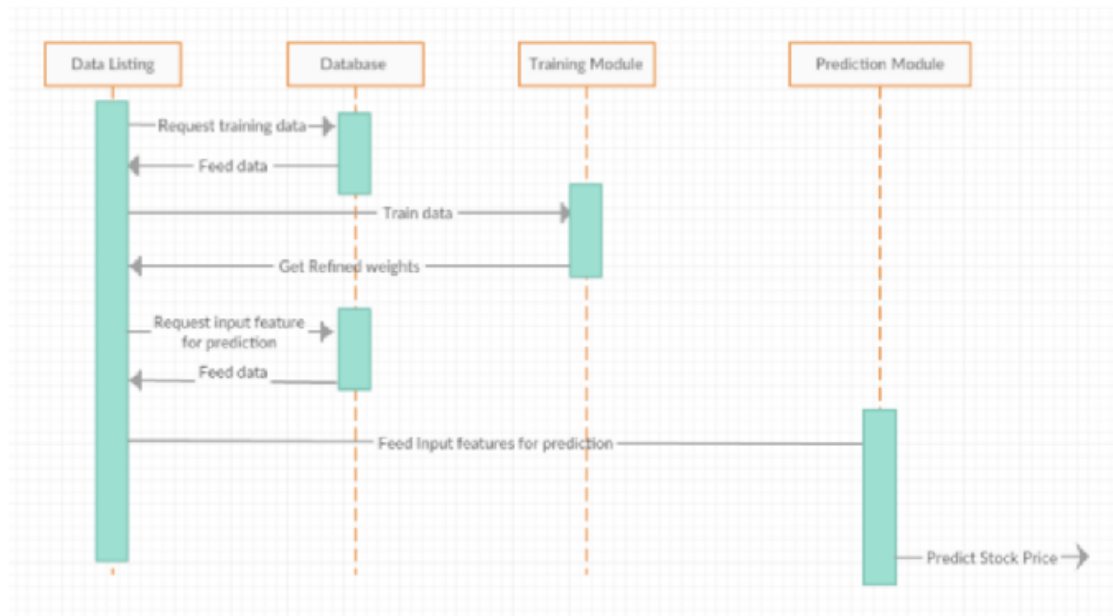


Figure 6.3: Sequence diagram for stock prediction

The sequence diagram for the stock analysis is shown in Figure 6.4.

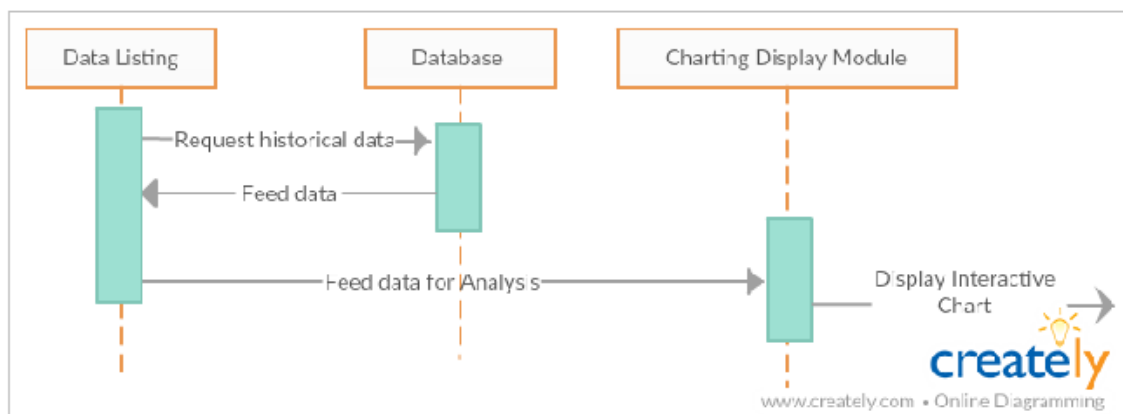


Figure 6.4: Sequence diagram for stock analysis

6.4. Use Case Modeling

A use case diagram shows the various actors that can act upon the system and what actions they can perform. The use case diagram for the system is depicted below in Figure 6.5.

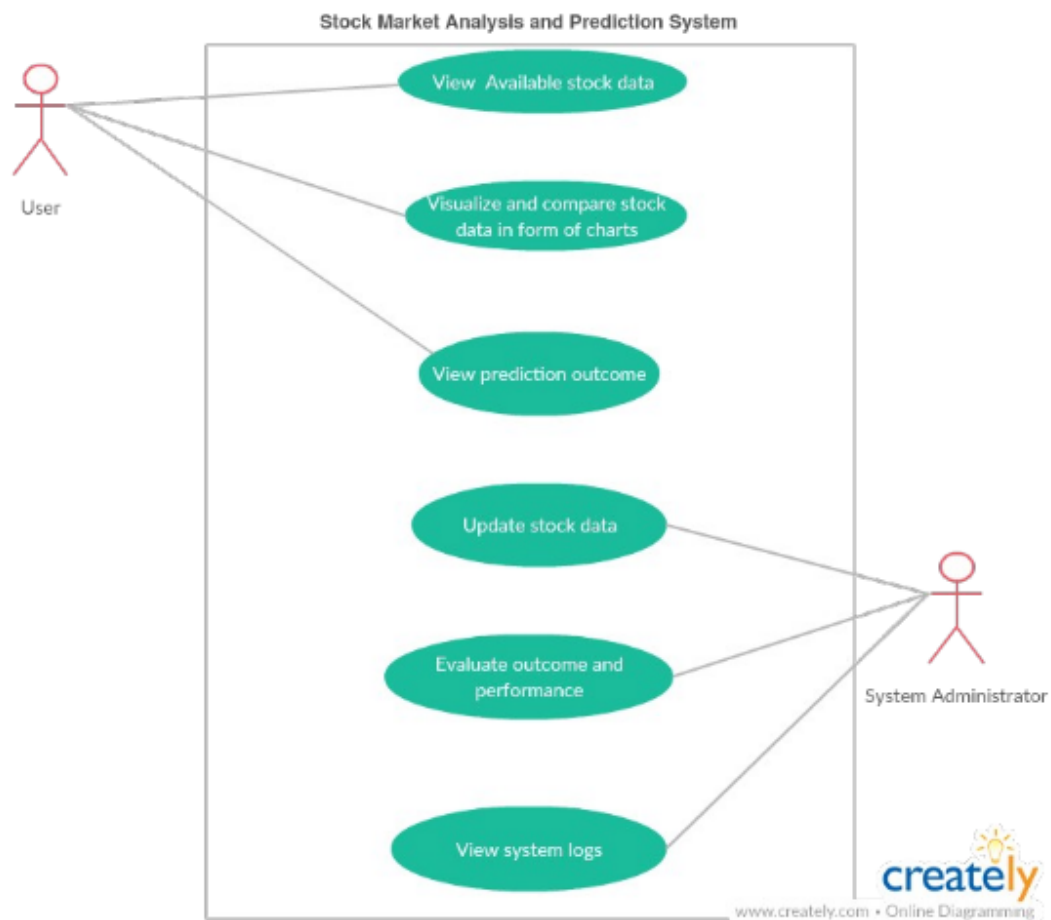


Figure 6.5: Top level usecase diagram for stock market analysis and prediction

6.5. Activity Diagram

The activity diagram for the system is shown in Figure 6.6.

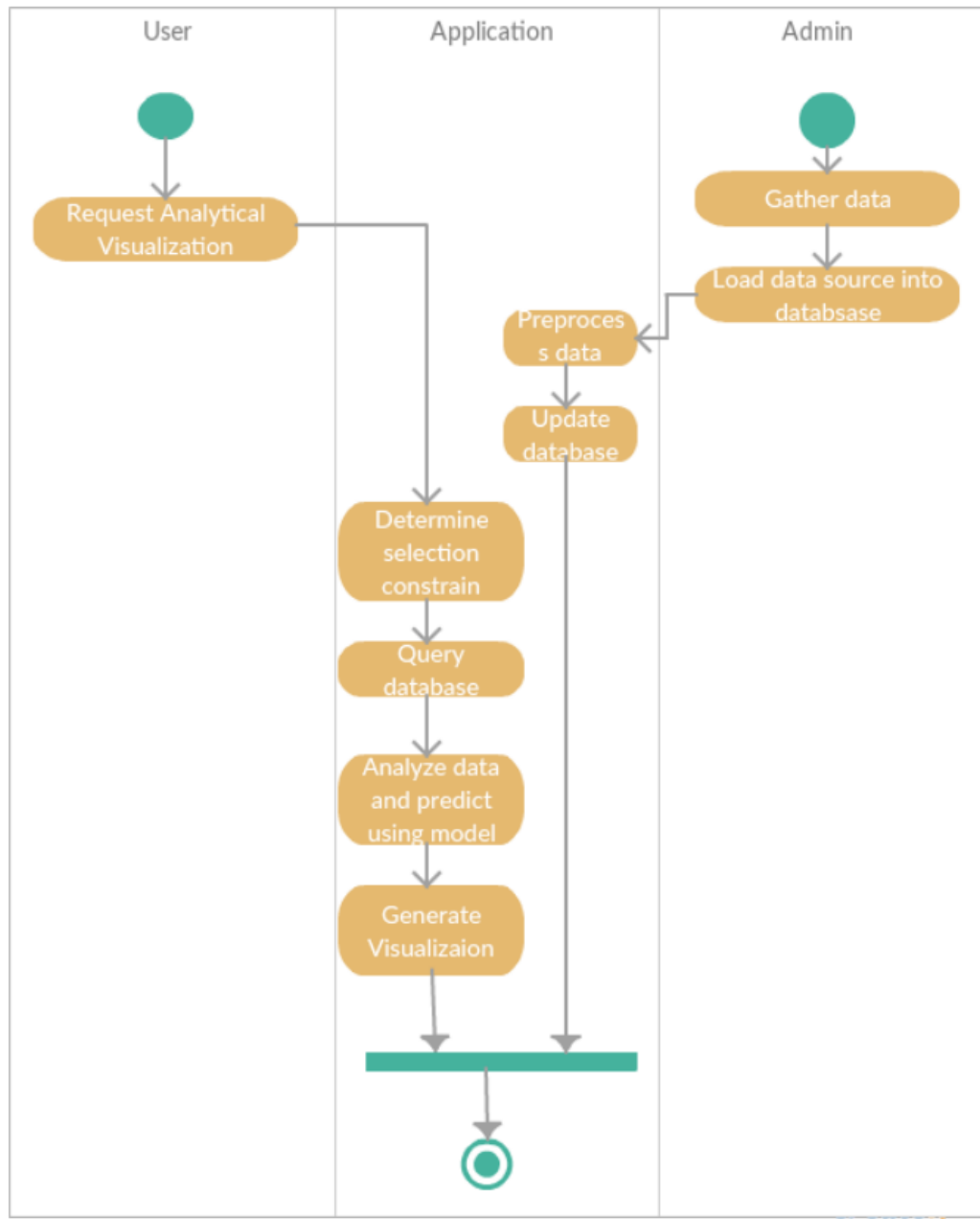


Figure 6.6: Activity diagram for stock market analysis and prediction

7. CONCLUSION

In this project, Stock Price of individual companies and sectors were predicted for the context of Nepal. Numeric data were collected from NEPSE website, Merolagani website and Sharesansar website. The price for the following day were successfully predicted with considerable accuracy. Alongside this, the strength of each companies were measured with the help of fundamental analysis. Web application has been developed in which user can see different data visualization and the predicted stock price.

REFERENCES

- [1] <http://airccj.org/CSCP/vol5/csit54304.pdf>
- [2] https://en.wikipedia.org/wiki/Multilayer_perceptron
- [3] Stock Prediction using Artificial Neural Networks, Abhishek Kar (Y8021), Dept. of Computer Science and Engineering, IIT Kanpur
- [4] kalyani Joshi, Prof. Bharathi H.N. ,Prof. Jyothi Rao, *Stock trend prediction using news sentiment analysis*, Retrieved from <https://arxiv.org/pdf/1607.01958.pdf>.
- [5] <http://searchbusinessanalytics.techtarget.com/definition/data-visualization>
- [6] <http://machinelearningmastery.com/feature-selection-machine-learning-python/>
- [7] Jan Ivar Larsen, (2010), *Predicting stock prices using technical analysis and machine learning*, Retrieved From <http://www.diva-portal.org/smash/get/diva2:354463/fulltext01.pdf>.
- [8] Basanta Joshi, Artificial Intelligence,<http://www.basantajoshi.com.np/courses/AI/>.
- [9] Stock Market *Wikipedia*,https://en.wikipedia.org/wiki/Stock_market.
- [10] Investopedia Staff(2014),*Stock Basic Tutorial [Online tutorial]*,Retrieved on September 16, 2014 from <http://www.investopedia.com/university/stocks/>.
- [11] S.Prasanna , Dr.D.Ezhilmaran (2013),*An analysis on stock market prediction using data mining techniques*,<http://www.ijcset.com/docs/IJCSET13-04-02-004.pdf>.
- [12] Soban Kumar Khadka ,*Historic Stock Market Crashes, Bubbles Financial Crises*,<http://nepalcastudent.com/old/index.php/economy-management/76-how-to-calculate-nepse-index-with-illustrative-example>.