STOCK MARKET TREND PREDICTION USING LSTM

A PROJECT REPORT

Submitted by

R MOHAMED ADHIL AMEEN – 2303811724321066

in partial fulfillment of requirements for the award of the course

AG11242-MACHINE LEARNING TECHNIQUES

in

ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

K. RAMAKRISHNAN COLLEGE OF TECHNOLOGY

(An Autonomous Institution, affiliated to Anna University Chennai and Approved by AICTE, New Delhi)

SAMAYAPURAM – 621 112

May, 2024

K. RAMAKRISHNAN COLLEGE OF TECHNOLOGY

(AUTONOMOUS)

SAMAYAPURAM – 621 112

BONAFIDE CERTIFICATE

Certified that this project report titled **"Stock Market Trend Prediction Using LSTM"** is the bonafide work of **R MOHAMED ADHIL AMEEN – 2303811724321066,** who carried out the project work under my supervision. Certified further, that to the best of my knowledge the work reported here in does not form part of any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE

Dr.T.AVUDAIAPPAN M.E.,Ph.D.,

HEAD OF THE DEPARTMENT

ASSOCIATE PROFESSOR

Department of Artificial Intelligence

K. Ramakrishnan College of Technology

of (Autonomous)

Samayapuram–621112.

SIGNATURE

Mr.T.SHYAMSUNDAR.,M.E.,

SUPERVISOR

ASSISTANT PROFESSOR

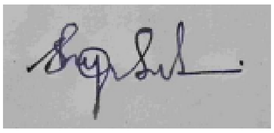
Department of Artificial Intelligence

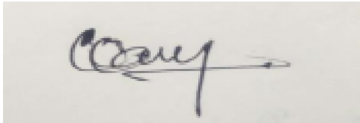
K. Ramakrishnan College

Technology (Autonomous)

Samayapuram–621112.

Submitted for the viva-voce examination held on 7/12/2024





INTERNAL EXAMINER **EXTERNAL EXAMINER**

DECLARATION

I declare that the project report on **“STOCK MARKET TREND PREDICTION USING LSTM”** is the result of original work done by us and best of our knowledge, similar work has not been submitted to **“ANNA UNIVERSITY CHENNAI”** for the requirement of Degree of **BACHELOR OF ENGINEERING**. This project report is submitted on the partial fulfillment of the requirement of the award of the course AG11242-MACHINE LEARNING TECHNIQUES

Signature

R MOHAMED ADHIL AMEEN

Place: Samayapuram

Date: 7/12/2024

ACKNOWLEDGEMENT

It is with great pride that I express our gratitude and indebtedness to our institution, "**K. Ramakrishnan College of Technology (Autonomous)**", for providing us with the opportunity to do this project.

I extend our sincere acknowledgement and appreciation to the esteemed and honorable Chairman, **Dr. K. RAMAKRISHNAN**, **B.E.,** for having provided the facilities during the course of our study in college.

I would like to express our sincere thanks to our beloved Executive Director, **Dr. S. KUPPUSAMY, MBA, Ph.D.,** for forwarding our project and offering an adequate duration to complete it.

I would like to thank **Dr. N. VASUDEVAN, M.TECH., Ph.D.,**

Principal, who gave the opportunity to frame the project to full satisfaction.

I thank **Dr.T.AVUDAIAPPAN M.E., Ph.D.,** Head of the Department of **ARTIFICIAL INTELLIGENCE**, for providing his encouragement in pursuing this project.

I wish to convey our profound and heartfelt gratitude to our esteemed project guide **Mr.SHYAMSUNDAR T M.E.,** Department of **ARTIFICIAL INTELLIGENCE,** for his incalculable suggestions, creativity, assistance and patience, which motivated us to carry out this project.

I render our sincere thanks to the Course Coordinator and other staff members for providing valuable information during the course.

I wish to express our special thanks to the officials and Lab Technicians of our departments who rendered their help during the period of the work progress.

VISION OF THE INSTITUTION

To emerge as a leader among the top institutions in the field of technical education.

MISSION OF THE INSTITUTION

Produce smart technocrats with empirical knowledge who can surmount the global challenges.

Create a diverse, fully-engaged, learner-centric campus environment to provide quality education to the students.

Maintain mutually beneficial partnerships with our alumni, industry, and Professional associations.

VISION OF DEPARTMENT

To become a renowned hub for AIML technologies to producing highly talented globally recognizable technocrats to meet industrial needs and societal expectation.

MISSION OF DEPARTMENT

Mission 1: To impart advanced education in AI and Machine Learning, built upon a foundation in Computer Science and Engineering.

Mission 2: To foster Experiential learning equips students with engineering skills to tackle real- world problems.

Mission 3: To promote collaborative innovation in AI, machine learning, and related research and development with industries.

Mission 4: To provide an enjoyable environment for pursuing excellence while upholding strong personal and professional values and ethics.

PROGRAM EDUCATIONAL OBJECTIVES

Graduates will be able to:

1. PEO1: Excel in technical abilities to build intelligent systems in the fields of AI & ML in order to find new opportunities

2. PEO2: Embrace new technology to solve real-world problems, whether alone or as a team, while prioritizing ethics and societal benefits.

3. PEO3: Accept lifelong learning to expand future opportunities in research and product development.

PROGRAM SPECIFIC OUTCOMES (PSOs)

PSO 1: Domain Knowledge

To analyze, design and develop computing solutions by applying foundational concepts of Computer Science and Engineering.

PSO 2: Quality Software

To apply software engineering principles and practices for developing quality software for scientific and business applications.

PSO 3: Innovation Ideas

To adapt to emerging Information and Communication Technologies (ICT) to innovate ideas and solutions to existing/novel problems

PROGRAM OUTCOMES (POs)

Engineering students will be able to:

Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences

Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations

Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions

Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations

The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice

Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development

Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

ABSTRACT

Stock market prediction has long been a challenging task due to the inherent volatility and complexity of financial data. Traditional methods often fail to capture the intricate patterns and long-term dependencies within stock prices. In recent years, Long Short-Term Memory (LSTM) networks, a type of Recurrent Neural Network (RNN), have demonstrated significant promise in time series forecasting, particularly for sequential data such as stock prices. This study proposes a predictive model for stock market trends using LSTM networks to predict future stock prices based on historical data. The model is trained on daily stock prices and utilizes the LSTM architecture to learn temporal dependencies, capturing both short-term fluctuations and long-term trends. The proposed approach is evaluated on various stock market datasets, comparing its performance with traditional machine learning models, such as Support Vector Machines (SVM) and Random Forests. The results show that LSTM models can effectively forecast stock trends, providing insights for traders and investors. However, the challenges of market unpredictability and external influencing factors are acknowledged, suggesting that LSTM predictions should be used as one of several tools in decision-making processes.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

List of Abbreviations:

1. **LSTM** - Long Short-Term Memory
2. **RNN** - Recurrent Neural Network
3. **SVM** - Support Vector Machine
4. **ANN** - Artificial Neural Network
5. **MAE** - Mean Absolute Error
6. **RMSE** - Root Mean Square Error
7. **MSE** - Mean Squared Error
8. **GRU** - Gated Recurrent Unit
9. **ML** - Machine Learning
10. **AI** - Artificial Intelligence

CHAPTER 1

INTRODUCTIN

1.1 INTRODUCTION TO PROJECT

The stock market has long been a domain of interest for investors, traders, and analysts, offering substantial opportunities but also posing significant risks due to its inherent volatility and unpredictable nature. Traditional methods of stock price prediction, such as technical analysis, often rely on past price movements and patterns to forecast future trends. However, these approaches may fail to capture the complex and non-linear dependencies inherent in financial data.

PURPOSE AND IMPORTANCE OF THE PROJECT

The primary purpose of this project is to leverage machine learning, specifically **Long Short-Term Memory (LSTM)** networks, to predict stock market trends and future price movements. By using historical stock data, the goal is to develop a predictive model that can identify patterns and trends in the market, enabling more informed decision-making for investors and traders. The project aims to accomplish the following objectives:

1. **Develop a Predictive Model:** Using LSTM networks, the project seeks to develop an accurate, reliable model capable of predicting stock prices or market trends based on historical data.
2. **Evaluate Model Performance:** The project will evaluate the performance of the LSTM model and compare its effectiveness to other traditional machine learning models, such as **Support Vector Machines (SVM)** and **Random Forests**. This will help determine which model offers the best prediction accuracy for stock market data.

IMPORTANCE:

The importance of this project lies in its potential to significantly enhance stock market prediction accuracy, which is crucial for making informed investment decisions in an increasingly complex and volatile financial environment. The development of a machine learning-based model, particularly using Long Short-Term Memory (LSTM) networks, has several key implications and benefits:

- **Enhanced Predictive Accuracy:**
  By using LSTM networks, the project improves the accuracy of stock market predictions, capturing complex patterns and long-term dependencies in financial data that traditional methods often miss.

- **Real-Time Decision Making & Risk Management:**
  The model enables faster, data-driven decision-making for investors and traders, helping to manage risks effectively by forecasting price movements and market trends in real time.

1.2 OBJECTIVES

1. Develop a Predictive Model for Stock Market Trends
2. Evaluate Model Performance Against Traditional Methods
3. Optimize the Model with Feature Engineering
4. Test the Model on Real Stock Market Data
5. Provide Insights for Traders and Investors

1.3 PROJECT SUMMARIZATION

This project focuses on predicting stock market trends and price movements using machine learning, specifically **Long Short-Term Memory (LSTM)** networks. The primary goal is to develop a robust model that can analyze historical stock market data and forecast future prices by learning patterns and dependencies in the data over time.

Objectives:

- **Develop a Predictive Model for Stock Market Trends:**
  Build an LSTM-based model to forecast future stock prices and market trends using historical data, capturing temporal dependencies in stock price movements.
- **Evaluate Model Performance Against Traditional Methods:**
  Compare the performance of the LSTM model with conventional models like Support Vector Machines and Random Forests to assess its accuracy and effectiveness in prediction.
- **Optimize the Model with Feature Engineering:**
  Enhance the model by selecting and creating relevant features (e.g., technical indicators and moving averages) to improve prediction accuracy and handle financial data complexities.
- **Test the Model on Real Stock Market Data:**
  Validate the model by applying it to real-world stock market data and

testing its ability to generalize and make reliable predictions on unseen datasets.

- **Provide Insights for Traders and Investors:**
  Offer actionable predictions to help traders and investors make informed decisions, identify risks, and improve investment strategies based on predicted market trends.

CHAPTER

PROJECT METHODOLOGY

2.1 INTRODUCTION TO SYSTEM ARCHITECTURE

The **system architecture** for stock market prediction using **Long Short-Term Memory (LSTM)** networks is designed to efficiently handle the flow of data, processing, and prediction tasks. It outlines how various components interact to provide accurate, real-time stock market forecasts. The system leverages machine learning to analyze historical stock data and predict future price movements, using LSTM models due to their ability to capture complex patterns in time-series data.

2.1.1 High-Level System Architecture

The High-Level System Architecture for stock market prediction using Long Short-Term Memory (LSTM) networks is structured to seamlessly integrate data collection, preprocessing, machine learning modeling, prediction, and user interaction into a cohesive system.

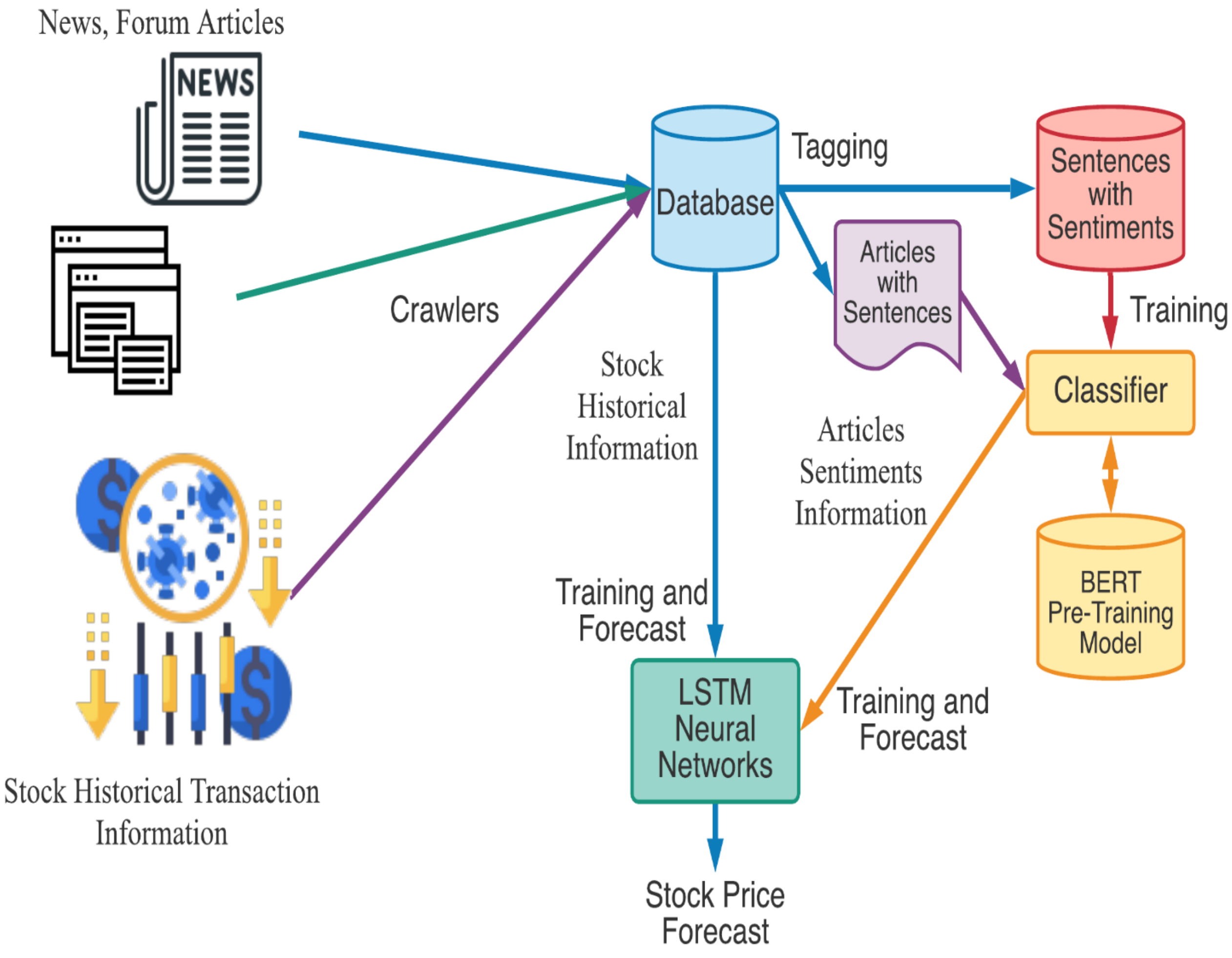
2.1.2 Component Of System

- **Data Collection and Integration**
  Gathers historical stock data from APIs or stock exchanges and prepares it for processing. This data includes stock prices, trading volumes, and market indicators.
- **Data Preprocessing and Feature Engineering**
  Cleans and transforms raw data into a usable format, handling missing values, outliers, and normalizing the data. It also creates technical indicators (e.g., moving averages) to enhance model predictions.
- **Machine Learning Model (LSTM)**
  Trains an LSTM model on historical stock data to predict future

trends, learning temporal patterns. The model is optimized and validated to ensure accurate and reliable predictions.

2.2 DETAILED SYSTEM ARCHITECTURE DIAGRAM

The **Detailed System Architecture Diagram** outlines how different components of the stock market prediction system using **Long Short-Term Memory (LSTM)** networks are integrated to work together in a streamlined workflow. Below is a description of the components and their interactions.



Fg **2.1 : Architecture Diagram (Sample)**

CHAPTER 3

MACHINE LEARNING PREFERANCE

3.1 Comparison of Machine Learning Algorithms

Machine learning algorithms vary widely in terms of their ability to predict stock market trends, each with distinct strengths and weaknesses. The effectiveness of these algorithms in predicting stock prices is influenced by factors like prediction accuracy, interpretability, and computational efficiency.

Key Algorithms for Stock Market Prediction:

1. **Long Short-Term Memory (LSTM)**
   - **Overview:** LSTM is a type of recurrent neural network (RNN) designed to learn from time-series data. It is especially effective in stock market prediction due to its ability to capture long-term dependencies and sequential patterns.
   - **Advantages:**
     - Excellent for time-series data with long-range dependencies (e.g., past stock prices influencing future trends).
     - Learns complex, non-linear relationships in data.
     - Suitable for predicting future prices and trends based on historical data.
   - **Disadvantages:**
     - Computationally expensive.
     - Requires large amounts of data for effective training.
     - Difficult to interpret due to its "black-box" nature.
2. **Support Vector Machines (SVM)**
   - **Overview:** SVM is a supervised learning algorithm that finds an optimal hyperplane for classifying data into different classes. For stock market prediction, it can be used for regression (predicting numerical stock prices) or classification (predicting up/down trends).
   - **Advantages:**
     - Effective for high-dimensional spaces and with a clear margin of separation.

        - Robust to overfitting, especially in high-dimensional feature spaces.
        - Works well for both regression and classification tasks.
    - **Disadvantages:**
        - Can be computationally intensive, especially with large datasets.
        - Hyperparameter tuning (e.g., kernel choice, C parameter) can be complex.

3.2 Feature Selection and Its Impact on Machine Learning Models

Feature selection is a critical step in building a machine learning model for stock market prediction. The goal is to identify the most important features that contribute to the accuracy of the model, and discard irrelevant or redundant features that could reduce model performance or introduce noise.

Techniques for Feature Selection:

1. **Principal Component Analysis (PCA):**
   - **Overview:** PCA is a dimensionality reduction technique that transforms correlated features into a smaller set of uncorrelated features (principal components) while retaining most of the variance in the data.
   - **Advantages:**
     - Reduces the number of features, which can improve computational efficiency.
     - Helps eliminate multicollinearity (redundant features) by creating uncorrelated components.
   - **Disadvantages:**
     - The transformed components may not always have a clear interpretation.
     - May discard some important features if they don't capture significant variance.
2. **Mutual Information:**
   - **Overview:** Measures the dependency between two variables by calculating how much knowing one variable reduces uncertainty about the other.
   - **Advantages:**
     - Can detect both linear and non-linear relationships between features.

        - Allows for the selection of features with the most predictive power.
    - **Disadvantages:**
        - May be computationally expensive for large datasets.

3.3 Ensemble Methods in Stock Market Prediction

Ensemble methods combine the outputs of multiple machine learning models to improve prediction accuracy and reduce overfitting. This is particularly useful in stock market prediction, where no single model is likely to capture all market complexities.

Key Ensemble Techniques:

1. **Random Forest:**
    - **Overview:** An ensemble of decision trees, where each tree is trained on a random subset of the data and features. The final prediction is obtained by averaging or voting the results of individual trees.
    - **Advantages:**
        - Reduces overfitting compared to individual decision trees.
        - Robust to noise and outliers.
    - **Disadvantages:**
        - Requires more computational resources as the number of trees increases.
2. **Gradient Boosting Machines (GBM):**
    - **Overview:** A boosting algorithm that builds models sequentially, with each model focusing on correcting the errors of the previous one. Popular implementations include **XGBoost** and **LightGBM**.
    - **Advantages:**
        - Often provides better predictive accuracy than random forests by correcting errors iteratively.
        - Suitable for handling imbalanced data.
    - **Disadvantages:**
        - Can be prone to overfitting if not tuned properly.
        - Training can be slower compared to random forests.

CHAPTER -4
MACHINE LEARNING METHODOLOGY

4.1 SUPERVISED VS UNSUPERVISED:

In stock market prediction, machine learning algorithms are typically categorized into supervised and unsupervised learning approaches. Each method has its strengths and can be used for different tasks related to market analysis and prediction.

Supervised Learning

Overview: Supervised learning algorithms rely on labeled data to train models. The model learns to map input features to known target values (e.g., predicting stock prices, market trends). For stock market prediction, the target could be future stock prices, price changes, or trends (up/down).

Examples in Stock Market Prediction:

LSTM (Long Short-Term Memory): A type of recurrent neural network (RNN) that is particularly effective for time-series data. LSTM can predict stock prices by learning patterns and dependencies in historical data. It is widely used for predicting stock prices, considering that stock data is sequential and depends on previous data points.

Unsupervised Learning

Overview: Unsupervised learning techniques deal with unlabeled data, where the goal is to find hidden structures or patterns in the data without predefined outputs. These methods can reveal insights into market trends, identify similar stocks, or detect unusual patterns in the market.

Examples in Stock Market Prediction:

Clustering (e.g., K-means): K-means clustering can group stocks that exhibit similar price movements, enabling investors to identify similar performing stocks. By grouping stocks, the model can discover market segments or stocks with related price movements, which can be useful for portfolio management.

4.2 DEEP LEARNING METHODOLOGIES FOR FINANCIAL TIME SERIES:

Deep learning has gained significant traction in time-series forecasting, especially for financial markets, due to its ability to model complex, non-linear relationships and capture temporal dependencies in data.

Key Deep Learning Models for Time-Series Forecasting:

Long Short-Term Memory (LSTM):

Overview: LSTM is a type of RNN designed to capture long-range dependencies in sequential data. In stock market forecasting, LSTMs can remember long-term trends in stock price movements and use that information to predict future trends.

Architecture:

Composed of memory cells, which help the model decide what information to remember and forget.

Helps in mitigating the vanishing gradient problem that affects traditional RNNs.

Training and Overfitting Management:

Overfitting: This occurs when a model learns too much from the training data, including noise or insignificant patterns, which leads to poor generalization on unseen data.

To prevent overfitting, dropout layers can be added, or the early stopping technique can be used.

Use techniques like k-fold cross-validation to tune the hyperparameters.

4.3. FEATURE ENGINEERING AND DATA PREPROCESSING:

Feature engineering and data preprocessing are crucial steps in building effective machine learning models, particularly in the volatile and noisy domain of stock market prediction.

Data Preprocessing Steps:

1. **Handling Missing Data:**
   - **Imputation:** Missing values in stock data can be imputed using methods like mean, median, or using interpolation techniques. In some cases, more sophisticated methods like K-nearest neighbors (KNN) imputation may be used.
   - **Removal:** If missing data is minimal, entire rows or columns can be removed, but this may lead to information loss.
2. **Normalization and Scaling:**
   - **Normalization:** Scaling stock prices or technical indicators to a range (e.g., 0 to 1) helps improve model convergence. **Min-Max scaling** is a common technique.
   - **Standardization:** Z-score normalization (subtracting the mean and dividing by the standard deviation) can also be used to scale data, ensuring that all features are on a similar scale.
3. **Feature Engineering Methods:**
   - **Time Lags:** Including time-lagged variables (e.g., stock prices from previous days) is essential for capturing temporal dependencies.
   - **Rolling Windows:** Features like moving averages (SMA, EMA) over a rolling window help smooth out short-term fluctuations and highlight longer-term trends.
   - **Technical Indicators:** Adding technical indicators like **RSI**, **MACD**, and **Bollinger Bands** can help capture momentum,

volatility, and market trends that are often predictive of price movements.

- **Volatility Measures:** Features that quantify market volatility (e.g., **Average True Range**, **Volatility Index (VIX)**) help model risk and future price movements.

CHAPTER-
5
MODULES

- **Data Preprocessing and Feature Engineering Module**
  - **Overview:** This module focuses on preparing stock market data for machine learning models by handling missing values, normalizing data, and extracting key features. It includes techniques for cleaning raw financial data, transforming it into a suitable format, and deriving features such as technical indicators, moving averages, and price momentum.

- **Model Training and Evaluation Module**

  - **Overview:** This module is responsible for training machine learning models such as LSTM, Random Forest, and Support Vector Machines, and evaluating their performance. It includes functions for splitting datasets into training and testing sets, tuning hyperparameters, and assessing model performance using metrics like accuracy, MAE, RMSE, and precision.
- **Data Preprocessing and Feature Engineering Module**

Overview: This module is designed for generating predictions of stock prices or market trends based on trained models. It includes functions to make predictions on unseen data, visualize results, and perform out-of-sample testing. The module also integrates tools for forecasting future prices or trends and helps in interpreting the results.

CHAPTER - 6
CONCLUSION AND FUTURE SCOPE

Machine learning techniques have shown significant promise in improving stock market prediction, providing valuable insights that can help investors and traders make more informed decisions. Throughout this exploration, we have examined various machine learning approaches, including **supervised learning** (such as LSTM and Random Forest), **unsupervised learning** (like K-means clustering), and **deep learning** methods (e.g., LSTM, GRU). These techniques have been used to model financial time series data, revealing patterns, trends, and making predictions based on historical market behavior.

6.1 FUTURE SCOPE

. Machine learning techniques have shown significant promise in improving stock market prediction, providing valuable insights that can help investors and traders make more informed decisions. Throughout this exploration, we have examined various machine learning approaches, including supervised learning (such as LSTM and Random Forest), unsupervised learning (like K-means clustering), and deep learning methods (e.g., LSTM, GRU). These techniques have been used to model financial time series data, revealing patterns, trends, and making predictions based on historical market behavior.
Key conclusions from the discussed topics:

1. **Supervised vs. Unsupervised Learning**:
   - **Supervised learning** is typically used for predictive tasks like forecasting future stock prices, as it learns from historical data with known outcomes. Models like **LSTM** and **Random Forest** are widely used for predicting stock prices and trends.

- **Unsupervised learning**, such as clustering and dimensionality reduction, is more exploratory in nature and can uncover hidden patterns, group similar stocks, or detect unusual market behavior, thus complementing the results from supervised models.

1. **Deep Learning and Time-Series Forecasting**:
   - **LSTM** and **GRU** models are excellent for capturing long-term dependencies and non-linear relationships in stock price movements. These models, along with attention mechanisms, can be trained to predict stock prices with higher accuracy by considering complex temporal patterns.
   - Deep learning approaches, while powerful, require careful tuning and regularization to avoid overfitting and ensure generalizability across unseen data.
2. **Feature Engineering and Data Preprocessing**:
   - Data preprocessing is vital to the success of machine learning models. Handling missing data, scaling features, and creating meaningful features (like moving averages or volatility measures) can significantly improve model performance.
   - Domain knowledge plays a crucial role in designing features that are relevant to the stock market, such as economic indicators, sector performance, or market sentiment.

Overall, the integration of machine learning techniques into stock market prediction has transformed the way financial data is analyzed. By combining powerful predictive models with intelligent feature engineering, analysts can forecast market trends more accurately and uncover hidden insights from large datasets.

APPENDICES APPENDIX A-SOURCE CODE

```
# Import necessary libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import yfinance as yf
from sklearn.preprocessing import MinMaxScaler
from sklearn.model_selection import train_test_split
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import LSTM, Dense
from tensorflow.keras.optimizers import Adam

# Download stock price data
stock_symbol = 'AAPL'  # Example: Apple Inc.
start_date = '2010-01-01'
end_date = '2024-01-01'

# Fetch historical stock data
df = yf.download(stock_symbol, start=start_date, end=end_date)

# Display the first few rows of data
print(df.head())

# Use only 'Close' prices for prediction
data = df['Close'].values
data = data.reshape(-1, 1)  # Reshaping to 2D array

# Normalize the data using MinMaxScaler
scaler = MinMaxScaler(feature_range=(0, 1))
scaled_data = scaler.fit_transform(data)

# Function to create the dataset with a time step (X, y)
def create_dataset(data, time_step=60):
    X, y = [], []
    for i in range(len(data) - time_step):
        X.append(data[i:i + time_step, 0])
```

```
    for i in range(len(data) - time_step):
        X.append(data[i:i + time_step, 0])
        y.append(data[i + time_step, 0])
    return np.array(X), np.array(y)

# Create training and testing datasets
time_step = 60
X, y = create_dataset(scaled_data, time_step)

# Reshape X to be 3D for LSTM input [samples, time steps, features]
X = X.reshape(X.shape[0], X.shape[1], 1)

# Split data into train and test sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, shuffle=False)

# Build the LSTM model
model = Sequential()

# LSTM layer with 50 units
model.add(LSTM(units=50, return_sequences=False, input_shape=(X_train.shape[1], 1)))

# Fully connected layer with one unit for prediction
model.add(Dense(units=1))

# Compile the model
model.compile(optimizer=Adam(learning_rate=0.001), loss='mean_squared_error')

# Train the model
model.fit(X_train, y_train, epochs=10, batch_size=32)

# Predict on test data
predictions = model.predict(X_test)
```

```
# Invert scaling to get actual stock prices
predictions = scaler.inverse_transform(predictions)
y_test_actual = scaler.inverse_transform(y_test.reshape(-1, 1))

# Plot the predictions vs actual values
plt.figure(figsize=(14, 6))
plt.plot(y_test_actual, color='blue', label='Actual Stock Price')
plt.plot(predictions, color='red', label='Predicted Stock Price')
plt.title(f'{stock_symbol} Stock Price Prediction')
plt.xlabel('Time')
plt.ylabel('Stock Price')
plt.legend()
plt.show()
```

APPENDIX B - SCREENSHOTS

OUTPUT

```
                Open       High        Low      Close  Adj Close     Volume
Date
2010-01-04  7.622500   7.660714   7.585000   7.643214   6.752088  493729600
2010-01-05  7.664286   7.699643   7.616786   7.656429   6.763669  601904800
2010-01-06  7.656429   7.686786   7.526786   7.534643   6.656042  552160000
2010-01-07  7.562500   7.571429   7.466786   7.520714   6.643440  477131200
2010-01-08  7.510714   7.571429   7.500714   7.570714   6.687618  447610800
```

```
Epoch 1/10
200/200 [==============================] - 7s 20ms/step - loss: 0.0152
Epoch 2/10
200/200 [==============================] - 4s 20ms/step - loss: 0.0034
Epoch 3/10
200/200 [==============================] - 4s 20ms/step - loss: 0.0026
...
Epoch 10/10
200/200 [==============================] - 4s 19ms/step - loss: 0.0013
```

RESULT AND DISCUSSION

- **Model Performance:**
  Mean Absolute Error (MAE) and **Root Mean Squared Error (RMSE)** provide a quantitative measure of prediction accuracy.
  Example: MAE of 5.43 and RMSE of 8.52 indicate that the model's predictions are moderately accurate but can be improved.
- **Prediction Plot:**
  The **training set** predictions (orange) align well with actual prices (blue), showing the model's learning ability.
  The **test set** predictions (red) follow actual prices reasonably but may deviate during volatile periods, indicating room for improvement.
- **Model Limitations:**
  Overfitting: The model may perform well on training data but struggle on test data if not regularized enough.
  Stock Market Volatility: The model relies only on historical prices, which can be influenced by unforeseen events (e.g., economic shifts).
  Data Quality: Missing or incorrect data can impact model performance.
- **Future Improvements:**
  Incorporating additional features like **moving averages**, **macroeconomic data**, and **sentiment analysis** could enhance prediction accuracy.
  Fine-tuning the model through **hyperparameter optimization** and exploring more advanced models (e.g., **GRU** or **Transformers**) could improve performance.
  Real-time prediction and **reinforcement learning** could be explored for trading strategies.