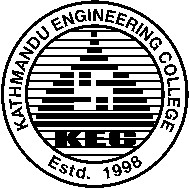
**TRIBHUVAN UNIVERSITY**

**INSTITUTE OF ENGINEERING**

Kathmandu Engineering College

Department of Computer Engineering



Mid-Term Project Report

On

**Stock Market Prediction**

[Code No: CT 654]

By

Abhishek Man Shrestha - 078BCT009

Ayush Acharya - 078BCT022

Bishranta Regmi - 078BCT027

Dhiraj Sah - 078BCT030

Kathmandu, Nepal

2081

# ABSTRACT

This project aims to create a stock market prediction system that combines historical dataset analysis with sentiment analysis from news articles and social media. By integrating machine learning algorithms, we seek to enhance the accuracy of forecasting future stock prices. Historical stock price data will be collected from reliable sources like NEPSE and NepseAlpha and sentiment data will be extracted using natural language processing techniques and algorithm like CNN. Through preprocessing and feature extraction, sentiment scores will be fused with historical data analysis to refine predictive models. Various machine learning algorithms, such as LSTM, CNNs, RNNs, will be explored for both dataset analysis and sentiment analysis. The project will conclude with the development of a user-friendly interface in the form of web app for visualizing prediction results and insights. Continuous evaluation and refinement of models will ensure their efficiency.

Keywords: *stock, historical dataset, sentiment, analysis*

# TABLE OF CONTENTS

[ABSTRACT i](#_Toc172631208)

[TABLE OF CONTENTS ii](#_Toc172631209)

[LIST OF FIGURES iv](#_Toc172631210)

[LIST OF ABBREVIATIONS v](#_Toc172631211)

[CHAPTER 1: INTRODUCTION 1](#_Toc172631212)

[1.1 BACKGROUND THEORY 1](#_Toc172631213)

[1.2 PROBLEM STATEMENT 2](#_Toc172631214)

[1.3 OBJECTIVES 2](#_Toc172631215)

[1.4 SCOPE AND APPLICATIONS 2](#_Toc172631216)

[CHAPTER 2: LITERATURE REVIEW 3](#_Toc172631217)

[2.1 EXISTING SYSTEMS 4](#_Toc172631218)

[2.1.1 QuantConnect 4](#_Toc172631219)

[2.1.2 MetaTrader5 4](#_Toc172631220)

[2.1.3 AlphaSense 5](#_Toc172631221)

[2.2 LIMITATIONS OF PREVIOUS SYSTEMS 5](#_Toc172631222)

[2.3 SOLUTIONS PROPOSED BY OUR SYSTEM 5](#_Toc172631223)

[CHAPTER 3: METHODOLOGY 6](#_Toc172631224)

[3.1 PROCESS MODEL 6](#_Toc172631225)

[3.1.1 ITERATIVE MODEL 6](#_Toc172631226)

[3.2 BLOCK DIAGRAM 8](#_Toc172631227)

[3.3 ALGORITHMS 10](#_Toc172631228)

[3.3.1 Recurrent Neural Network (RNNs) 10](#_Toc172631229)

[3.3.2 Convolutional Neural Networks (CNNs) 10](#_Toc172631230)

[3.3.3 Long Short-Term Memory (LSTM) 11](#_Toc172631231)

[3.4 FLOWCHART 12](#_Toc172631232)

[3.5 Data Collection 13](#_Toc172631233)

[3.5.1 Parameters 13](#_Toc172631234)

[3.5.2 Data Preprocessing 14](#_Toc172631235)

[3.5.3 Data Normalization 14](#_Toc172631236)

[3.6 Necessary UML Diagrams 14](#_Toc172631237)

[3.6.1 Data Flow Diagram 14](#_Toc172631238)

[3.6.2 Use Case Diagram 17](#_Toc172631239)

[3.6.3 UML Sequence Diagram 18](#_Toc172631240)

[3.7 TOOLS TO BE USED 19](#_Toc172631241)

[3.7.1 Python 19](#_Toc172631242)

[3.7.2 HTML/CSS 19](#_Toc172631243)

[3.7.3 Flask 19](#_Toc172631244)

[3.7.4 Stack Overflow 19](#_Toc172631245)

[3.7.5 Github 19](#_Toc172631246)

[3.7.6 ChatGPT 19](#_Toc172631247)

[CHAPTER 4: EPILOGUE 20](#_Toc172631248)

[4.1 TASK COMPLETED 20](#_Toc172631249)

[4.2 TASK REMAINING 20](#_Toc172631250)

[4.2 GANTT CHART 21](#_Toc172631251)

[REFERENCES 22](#_Toc172631252)

[BIBLIOGRAPHY 23](#_Toc172631253)

[SCREENSHOTS 24](#_Toc172631254)

# LIST OF FIGURES

Figure 1: Iterative Process Model .................................................................................6

Figure 2: Block Diagram of the Proposed System .......................................................8

Figure 3: Recurrent Neural Network...........................................................................10

Figure 4: Convolution Neural Network.......................................................................10

Figure 5: Long Short-Term Memory...........................................................................11

Figure 6: Flowchart......................................................................................................12

Figure 7: Data Flow Diagram Level 0……………………………………………….14

Figure 8: Data Flow Diagram Level 1……………………………………………….15

Figure 9: Data Flow Diagram Level 2……………………………………………….16

Figure 10: Use Case Diagram…………………………………………………...…...17

Figure 11: UML Sequence Diagram…………………………………………………18

Figure 12: Gantt Chart.................................................................................................21

# LIST OF ABBREVIATIONS

**ANN** Artificial Neural Network

**BGRU** Bidirectional Gated Recurrent Unit

**CNN** Convolution Neural Network

**CSS** Cascading Style Sheets

**HTML** HyperText Markup Language

**HTTP** HyperText Transfer Protocol

**LSTM** Long Short-Term Memory

**RNN** Recurrent Neural Network

**SVM** Support Vector Machine

# CHAPTER 1: INTRODUCTION

## 1.1 BACKGROUND THEORY

The stock market is a system where people can buy and sell shares of publicly traded companies. Companies issue shares of stock to the public to raise funds for their operations and growth. The stock market operates on the principle of supply and demand. When demand for a company's stock is high, its share price rises, and vice versa. Investors purchase these shares because they believe the company will succeed and grow over time, increasing the value of their investment. If the company performs well, the stock price usually goes up, allowing investors to sell their shares at a higher price and make a profit.

The Nepal Stock Exchange (NEPSE) is the main stock exchange of Nepal. NEPSE opened its trading floor on 13 January 1994. It was established to provide a marketplace for the trading of securities in Nepal. The exchange facilitates transactions for both individual and institutional investors and plays a vital role in the economic development of Nepal by enabling companies to access public funds for expansion and growth.

Stock market prediction involves forecasting the future value of a company's stock, which is crucial for economic growth. By studying patterns in market data, we can predict when to buy stocks and potentially earn profits. This prediction relies on understanding supply and demand. By analyzing large amounts of market data, we can find patterns that give us a good chance of making accurate predictions. Stock market prediction consists of statistical analysis, machine learning, and sentiment analysis to forecast future stock prices using historical data and market sentiment. Integrating these methods and using machine learning helps to develop predictive models offering valuable insights of the market.

## 1.2 PROBLEM STATEMENT

The stock market is vast and unpredictable, with significant fluctuations making it challenging to anticipate whether prices will rise or fall the next day. Investing in stocks can result substantial profits or result in significant losses, making it a scary task for many. Manual methods of predicting future market trends are tedious and often inaccurate, especially when the financial markets are influenced by factors like politics, economics, and investor psychology.

## 1.3 OBJECTIVES

This project was created with the purpose to fulfill the following objectives:

* To analyze historical trading data (date, closing price, etc.) and sentimental data (news headlines, articles, posts, etc.) to understand market trends.
* To predict future price using the information from analyzed data implementing machine learning algorithms like RNN, CNN and LSTM.

## 1.4 SCOPE AND APPLICATIONS

The applications of this project are as follows:

* To bridge communication between financial investors and industries as investors can find right companies for investments and companies can find investors for their growth
* To assist the investors and people to make sensible decisions for profitable investments through analysis of historical trading data.

# CHAPTER 2: LITERATURE REVIEW

A study done by Polepally, Vijay & Jakka, Neha & Vishnukanth, Pendly & Raj, Rachakatla & Anish, Gudavelli presents a comprehensive study of machine learning algorithms for stock price prediction. In their 2023 study, they explored the effectiveness of the LSTM-RNN algorithm for predicting stock prices. Presented at the 7th International Conference on Intelligent Computing and Control Systems (ICICCS), their research highlights how LSTM-RNN can capture long-term dependencies in sequential data, making it highly suitable for stock market prediction. This algorithm is adept at handling time series data, which is critical for accurate forecasting in financial markets. However, their approach primarily focuses on historical data without incorporating sentiment analysis, potentially limiting its predictive capabilities in response to market sentiment shifts [1].

Lal and Timalsina's 2022 paper, presented at the 11th IOE Graduate Conference, investigated the use of a CNN-BGRU method for stock price prediction. The combination of Convolutional Neural Networks (CNN) for feature extraction and Bidirectional Gated Recurrent Units (BGRU) for handling sequential data shows promise in forecasting stock prices. Their method leverages CNN’s strength in identifying important patterns in data and BGRU’s ability to process time-dependent sequences effectively. Despite its potential, this method can be complex and resource-intensive, and it does not explicitly include sentiment analysis, which could enhance prediction accuracy [2].

In the 2023 publication in the International Journal of Creative Research Thoughts (IJCRT), Kabir, Sobur, and Amin compared various machine learning models for stock price prediction. Their study evaluates the performance of different algorithms under diverse market conditions, providing insights into their strengths and weaknesses. While their comprehensive approach offers valuable comparisons, it does not integrate additional data sources such as sentiment analysis from news or social media, which can be crucial for improving the accuracy of stock predictions [3].

Choi’s 2018 research discusses a hybrid model combining ARIMA and LSTM for predicting stock price correlations. The hybrid ARIMA-LSTM model leverages the statistical strengths of ARIMA and the deep learning capabilities of LSTM, resulting in improved accuracy for correlation predictions between stocks. This approach demonstrates the benefits of combining traditional statistical methods with modern machine learning techniques. However, the model's complexity may pose challenges for practical implementation, and it focuses on correlation prediction rather than direct price forecasting [4].

Patel, J., Shah, S., Thakkar, P., & Kotecha, K. in their 2015 paper published in Expert Systems with Applications, Patel and colleagues examined various machine learning techniques, including ANN, SVM, random forest, and naive Bayes, for predicting stock prices and index movements. Their study emphasizes the importance of trend deterministic data preparation to enhance the accuracy of machine learning models. By comparing these techniques, the authors provide insights into their effectiveness for financial forecasting. Despite the comprehensive analysis, the study does not incorporate sentiment analysis, which could provide a more holistic view of market dynamics [5].

## 2.1 EXISTING SYSTEMS

2.1.1 QuantConnect

An algorithmic trading platform that allows users to create and back test trading strategies which requires a strong understanding of programming and algorithmic trading concepts, making it less accessible to beginners.

2.1.2 MetaTrader5

A popular trading platform offering tools for technical analysis and automated trading strategies that primarily focused on technical analysis and doesn't inherently support machine learning models or sentiment analysis.

2.1.3 AlphaSense

An AI-powered market intelligence platform that aggregates and analyzes financial data, news, and research reports but doesn't provide specific stock price prediction capabilities.

## 2.2 LIMITATIONS OF PREVIOUS SYSTEMS

* Many platforms like QuantConnect require advanced programming knowledge and have steep learning curves, making them less accessible to beginner users.
* Advanced platforms like AlphaSense have high subscription fees, which can be prohibitive for individual investors or small teams.
* Most tools focus primarily on technical analysis and do not natively support the integration of sentiment analysis from news and social media, limiting the comprehensiveness of predictions.

## 2.3 SOLUTIONS PROPOSED BY OUR SYSTEM

* Use of effective model: This project uses the LSTM algorithm, which excels in stock price forecasting due to its ability to handle time series data and avoid memory issues which is a major problem with RNN.
* Integration of Sentiment Analysis: This system incorporates sentiment analysis from news and social media to further enhance prediction reliability using CNN algorithm. By combining LSTM with CNN, our system aims to deliver superior accuracy in predicting stock values.
* Accessibility: Using widely known programming languages, frameworks, open-source libraries and tools keeps costs low, makes the solution more accessible to broader audience, individual users and small teams.

# CHAPTER 3: METHODOLOGY

## 3.1 PROCESS MODEL

### 3.1.1 ITERATIVE MODEL

*Figure 1: Iterative Process Model*

In the iterative model, the process starts with a simple implementation of the software requirements and iteratively enhances the versions until the complete system is implemented and ready to be deployed.

The various phases of iterative model are as follows:

**1. Requirement analysis:** In the first phase, the identification of the key requirements for the stock prediction system, such as historical data analysis, sentiment analysis, and the integration of both is done. To develop the software under the incremental model, this phase performs a crucial role.

**2. Design:** In this phase, design of the features to be used in the model, such as moving averages from historical data and sentiment scores from text data are done along with simple LSTM model for time-series prediction using historical data and a basic sentiment analysis module to process textual data.

**3. Implementation:** Implementation phase enables the coding phase of the development system. It involves the final coding that implements a basic LSTM model for predicting stock prices based on historical data and integrate a sentiment analysis module using libraries like TextBlob or NLTK.

**4. Testing:** The testing phase checks the performance of each existing function, tests the LSTM model's accuracy in predicting stock prices and verifies that the sentiment scores are correctly influencing the predictions.

**5. Review:** After testing the model's performance, identifying areas for improvement, such as enhancing sentiment analysis or refining the LSTM model is done. Planning to integrate more advanced techniques or additional data sources in the next iteration is done if necessary.

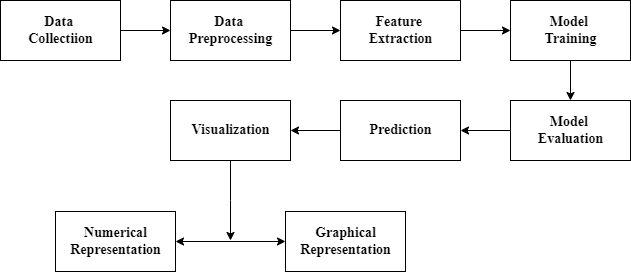
**6. Deployment:** After completing all the phases, software is deployed to its work environment where users can interact with it and provide further feedback if any improvement or correction is required.

**7. Maintenance:** In the maintenance phase, after deployment of the software in the working environment there may be some bugs, some errors or new updates are required. Maintenance involves debugging and addition of new options.

Advantages of iterative process model:

* Repeated cycles of development allow for continuous improvement and refinement of the software.
* Early prototypes provide a possible product that user can review and provide feedback on.
* Frequent testing and evaluation throughout each iteration help identify and fix defects early.
* Flexibility in adjusting changes is increased, as each iteration provides an opportunity to adjust requirements based on user feedback and evolving needs.

## 3.2 BLOCK DIAGRAM



*Figure 2: Block Diagram of the Proposed System*

This block diagram outlines the main components of the stock prediction system that are explained below:

**1. Data Collection:** This step includes gathering of historical stock price data of companies from credible sources like NEPSE, Nepse Alpha, ShareSansar, etc. and textual data from various sources like news websites, social media, and financial forums related to specific companies by scrapping the data available for general public.

**2. Data Preprocessing:** This involves cleansing and normalization of historical stock data like handling missing values and scaling, and creating additional features like moving averages. This also include tokenization, stop-word removal of textual data that prepares the text for sentiment analysis by converting it into a format suitable for machine learning algorithms.

**3. Feature Extraction:** Extraction of relevant features and technical indicators from the preprocessed data is done in this step. The features may include moving averages, Relative Strength Index (RSI), sentiment scores, sentiment polarity and sentiment subjectivity.

**4. Model Training:** LSTM (Long Short-Term Memory) neural network is employed for time-series analysis of historical stock data and utilization of CNN (Convolutional Neural Network) algorithm is done to analyze textual data and extract sentiment-related features. The models are properly trained using the collected datasets to provide output as accurate as possible.

**5. Model Integration:** The extracted features from both historical and textual data are combined into a single feature set. This integrated feature set provides comprehensive input for the predictive models, capturing both market trends and sentiment factors. This can be achieved by libraries like TextBlob or NLTK.

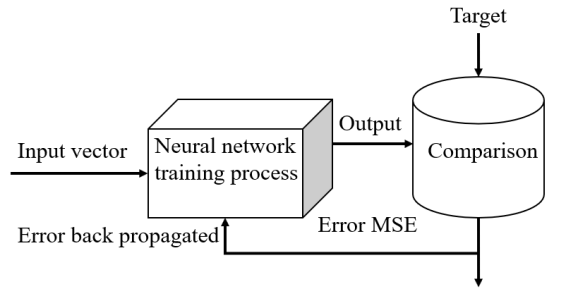
**5. Model Evaluation:** Evaluation of the LSTM model's accuracy in predicting stock prices based on given data is done and verification is done if sentiment scores are correctly influencing the predictions or not.

**6. Prediction:** The integrated model generates predictions for future stock prices based on the combined feature set. These predictions provide valuable insights into potential market movements and assist investors in making informed decisions.

**7. Visualization:** The predicted stock prices, along with relevant analytics and insights, are visualized and presented in user-friendly formats in a web application such as charts and graphs. These visualizations can help in interpreting the model predictions and understanding market trends.

## 3.3 ALGORITHMS

### 3.3.1 Recurrent Neural Network (RNNs)

Recurrent Neural Networks (RNNs) are a type of neural network that are designed to process sequential data. They can analyze data with a temporal dimension, such as time series, speech, and text. RNNs can do this by using a hidden state passed from one timestep to the next. The hidden state is updated at each timestep based on the input and the previous hidden state.

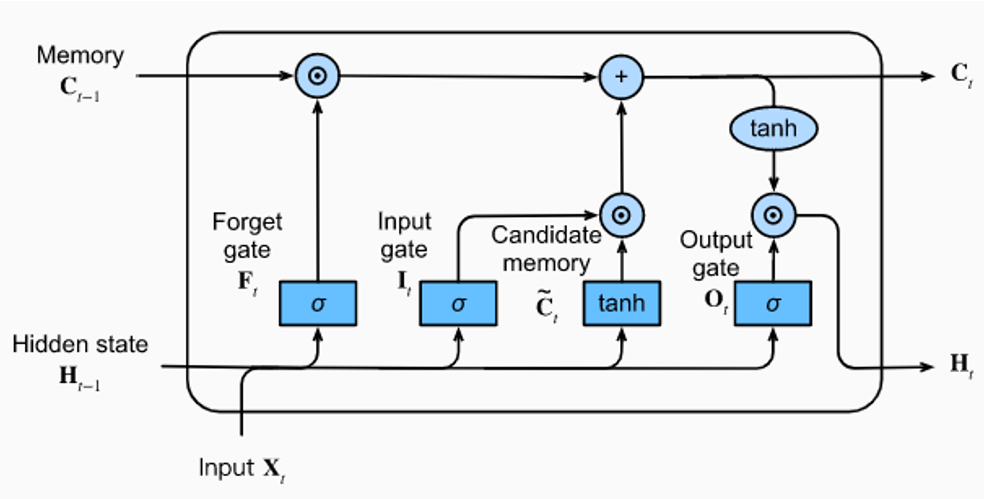
*Figure 3: Recurrent Neural Network*

### 3.3.2 Convolutional Neural Networks (CNNs)

 Convolutional Neural Networks (CNNs) are powerful deep learning models primarily used for image and text data analysis. In the context of stock market prediction, CNNs excel at sentiment analysis by extracting meaningful patterns from textual data such as news articles and social media posts. By identifying sentiments and trends, CNNs enhance the overall predictive accuracy when combined with time-series models like LSTM.

*Figure 4: Convolution Neural Network*

### 3.3.3 Long Short-Term Memory (LSTM)

LSTMs Long Short-Term Memory is a type of RNNs that can detain long-term dependencies in sequential data. They use memory cell and gates to control the flow of information, allowing them to selectively retain or discard information as needed and thus avoid the vanishing gradient problem that plagues traditional RNNs. LSTMs are widely used in applications such as [natural language processing](https://www.simplilearn.com/tutorials/artificial-intelligence-tutorial/what-is-natural-language-processing-nlp), [speech recognition](https://www.simplilearn.com/tutorials/python-tutorial/speech-recognition-in-python), and time series forecasting.

*Figure 5: Long Short-Term Memory*

There are three types of gates in an LSTM:

**Input gate:** It decides which information to store in the memory cell. It is trained to open when the input is important and close when it is not.

**Forget gate:** It decides which information to discard from the memory cell. It is trained to open when the information is no longer important and close when it is.

**Output gate:** It is responsible for deciding which information to use for the output of LSTM. It is trained to open when the information is important and close when it is not.

The gates in an LSTM are trained to open and close based on the input and the previous hidden state. This allows the LSTM to selectively retain or discard information, making it more effective at capturing long-term dependencies.

## 3.4 FLOWCHART

*Figure 6: Flowchart*

## 3.5 Data Collection

To accurately predict stock prices, we need a lot of data that's both reliab’ e and correct. We got our data from the Nepal Stock Exchange (NEPSE) after getting formal permission from the bank. This data includes things like the highest price, lowest price, closing price, and how much of the stock was traded. We use this information to train our prediction system and see how well it works.

### 3.5.1 Parameters

#### 3.5.1.1 High Price

The high price is the price at which a highest closing price of a stock over past 52 weeks. adjusted for any stock splits or the highest intraday price of a stock in the most recent trading sessions.

#### 3.5.1.2 Closing Price

The closing price is the final price at which a share is traded on a given trading day. The closing price represents the most up-to-date price of a share until trading commences again on the next trading day. For example, NEPSE closes precisely at 3:00 pm on working days**.**

#### 3.5.1.3 Stock Volume

The total amount of stock traded in a trading day is the stock volume of that day. For example, if there are two companies A and B with company A's 100 stock traded and B's 200 stocks traded then the Stock Volume=300.

#### 3.5.1.4 Low Price

The low price is the price which is the lowest closing price of a stock over past 52 weeks, adjusted for any stock splits or the lowest intraday price of a stock in the most recent trading sessions.

#### 3.5.1.5 Opening Price

The opening price is the price at which a security first trades upon the opening of an exchange on a trading day; for example, the NEPSE opens at precisely 11:00 a.m. Nepalese time. The price of the first trade for any listed stock is its daily opening price.

### 3.5.2 Data Preprocessing

The raw data that had been previously collected are always in the required format to be fed into the artificial neural network. If wrong data is fed into the system, the system gets changed according to the wrong data which decreases the accuracy of the system So, the data were preprocessed and the incorrect data, missing data were corrected and added in the system.

### 3.5.3 Data Normalization

The data collected for different parameters vary from each other over a wide range which may reduce the efficiency of the system. So, the data needs to be normalized before being fed into the system. The data are normalized in the range [0, 1] using minmax function as below:

**Normalized value of X= (X- Min value of X)/(MAX-MIN)**

## 3.6 Necessary UML Diagrams

### 3.6.1 Data Flow Diagram

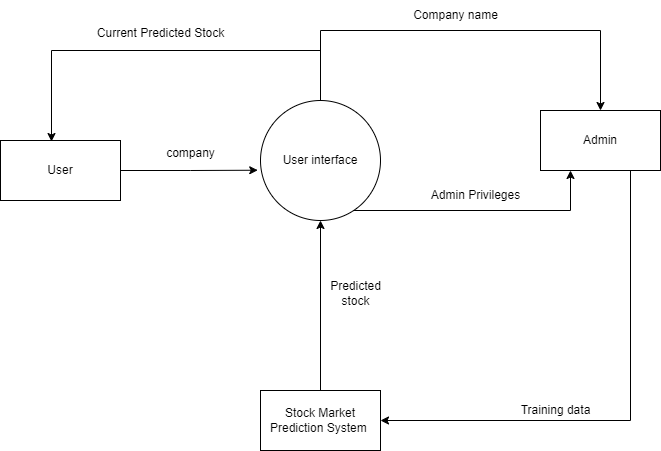
The Data Flow Diagram of the Stock Market Prediction is shown below:

**Data Flow Diagram - 0**



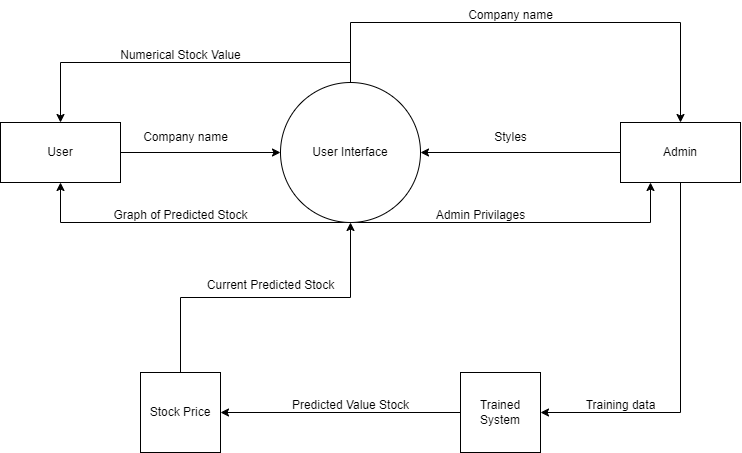
*Figure 7: Data Flow Diagram 0*

**Data Flow Diagram – 1**



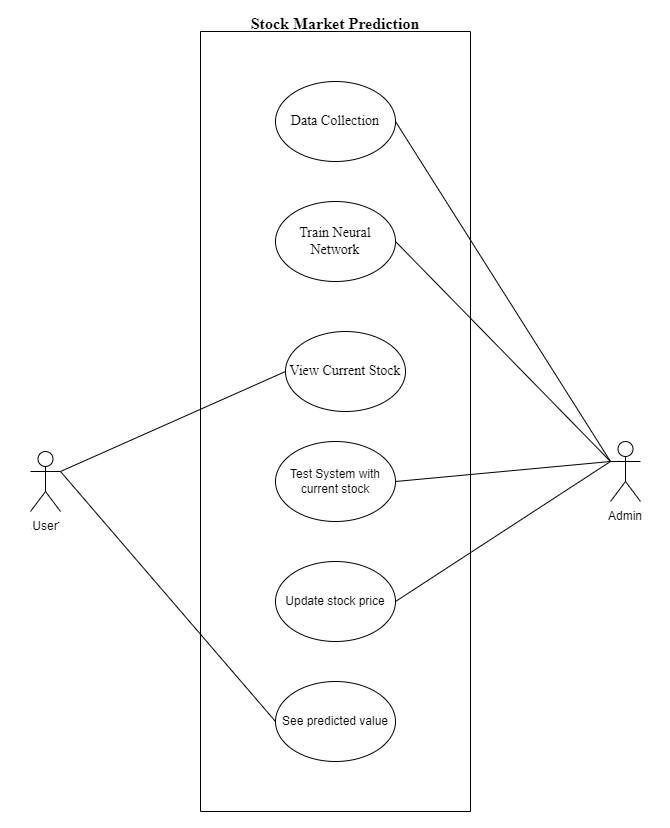
*Figure 8: Data Flow Diagram Level - 1*

**Data Flow Diagram Level – 2**



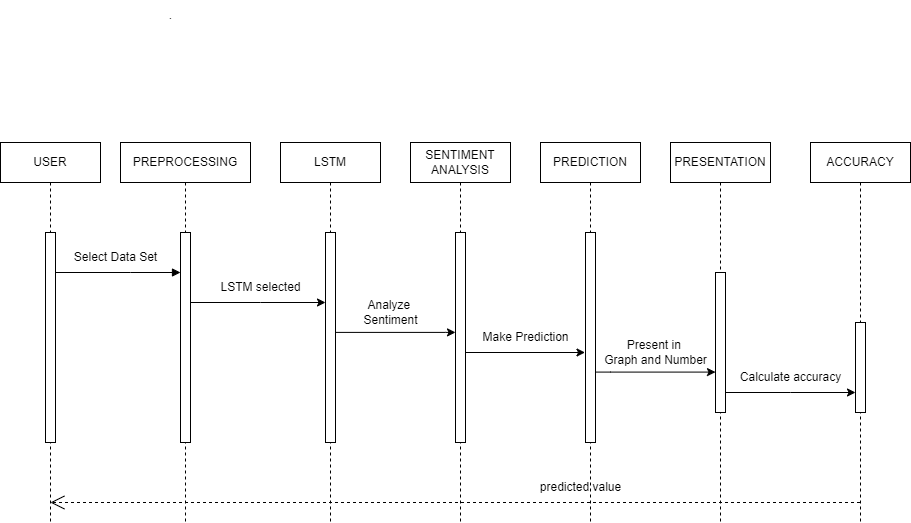
*Figure 9: Data Flow Diagram Level-2*

### 3.6.2 Use Case Diagram



*Figure 10: Use Case Diagram*

### 3.6.3 UML Sequence Diagram



*Figure 11: UML Sequence Diagram*

## 3.7 TOOLS TO BE USED

### 3.7.1 Python

Python is used for backend development, data preprocessing, and machine learning model implementation. Python libraries like NumPy, Pandas facilitate data analysis, machine learning tasks, and natural language processing.

### HTML/CSS

HTML structures the content of web pages, while CSS styles and formats their appearance. HTML and CSS are employed for frontend development, creating the user interface for displaying prediction results.

### Flask

Flask serves as the backend framework. Flask integrates the frontend of web application with the backend of prediction model done in python, enabling the creation of dynamic interface for users to interact.

### Stack Overflow

Stack Overflow serves as a valuable resource for troubleshooting technical issues and seeking solutions to coding challenges. Developers can leverage Stack Overflow to find answers, ask questions, and access a vast repository of knowledge on relevant topics.

### Github

GitHub hosts the project repository, facilitating version control, collaboration, and code management. Developers can use GitHub to share code, collaborate with team members, and contribute to open-source projects relevant to stock market prediction.

### ChatGPT

ChatGPT provides assistance and guidance on project-related topics through natural language conversations. Developers can interact with ChatGPT to brainstorm ideas, seek advice, and access resources for refining their project strategy and implementation.

# CHAPTER 4: EPILOGUE

## 4.1 TASK COMPLETED

* **Data Collection**: Collected and organized historical stock data for multiple companies.
* **User Interface Design**: Designed and implemented the home, login, and signup pages with a consistent dark theme.
* **Functionality**: Integrated basic functionality for selecting parameters to predict price of stock company symbols.

## 4.2 TASK REMAINING

* **Backend Integration**: Develop backend logic to process user inputs and fetch corresponding stock data.
* **Data Analysis**: Integrate historical data analysis and sentiment analysis for enhanced stock predictions.
* **UI Enhancements**: Improve UI responsiveness, add form validation, and enhance data visualization with interactive charts.
* **Additional Features**: Develop detailed profile and about pages, and implement password recovery and account management.
* **Testing and Debugging**: Conduct thorough testing, fix bugs, and optimize performance for faster data processing.
* **Deployment**: Set up hosting, deploy the application to a live server, and monitor for issues while gathering user feedback.

## 4.2 GANTT CHART

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Tasks | JUN | JUL | AUG | SEP | OCT | NOV | DEC | JAN | FEB | MAR |
| Documentation |  |  |  |  |  |  |  |  |  |  |
| Planning |  |  |  |  |  |  |  |  |  |  |
| Analysis |  |  |  |  |  |  |  |  |  |  |
| Design |  |  |  |  |  |  |  |  |  |  |
| Implementation |  |  |  |  |  |  |  |  |  |  |
| Testing |  |  |  |  |  |  |  |  |  |  |

*Figure 12: Gantt Chart*

# REFERENCES

1. Polepally, Vijay & Jakka, Neha & Vishnukanth, Pendly & Raj, Rachakatla & Anish, Gudavelli. (2023). An Efficient Way to Predict Stock Price using LSTM-RNN Algorithm. *7th International Conference on Intelligent Computing and Control Systems (ICICCS), 1240-1244,* 2023.
2. Janak Kumar Lal, Arun Kumar Timalsina. (2022). A CNN-BGRU Method for Stock Price Prediction. Proceedings of 11th IOE Graduate Conference, *2350-8914 (Online), 2350-8906 (Print),* 2022.
3. Kabir, Md Humayun & Sobur, Abdus & Amin, Md Ruhul. (2023). Stock Price Prediction Using the Machine Learning. *International Journal of Creative Research Thoughts (IJCRT). f946-f950,* 2023
4. Choi, Hyeong. (2018). Stock Price Correlation Coefficient Prediction with ARIMA-LSTM Hybrid Model, 2018.
5. Patel, J., Shah, S., Thakkar, P., & Kotecha, K. (2015). Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques. Expert Systems with Applications, 42(1), 259-268.

# BIBLIOGRAPHY

* https://www.simplilearn.com/tutorials/artificial-intelligence-tutorial/lstm
* https://www ijcrt.org/papers/IJCRT2307700.pdf
* https://ieeexplore.ieee.org/abstract/document/10142902
* https://www.youtube.com/watch?v=b61DPVFX03I
* http://conference.ioe.edu.np/publications/ioegc11/ioegc-11-004-11005.pdf
* <https://www.geeksforgeeks.org/convolutional-neural-network-cnn-in-machine-learning>

# SCREENSHOTS

