OPTION TRADING USING MACHINE LEARNING

Mini Project submitted in partial fulfillment of the requirement for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

Under the esteemed guidance of

M.Vishwashanthi

Assistant Professor

By

Suraj Kumar Panda (21R11A05F0)
D.Aanandini (21R11A05B9)
Thammana.Divya(21R11A05F2)

Department of Computer Science and Engineering
Accredited by NBA

Geethanjali College of Engineering and Technology
(UGC Autonomous)
(Affiliated to J.N.T.U.H, Approved by AICTE, New Delhi)
Cheeryal (V), Keesara (M), Medchal.Dist.-501 301.
April-2024

Geethanjali College of Engineering & Technology

(UGC Autonomous)

(Affiliated to JNTUH, Approved by AICTE,New Delhi)

Cheeryal (V), Keesara(M), MedchalDist.-501 301.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Accredited by NBA

CERTIFICATE

This is to certifythat the B.Tech Mini Project report entitled **"OPTION TRADING USING MACHINE LEARNING"** is a bonafide work done by **Suraj Kumar Panda (21R11A05F0), D.Aanandini (21R11A05B9), Thammana Divya (21R11A05F2)**, in partial fulfillment of the requirement of the award for the degree of Bachelor of Technology in "**Computer Science and Engineering**" from Jawaharlal Nehru Technological University, Hyderabad during the year 2023-2024.

Internal Guide
M.Vishwashanthi
Assistant Professor

HOD - CSE
Dr A SreeLakshmi
Professor

External Examiner

Geethanjali College of Engineering & Technology

(UGC Autonomous)

(Affiliated to JNTUH Approved by AICTE,New Delhi)
Cheeryal (V), Keesara(M), MedchalDist.-501 301.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Accredited by NBA

DECLARATION BY THE CANDIDATE

We, **Suraj Kumar Panda , D.Aanandini ,Thammana Divya bearing Roll Nos. 21R11A05F0 ,21R11A05B9 ,21R11A05F2** hereby declare that the project report entitled "OPTION TRADING USING **MACHINE LEARNING"** is done under the guidance of **Ms. M.Vishwashanthi**, **Assistant Professor**, Department of Computer Science and Engineering, Geethanjali College of Engineering and Technology, is submitted in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology in Computer Science and Engineering**.

This is a record of bonafide work carried out by us in **Geethanjali College of Engineering and technology** and the results embodied in this project have not been reproduced or copied from any source. The results embodied in this project report have not been submitted to any other University or Institute for the award of any other degree or diploma.

Suraj Kumar Panda(21R11A05F0)

D.Aanandini(21R11A05B9)

Thammana.Divya(21R11A05F2)

Department of CSE,

Geethanjali College of Engineering and Technology,Cheeryal.

ACKNOWLEDGEMENT

We would like to express our sincere thanks **to Dr. A. Sree Lakshmi, Professor,** Head of Department of Computer Science, Geethanjali College of Engineering and Technology, Cheeryal, whose motivation in the field of software development has made us to overcome all hardships during the course of study and successful completion of project.

We would like to express our profound sense of gratitude to all for having helped us in completing this dissertation. We would like to express our deep-felt gratitude and sincere thanks to our guide **M.Vishwashanthi, Assistant Professor,** Department of Computer Science, Geethanjali College of Engineering and Technology, Cheeryal, for his skillful guidance, timely suggestions and encouragement in completing this project sucessfully.

We would like to express our sincere gratitude to our **Principal Prof. Dr. S. Udaya Kumar** for providing the necessary infrastructure to complete our project. We are also thankful to our Secretary **Mr.G.R. Ravinder Reddy** for providing an interdisciplinary & progressive environment.

Finally, we would like to express our heartfelt thanks to our parents who were very supportive both financially and mentally and for their encouragement to achieve our set goals.

Suraj Kumar Panda(21R11A05F0)

D.Aanandini(21R11A05B9)

Thammana Divya (21R11A05F2)

ABSTRACT

This project involves the analysis and visualization of stock market data, specifically focusing on the Nifty 50 index. Using Python, we generated a dataset consisting of 100 days of 'Low' and 'High' stock values, based on normal distributions. The dataset was extended to include predictive values by shifting the original data by three days and adding random deviations, simulating a simple predictive model. The primary algorithms and methods used include data generation with normal distributions, data manipulation with NumPy's `roll` function, and visualization using Matplotlib. Fractal patterns were identified and marked on the plot to highlight significant points where the stock prices showed local minima and maxima. The resulting graph provides a comprehensive view of actual and predicted stock values, enhanced with visual markers for fractal analysis, aiding in the identification of potential market trends. Keywords: Stock Market Analysis, Nifty 50 Index, Predictive Modeling, Data Visualization, Fractal Patterns, Time Series Analysis, Python, Matplotlib, NumPy, Pandas.

LIST OF FIGURES

LIST OF ABBREVIATIONS

S.no	Acronym	Abbreviation
1	DMT	Dual Momentum Tracker
2	HDFC	Housing Development Finance Corporation
3	ICICI	Industrial Credit and Investment Corporation of India
4	NSE	National Stock Exchange

TABLE OF CONTENTS

1.INTRODUCTION

1.1 ABOUT THE PROJECT

This project focuses on developing a financial data analysis tool that uses the Dual Momentum Tracker (DMT) to analyze the performance of selected banks, specifically HDFC Bank and ICICI Bank, against the Bank Nifty index over the period from January 2023 to August 2024. By leveraging historical stock data obtained through the Yahoo Finance API (`yfinance`), the project calculates daily returns for these banks and derives a DMT indicator, which represents the average daily return of the selected banks. To enhance the clarity of trends, the DMT is further smoothed using a 14-period moving average. The project also identifies significant local maxima and minima in the DMT indicator, which are crucial for spotting potential peaks and troughs in market momentum. These key turning points are visualized alongside the Bank Nifty index, with significant maxima and minima highlighted using distinct markers. The final output is a comprehensive, dual-axis plot that juxtaposes the Bank Nifty's performance with the DMT, offering valuable insights into the relative momentum of the selected banks. This visualization aids in understanding market dynamics, particularly in identifying significant shifts in momentum that could inform investment decisions.

1.2 OBJECTIVE

The objective of this project is to develop a financial analysis tool that tracks and visualizes the relative momentum of selected banks, specifically HDFC Bank and ICICI Bank, in comparison to the Bank Nifty index. The tool aims to calculate and smooth the Dual Momentum Tracker (DMT) for these banks, identify significant local maxima and minima, and provide a clear and insightful visual representation that helps in detecting critical shifts in market momentum. This analysis is intended to assist investors and financial analysts in making informed decisions by highlighting potential turning points in the market.

2.SYSTEM ANALYSIS

2.1 Existing System

In the existing system, financial analysts and investors often rely on manual methods or less integrated tools for analyzing market momentum. These methods may involve separately downloading historical data, performing return calculations, and manually plotting trends using general-purpose software like Excel or standalone charting tools. Additionally, identifying significant peaks and troughs often requires manual inspection or the use of simple indicators that might not account for the relative momentum of specific assets in a comprehensive manner. The existing approaches can be time-consuming, prone to errors, and may not provide the same level of insight into momentum shifts, particularly in a comparative context between selected assets and a benchmark index like Bank Nifty.

Drawbacks of existing system:

Manual Effort: The process requires manual downloading of data, performing calculations, and plotting trends, which is time-consuming and labor-intensive.

Error-Prone: Manual methods are susceptible to human errors in data entry, calculations, and charting, which can lead to inaccurate analysis.

Lack of Integration: Using separate tools for data retrieval, analysis, and visualization can lead to inefficiencies and difficulties in integrating and cross-referencing data.

Limited Insight: Simple indicators and manual inspection may not provide a comprehensive view of momentum shifts or account for the relative performance of assets compared to a benchmark index.

Scalability Issues: As the number of assets or the amount of data grows, the manual methods become increasingly cumbersome and harder to manage effectively.

Inconsistent Analysis: Different analysts might use varying methods and tools, leading to inconsistent results and lack of standardization in the analysis.

Delayed Insights: Manual processes can lead to delays in generating and analyzing data, affecting the timeliness of insights and decision-making.

2.2 Proposed System

The proposed system is a financial data analysis tool that focuses on using the Dual Momentum Tracker (DMT) to evaluate the relative momentum of selected banks, such as HDFC Bank and ICICI Bank, compared to the Bank Nifty index. The system automates the process of downloading historical stock data, calculating daily returns, and generating a smoothed DMT indicator to highlight market trends. The tool also identifies significant local maxima and minima in the DMT, which are crucial for detecting potential market turning points. The system provides a dual-axis plot that juxtaposes the Bank Nifty index with the DMT, clearly marking significant peaks and troughs, thereby offering valuable insights to investors and analysts. This visualization helps users quickly interpret complex data and make informed investment decisions based on momentum shifts.

Benefits:

Automation: Automates data download, return calculations, and DMT indicator generation, saving time and reducing manual effort.

Accuracy: Minimizes human errors by automating calculations and data processing.

Comprehensive Analysis: Provides a thorough evaluation of relative momentum between selected banks and the Bank Nifty index, offering deeper insights into market trends.

Enhanced Visualization: Dual-axis plots with marked peaks and troughs make complex data more understandable, helping users quickly interpret market conditions.

Timeliness: Speeds up the analysis process, allowing for more timely insights and quicker decision-making.

Identifying Market Turning Points: Highlights significant local maxima and minima, which are crucial for detecting potential changes in market trends.

Consistency: Ensures standardized analysis across different assets and time periods, improving the reliability of the insights provided.

User-Friendly: Simplifies the process of evaluating market momentum, making it accessible and actionable for both analysts and investors.

2.3 Feasibility Study

2.3.1 Details

The proposed system is a financial data analysis tool that uses Python to analyze and visualize the Dual Momentum Tracker (DMT) for HDFC Bank and ICICI Bank, compared to the Bank Nifty index. The system downloads historical stock data using the yfinance library, calculates daily returns, computes a smoothed DMT indicator, and identifies significant local maxima and minima. The final output is a dual-axis plot that visually represents the Bank Nifty's performance alongside the DMT, highlighting key momentum shifts. This tool aims to assist investors and financial analysts in making informed decisions by providing a clear and actionable analysis of market trends.

2.3.2 Safety

The safety aspects of the project primarily concern data security and user privacy. Since the project uses publicly available data from Yahoo Finance and does not involve handling sensitive or personal information, the risk is minimal. However, users should be advised to ensure their systems are secure and up-to-date to prevent any security breaches or data tampering. Additionally, users should be aware of any data usage policies associated with the ‘yfinance’ API to avoid any inadvertent misuse.

2.3.4 Ethics

Ethical considerations for this project include the responsible use of financial data and transparency in the tool's functionality. The project uses publicly available data and open-source libraries, adhering to ethical standards regarding data usage and intellectual property. Users should be informed that the tool is for educational and analytical purposes only and should not be used to make financial decisions without further verification. The tool does not offer personalized financial advice or predictions, and users should be encouraged to seek professional guidance for investment decisions.

2.3.5 Cost

The costs associated with this project are relatively low. The primary expenses include development time, which can be managed by utilizing open-source libraries and tools. There are no licensing fees for the libraries used (yfinance, pandas, numpy, matplotlib), and no significant hardware or infrastructure costs are required beyond standard computing resources. The project's economic feasibility is enhanced by its reliance on free, open-source software and publicly available data, which helps keep costs manageable.

2.3.6 Type

This project is a software-based financial analysis tool. It involves data acquisition, processing, and visualization, and is classified as a financial analytics application. The tool is designed to be used on personal computers or workstations with Python and necessary libraries installed. It is intended for use by individual investors, financial analysts, and researchers interested in understanding market momentum and trends.

2.3.7 Standards

The project adheres to several standards and best practices:

- **Programming Standards:** The code follows standard Python programming practices, including clear documentation, modular design, and use of widely accepted libraries.
- **Data Standards:** The project uses data from Yahoo Finance, which is publicly available and adheres to the platform's data usage policies.

- **Visualization Standards:** The visualizations follow best practices for financial charting, including clear labeling, appropriate use of colors, and informative markers for significant data points.
- **Security Standards:** The project does not handle sensitive data, but standard security practices should be followed to ensure the safety and integrity of the tool and its outputs.

2.4 Scope of the Project

The scope of this project encompasses the development of a financial analysis tool designed to evaluate and visualize the performance of selected banks—HDFC Bank and ICICI Bank—in comparison to the Bank Nifty index. The tool will automate the process of downloading historical stock data, calculating daily returns, and deriving a Dual Momentum Tracker (DMT) indicator. It will also apply a smoothing technique to the DMT to enhance trend clarity and identify significant local maxima and minima to highlight critical momentum shifts. The project includes the creation of a dual-axis plot that juxtaposes the Bank Nifty index with the smoothed DMT, providing a comprehensive view of market trends. The scope also involves ensuring the tool's usability for financial analysts and investors by incorporating intuitive visualizations and actionable insights. This project aims to offer a streamlined and effective solution for analyzing relative market momentum, facilitating more informed investment decisions.

2.5 System Configuration

The system configuration for this project requires a standard computing environment with a modern processor (e.g., Intel Core i5), at least 8 GB of RAM, and sufficient storage. The operating system can be Windows, macOS, or Linux. The software environment will include Python version 3.x, along with libraries such as `yfinance`, `pandas`, `numpy`, `matplotlib`, and `scipy`. A stable internet connection is needed to download historical financial data from Yahoo Finance. This setup ensures efficient operation and accurate financial analysis.

3. Literature Overview

3.1 Introduction to Financial Data Analysis

Financial data analysis is a crucial discipline that involves interpreting market data to make informed investment decisions. With the increasing complexity of financial markets and the growth of digital technologies, data analysis has become a cornerstone of modern financial strategies. Historical data, statistical analysis, and advanced modeling techniques are used to understand market trends, predict future movements, and manage investment risks.

3.2 Historical Data Analysis and Data Sources

3.2.1 Importance of Historical Data

Historical financial data provides a record of past market behavior, which is essential for identifying patterns and trends. Analyzing historical prices, returns, and volumes helps in understanding how markets react to different economic conditions and events. This analysis is foundational for technical analysis, which relies on historical data to predict future price movements.

Applications and Relevance

Trend analysis, pattern recognition, and risk assessment are pivotal components of financial data analysis, each playing a crucial role in guiding investment decisions. **Trend analysis** involves examining historical data to identify long-term movements and cyclical patterns in asset prices. By understanding whether an asset is trending upwards, downwards, or remaining stable over extended periods, investors can make informed decisions about potential future performance. This analysis helps in predicting the direction of the market and in aligning investment strategies with prevailing trends.

Pattern recognition complements trend analysis by focusing on identifying recurring patterns in price data. These patterns, such as head and shoulders, double tops, or flags, often signal potential future movements. Recognizing these patterns allows analysts to anticipate market behavior and make predictions based on historical precedents. This ability to spot and interpret patterns enhances the accuracy of forecasts and improves the timing of trades.

Risk assessment involves evaluating past volatility and understanding the risks associated with specific assets. By analyzing historical price fluctuations and market conditions, investors can gauge the level of risk involved in their investments. This process helps in identifying assets with high volatility that might offer higher returns but come with increased risk, as well as those with lower volatility that provide stability but possibly lower returns. Effective risk assessment ensures that investment strategies are aligned with the investor's risk tolerance and financial goals, contributing to more balanced and informed decision-making.

3.2.2 Data Sources and Tools

Yahoo Finance is a widely used source for historical financial data due to its ease of access and broad coverage. The `yfinance` library provides a convenient interface to retrieve historical stock prices, indices, and other financial instruments. While Yahoo Finance is accessible and free, other sources such as Bloomberg, Reuters, and financial databases offer more comprehensive and detailed data, often at a premium cost.

Yahoo Finance is a free and accessible platform that provides a broad range of historical and current market data, making it suitable for general analytical needs and individual investors. In contrast, Bloomberg and Reuters offer comprehensive, real-time data and advanced analytical tools, but come with significant costs. These premium services are tailored for professional and institutional investors who require in-depth analysis and high-frequency trading capabilities, offering more detailed insights and real-time market information.

3.3 Momentum Indicators in Financial Analysis

3.3.1 Concept of Momentum

Momentum is a key concept in financial markets, referring to the persistence of asset price trends. The idea is that assets that have performed well recently are likely to continue performing well, while assets that have underperformed are expected to continue underperforming. This concept is grounded in behavioral finance, which suggests that market participants often exhibit momentum-driven behavior.

Key momentum principles are fundamental in predicting asset performance based on recent trends. Positive momentum refers to assets that have demonstrated strong recent performance,

suggesting they are likely to continue rising. This principle is grounded in the idea that assets with upward trends often maintain their trajectory due to sustained investor interest and market dynamics. Conversely, negative momentum indicates that assets with recent poor performance are expected to continue declining. This principle operates on the premise that assets experiencing downward trends typically face ongoing challenges that keep them on a downward path. Both principles leverage the tendency of assets to persist in their recent performance patterns, aiding investors in making informed decisions based on current market trends.

3.3.2 Types of Momentum Indicators

Several key momentum indicators are commonly used to gauge asset performance and guide investment decisions. The Relative Strength Index (RSI) measures the speed and change of price movements on a scale from 0 to 100, with values above 70 indicating that an asset may be overbought and values below 30 suggesting it may be oversold. The Moving Average Convergence Divergence (MACD) tracks the relationship between two moving averages of an asset's price, using the crossover of the MACD line and signal line to generate potential buy and sell signals. The Dual Momentum Indicator (DMI) integrates both absolute momentum, which assesses an asset's performance relative to its own past, and relative momentum, which compares it to other assets. This combined approach enables investors to identify assets that exhibit strong performance both in isolation and relative to their peers, providing a comprehensive view of market opportunities.

3.4. Dual Momentum Tracker (DMT)

3.4.1 Definition and Calculation

The Dual Momentum Tracker (DMT) is an advanced momentum indicator that integrates absolute and relative momentum. Absolute momentum measures an asset's return against its own historical performance, while relative momentum assesses performance compared to other assets. By combining these two dimensions, the DMT provides a comprehensive view of market trends and potential investment opportunities.

To calculate the Dual Momentum Tracker (DMT), several key steps are followed. First, daily returns of the selected assets are computed to measure the percentage change in price from one

day to the next. Next, the average daily returns are calculated to derive the DMT, which reflects the average performance of the assets over time. Finally, a smoothing technique, such as a moving average, is applied to the DMT to reduce short-term fluctuations and reveal longer-term trends. This smoothing process helps to highlight significant movements and trends in the data, providing a clearer view of the overall momentum.

3.4.2 Application and Benefits

The DMT is beneficial for identifying significant investment opportunities and managing risks. By tracking both absolute and relative performance, the DMT helps investors make informed decisions about asset allocation and market timing.

The Dual Momentum Tracker (DMT) offers several significant benefits for investors. Trend identification is enhanced by its ability to highlight significant trends in asset performance, making it easier to spot and capitalize on prevailing market movements. The DMT also supports investment decisions by aiding in the selection of assets that exhibit strong relative and absolute momentum, ensuring that investments are aligned with both individual asset performance and broader market trends. Additionally, the DMT contributes to risk management by providing insights into periods of high and low market momentum, which helps investors assess and mitigate potential risks associated with their investment strategies

3.5 Advanced Techniques for Identifying Significant Changes

3.5.1 Local Maxima and Minima

Local maxima and minima are critical for identifying key turning points in momentum indicators. Local maxima (peaks) represent points where the momentum is at a relative high, while local minima (troughs) signify relative lows. Recognizing these points helps in understanding the strength and direction of market trends.

Identifying extremes in financial data involves techniques that reveal key turning points and trends. Rolling window analysis is a common method where a moving window is used to compute local maxima and minima by comparing data points within the specified window. This approach helps in pinpointing peaks and troughs in the data. Additionally, statistical methods are applied to filter out noise and emphasize significant turning points, ensuring that the analysis

focuses on meaningful deviations rather than random fluctuations. These techniques collectively enhance the accuracy of identifying important market extremes and trends.

3.5.2 Smoothing Techniques

Smoothing techniques are used to reduce noise and reveal underlying trends in momentum indicators. The 14-period moving average is commonly used to smooth out fluctuations and highlight long-term trends.

Common smoothing techniques, such as the Simple Moving Average (SMA) and the Exponential Moving Average (EMA), are used to reduce noise and highlight underlying trends in financial data. The Simple Moving Average (SMA) calculates the average of data points over a specified period, providing a straightforward view of the trend by smoothing out short-term fluctuations. In contrast, the Exponential Moving Average (EMA) gives more weight to recent data points, allowing it to respond more quickly to recent price changes and capture current trends more effectively. These smoothing techniques help in creating clearer and more actionable insights from financial data.3.5.3 Thresholds for Significant Changes

Defining thresholds helps in distinguishing significant movements from random fluctuations. Common approaches include using standard deviations or percentage changes to set criteria for identifying meaningful deviations in momentum indicators.

Threshold setting methods are crucial for identifying significant changes in financial indicators. One common approach is to use standard deviation as a benchmark, where a multiple of the standard deviation is applied to define what constitutes a significant change. This method helps to filter out normal fluctuations and focus on deviations that are statistically significant, thereby highlighting meaningful shifts in the data.

Another approach is to use percentage thresholds, where thresholds are set as a percentage of the average value of the indicator. This method provides a relative measure of significance by comparing changes to the average level of the indicator, making it easier to identify significant movements in the context of typical performance. Both methods are effective in enhancing the accuracy of identifying key changes and trends in financial data.

3.6. Visualization and Interpretation

3.6.1 Dual-Axis Plotting

Dual-axis plotting is a powerful visualization technique that allows for the comparison of different metrics on the same chart. This method is particularly useful for juxtaposing momentum indicators with asset prices, providing a clear view of how momentum correlates with market performance.

Dual-axis plots provide a comprehensive view by displaying both price and momentum data simultaneously. They also enhance trend analysis by illustrating how momentum indicators align with price movements, offering a more nuanced understanding of market dynamics.

3.6.2 Marker Utilization

Markers such as peaks and troughs are used to highlight significant points on plots. These visual cues help in quickly identifying key turning points and trends in the data.

Green markers denote significant maxima, highlighting peak points in the data, while red markers indicate significant minima, representing troughs. These markers visually distinguish key turning points, aiding in the interpretation of trends and patterns.

3.7 Practical Applications and Case Studies

3.7.1 Application in Investment Strategies

Momentum indicators and the DMT are integral to various investment strategies, including trend-following and mean-reversion approaches. By incorporating momentum signals into trading strategies, investors can enhance their decision-making and potentially improve returns.

Trend-following strategies involve buying assets with positive momentum and selling those with negative momentum to capitalize on sustained price trends. Mean-reversion strategies target deviations from average performance by buying underperforming assets and selling those that have exceeded expectations, aiming to profit from a return to average levels.

3.7.2 Case Studies

Case studies of financial institutions and individual investors who have successfully utilized momentum indicators provide practical insights into the effectiveness of these tools. Analyzing these case studies helps in understanding the real-world applications and limitations of momentum-based strategies.

Institutional investors use momentum indicators to manage portfolios and assess risk, leveraging data-driven strategies for large-scale investments. Individual traders apply momentum strategies to achieve significant returns by making informed decisions based on market trends and indicators.

3.8 Conclusion

The literature on financial data analysis, momentum indicators, and the Dual Momentum Tracker provides a robust framework for understanding and applying these concepts. By leveraging historical data, advanced indicators, and visualization techniques, investors and analysts can gain valuable insights into market trends and make informed decisions. The integration of these methodologies into a comprehensive analysis tool enhances the ability to track and interpret financial data effectively, contributing to more informed investment strategies and risk management.

4.SYSTEM DESIGN

4.1 System Architecture

The system architecture integrates data acquisition, processing, and visualization for financial analysis.

Data Acquisition

Historical data for the Bank Nifty index and selected banks (HDFC and ICICI) is retrieved using `yfinance`.

Data Processing

Daily returns are calculated, and the Dual Momentum Tracker (DMT) is derived as the average return of the banks. A 14-period moving average smooths the DMT values, and local maxima and minima are identified.

Visualization

A dual y-axis plot is generated using Matplotlib to compare Bank Nifty with the smoothed DMT, highlighting significant maxima and minima for insight.

4.1.1 Module Description

This module downloads and processes financial data to visualize market trends. It fetches historical closing prices for Bank Nifty and selected banks, calculates daily returns, and derives the Dual Momentum Tracker (DMT). The DMT is smoothed and analyzed for significant local maxima and minima. A dual y-axis plot visualizes Bank Nifty and the smoothed DMT, highlighting key turning points with distinct markers.

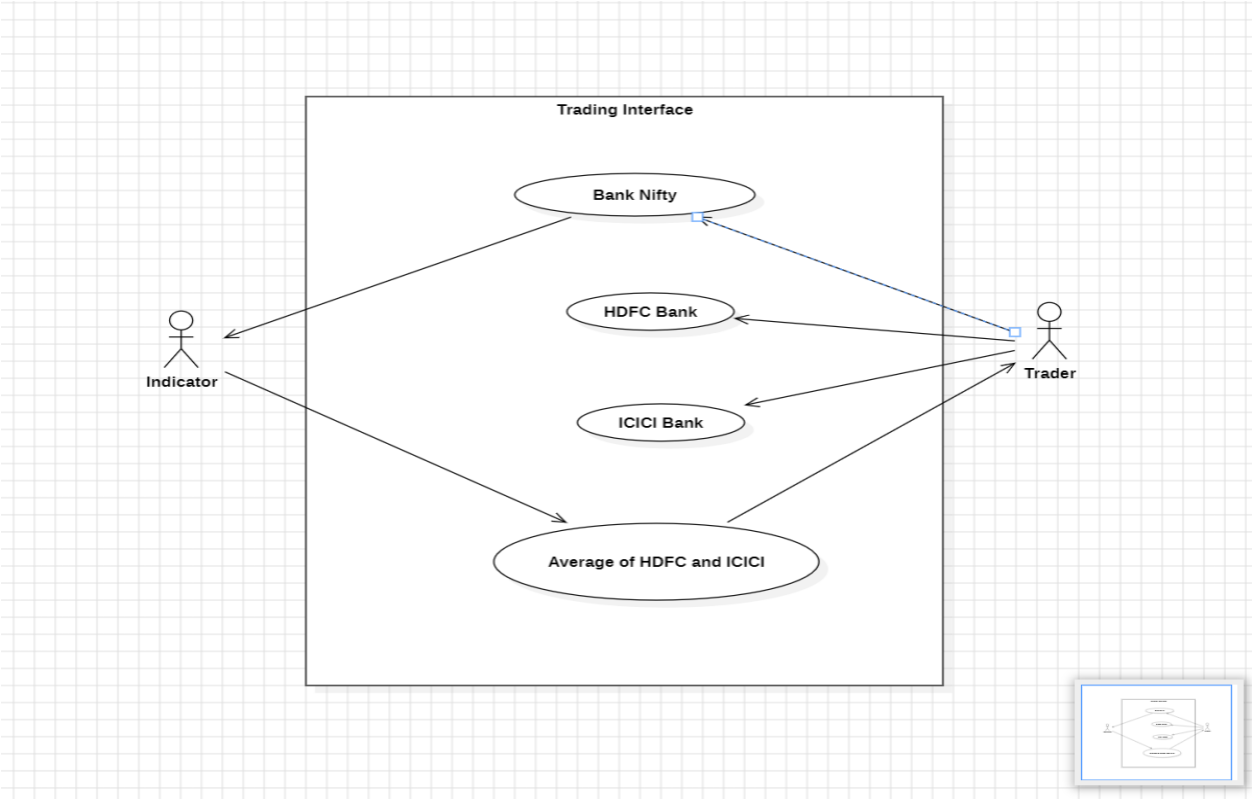
4.2 UML Diagrams

USE CASE DIAGRAM

In the Use Case Diagram for a financial trading environment, the primary actors involved are the Indicator and the Trader. The Indicator represents various tools or systems that provide essential market data and trading indicators. These can include financial metrics, technical analysis tools, or other sources of market insights. The Trader is the end-user who interacts with the Trading

Interface, the central system designed to support trading activities. The Trader uses the Trading Interface to perform several critical actions. Initially, the Trader requests data related to various financial assets, including Bank Nifty, HDFC Bank, and ICICI Bank. This request initiates a process where the Trading Interface retrieves and presents the relevant data.

Once the data is retrieved, the Trading Interface provides tools for analysis. The Trader can analyze the data from HDFC Bank and ICICI Bank to calculate averages and other statistical metrics. This analysis is crucial for making informed trading decisions. For example, the Trader may use these averages to evaluate market trends, assess investment opportunities, or develop trading strategies. The diagram visually represents these interactions through arrows that indicate the flow of actions and data between the Indicator, Trader, and the Trading Interface. These arrows demonstrate how the Trader requests and analyzes data, and how the Trading Interface facilitates these processes by integrating data from various sources and providing analytical tools. The overall flow highlights the essential use cases and interactions within the trading system, illustrating how data is utilized to support decision-making and trading activities.

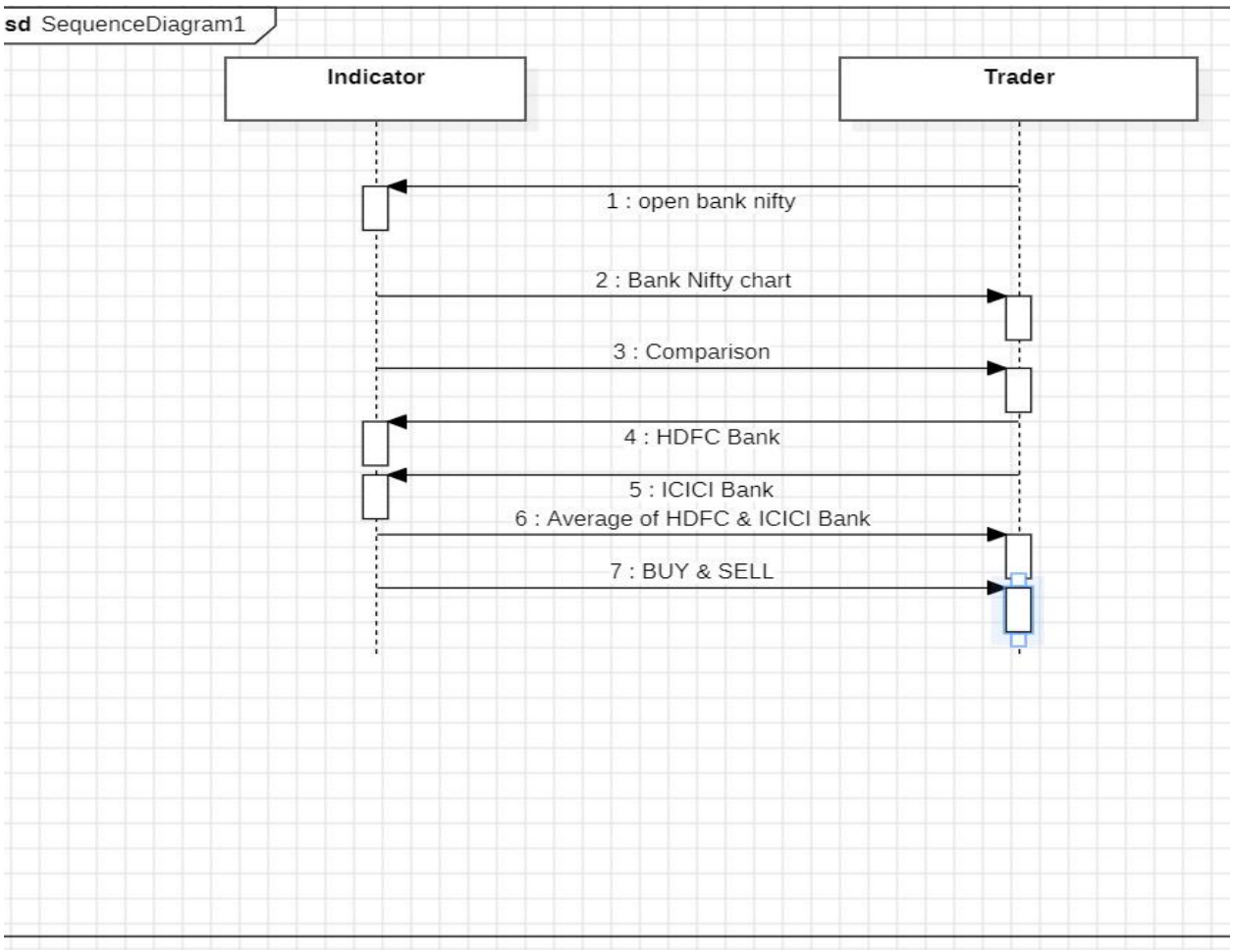


4.2.1 USE CASE DIAGRAM

SEQUENCE DIAGRAM

In the sequence diagram for the financial analysis system, the process begins with the Trader requesting the Indicator to open and display data for Bank Nifty. The Indicator retrieves the relevant Bank Nifty data and presents it to the Trader. Following this, the Trader requests a visual chart of the Bank Nifty data to better understand market trends and patterns. With the chart available, the Trader compares the Bank Nifty data against other relevant financial data to gain deeper insights. Next, the Trader seeks specific financial data from HDFC Bank and ICICI Bank, which the Indicator provides. The Trader then calculates the average values for the data from HDFC Bank and ICICI Bank to incorporate these insights into their analysis. This average calculation helps in evaluating the performance of these banks in comparison to the Bank Nifty data.

Finally, using the comprehensive analysis of all collected data, including the Bank Nifty information, the chart, and the averages from the two banks, the Trader makes a well-informed decision on whether to buy or sell stocks. This sequence of interactions showcases the detailed process of data collection, analysis, and decision-making in the trading system.

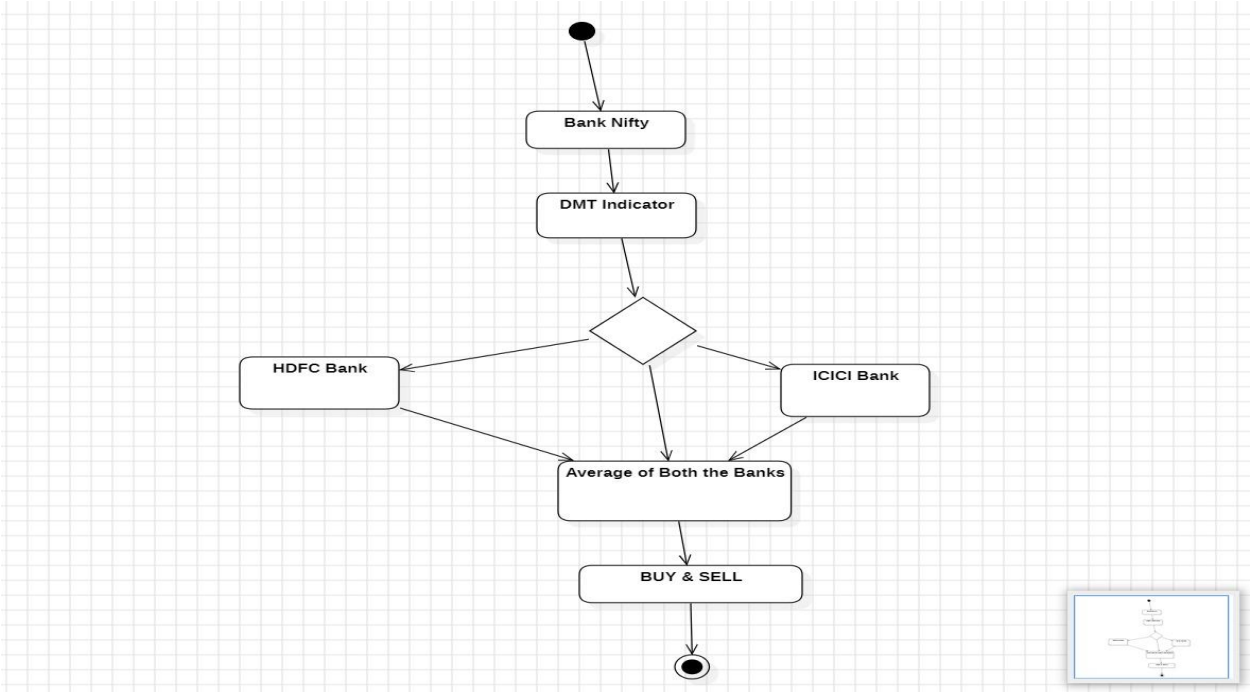


4.2.2 SEQUENCE DIAGRAM

Activity Diagram

The Activity Diagram for calculating the Dual Momentum Tracker (DMT) indicator begins with input data from Bank Nifty. This input initiates the calculation of the DMT indicator, which is used to assess market momentum. Following the DMT calculation, the workflow involves analyzing the performance of individual banks, specifically HDFC Bank and ICICI Bank. This analysis evaluates the financial data from each bank to gauge their performance. Next, the diagram shows the averaging of the data from HDFC Bank and ICICI Bank, providing a consolidated metric for comparison. This averaged data is then used in the decision-making process. Finally, the Trader makes trading decisions—whether to buy or sell stocks—based on the insights gained from the DMT and the averaged data. The Activity Diagram illustrates the

sequential flow of these activities and decision points, providing a clear view of the trading process based on financial data analysis.



4.2.3 Activity Diagram

4.3 System Design

4.3.1 Modular Design

The system is structured into distinct modules for efficient data processing and analysis. The Data Acquisition Module retrieves historical financial data using the `yfinance` library. The Data Processing Module computes daily returns and the Dual Momentum Tracker (DMT), applying smoothing techniques for clarity. The Analysis Module identifies significant local maxima and minima in the DMT to pinpoint critical market movements. Finally, the Visualization Module generates dual-axis plots to effectively present the relationship between Bank Nifty and the DMT indicators.

4.3.2 Database Design

The system utilizes datasets comprising historical closing prices for the Bank Nifty index and two major banks: HDFC and ICICI. Data is sourced directly from Yahoo Finance via the `yfinance` API, ensuring up-to-date and reliable information. The data is organized into pandas DataFrames, facilitating seamless computation of returns and momentum indicators. This structured approach allows for efficient data manipulation and easy scalability to include additional financial instruments if required..

5.IMPLEMENTATION

Implementation of Bank Nifty Line Graph

```
import yfinance as yf
import matplotlib.pyplot as plt
bank_nifty = yf.download("^NSEBANK", start="2020-01-01",
end="2024-08-01")
plt.figure(figsize=(10, 6))
plt.plot(bank_nifty.index, bank_nifty['Close'], label='Bank Nifty Closing Price', color='blue')
plt.title("Bank Nifty Historical Prices")
plt.xlabel("Date")
plt.ylabel("Price")
plt.legend()
plt.grid(True)
plt.show()
```

Dual momentum tracker calculation code

```
import yfinance as yf
import numpy as np
import pandas as pd
banks = ["HDFCBANK.NS", "ICICIBANK.NS"]
data = yf.download(banks, start="2023-01-01", end="2024-08-01")['Close']
dmt_df = pd.DataFrame(index=data.index)
bank_returns = data.pct_change()
dmt_df['DMT'] = bank_returns.mean(axis=1)
dmt_df['DMT_Smoothed'] = dmt_df['DMT'].rolling(window=14).mean()
dmt_df['Min'] = dmt_df['DMT_Smoothed'].rolling(window=5, center=True).apply(lambda x:
x.argmin(), raw=True)
dmt_df['Max'] = dmt_df['DMT_Smoothed'].rolling(window=5, center=True).apply(lambda x:
x.argmax(), raw=True)
threshold = 0.02 * dmt_df['DMT_Smoothed'].std()
```

```
significant_maxima = dmt_df['DMT_Smoothed'] >dmt_df['DMT_Smoothed'].rolling(window=5, center=True).max() - threshold
significant_minima = dmt_df['DMT_Smoothed'] <dmt_df['DMT_Smoothed'].rolling(window=5, center=True).min() + threshold
print("Significant Maxima Dates:\n", dmt_df.index[significant_maxima])
print("Significant Minima Dates:\n", dmt_df.index[significant_minima])
```

Implementation code of including both Bank Nifty and DMT

```
import yfinance as yf
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
banks = ["HDFCBANK.NS", "ICICIBANK.NS"]
bank_nifty = yf.download("^NSEBANK", start="2023-01-01", end="2024-08-01")['Close']
data = yf.download(banks, start="2023-01-01", end="2024-08-01")['Close']
dmt_df = pd.DataFrame(index=bank_nifty.index)
bank_returns = data.pct_change()
dmt_df['DMT'] = bank_returns.mean(axis=1)
dmt_df['DMT_Smoothed'] = dmt_df['DMT'].rolling(window=14).mean()
dmt_df['Min'] = dmt_df['DMT_Smoothed'].rolling(window=5, center=True).apply(lambda x: x.argmin(), raw=True)
dmt_df['Max'] = dmt_df['DMT_Smoothed'].rolling(window=5, center=True).apply(lambda x: x.argmax(), raw=True)
threshold = 0.02 * dmt_df['DMT_Smoothed'].std()
significant_maxima = dmt_df['DMT_Smoothed'] >dmt_df['DMT_Smoothed'].rolling(window=5, center=True).max() - threshold
significant_minima = dmt_df['DMT_Smoothed'] <dmt_df['DMT_Smoothed'].rolling(window=5, center=True).min() + threshold
fig, ax1 = plt.subplots(figsize=(12, 6))
ax1.set_xlabel('Date')
ax1.set_ylabel('Bank Nifty', color='darkblue')
```

```
ax1.plot(bank_nifty.index, bank_nifty, color='darkblue', label='Bank Nifty Closing Price', linewidth=2)
ax1.tick_params(axis='y', labelcolor='darkblue')
ax2 = ax1.twinx()
ax2.set_ylabel('DMT (Smoothed)', color='grey')
ax2.plot(dmt_df.index, dmt_df['DMT_Smoothed'], color='grey', label='DMT (Smoothed)', linewidth=2)
ax2.tick_params(axis='y', labelcolor='grey')
ax2.scatter(dmt_df.index[significant_maxima], dmt_df['DMT_Smoothed'][significant_maxima], color='green', marker='^', s=100, label='Significant Maxima (Green)')
ax2.scatter(dmt_df.index[significant_minima], dmt_df['DMT_Smoothed'][significant_minima], color='red', marker='v', s=100, label='Significant Minima (Red)')
plt.title('Bank Nifty vs Dual Momentum Tracker (DMT)', fontsize=14)
ax1.grid(True)
ax1.legend(loc='upper left')
ax2.legend(loc='upper right')
fig.tight_layout()
plt.show()
```

6.TESTING

6.1 Testing

1. **Data Fetch Check**: Insert print statements to verify the data structure and the dates being fetched.

```
print(data.head())  # Check the first few rows of bank data

print(bank_nifty.head())
```

2. DMT Calculation Check: Print the DMT and DMT_Smoothed columns to ensure they are being calculated correctly.

```
print(dmt_df[['DMT', 'DMT_Smoothed']].head())  # Verify the DMT calculation
```

3.Significant Maxima and Minima Check: Print the dates and values of the significant maxima and minima to verify that they are being identified correctly.

```
print("Significant Maxima:", dmt_df['DMT_Smoothed'][significant_maxima])

print("Significant Minima:", dmt_df['DMT_Smoothed'][significant_minima])
```

4.Plot Testing: Simply run the code to verify that the plot renders without errors. Ensure that the plot contains:

- Bank Nifty in dark blue
- DMT Smoothed line in grey
- Green markers for significant maxima and red markers for significant minima

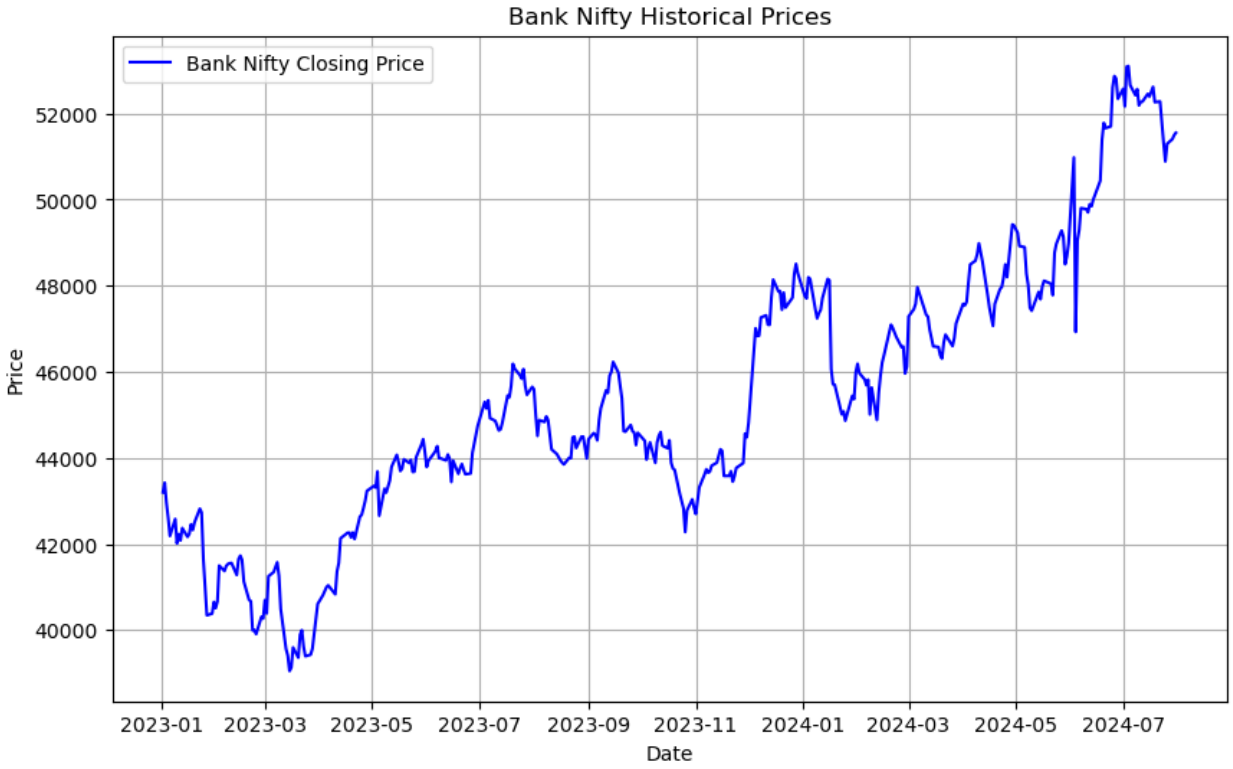
6.2 Test cases

Test Case ID	Test Case Description	Steps to Execute	Expected Output	Test Result
TC_01	Data Fetch Test	1. Run the code and check if bank_nifty and data DataFrames are non-empty. 2. Verify if the date range matches January 2023 to August 2024. 3. Check if the HDFCBANK.NS and ICICIBANK.NS columns are present in data.	1. DataFrames bank_nifty and data should contain historical data. 2. The date range should span from 2023-01-01 to 2024-08-01. 3. The data DataFrame should include both HDFCBANK.NS and ICICIBANK.NS columns.	Pass
TC_02	DMT Calculation Test	1. Verify that the DMT column is computed as the mean of daily returns of HDFC and ICICI. 2. Check if DMT_Smoothed applies a 14-period moving average correctly.	1. The DMT column should correctly reflect the average daily return of the two banks. 2. The DMT_Smoothed column should show smoothed values using a 14-day moving average.	Pass
TC_03	Local Maxima and Minima Test	1. Ensure that the rolling window and threshold identify significant maxima and minima. 2. Check that the code detects both minima and	1. Local maxima &minima should be identified and correctly Marked in the dataset. 2significant peaks and troughs should present after running the	Pass

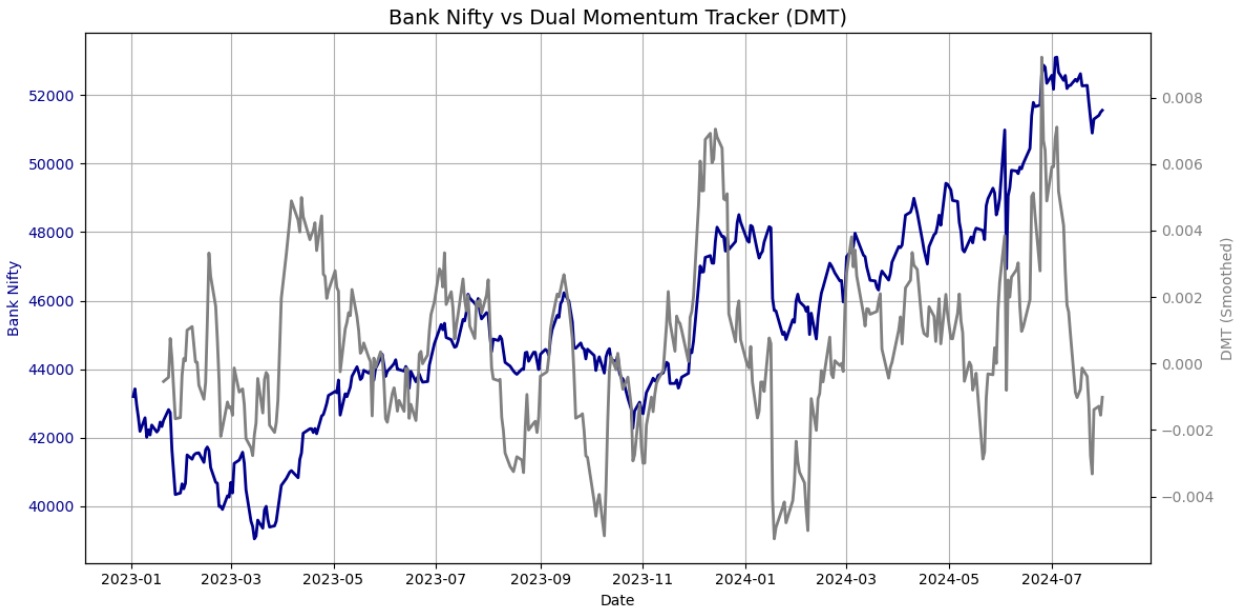
		maxima in the dataset.	analysis.	
TC_04	Threshold Calculation Test	1. Calculate the threshold based on 2% of the standard deviation of DMT_Smoothed. 2. Ensure the threshold is positive.	1. The threshold should be a positive number and reflect 2% of the standard deviation of the DMT_Smoothed column.	Pass
TC_05	Plot Integrity Test	1. Ensure that bank_nifty data is plotted on the left y-axis in blue. 2. Check that the DMT_Smoothed line is plotted on the right y-axis in grey. 3. Validate that significant maxima (green) and minima (red) are marked.	1. The plot should show bank_nifty data on the left y-axis (blue) and DMT_Smoothed on the right y-axis (grey). 2. Maxima should be marked with green arrows, and minima with red arrows.	Pass
TC_06	No Empty Data Post Calculation	1. After calculation of DMT and DMT_Smoothed, verify that none of the columns (DMT, DMT_Smoothed, Max, Min) contain only NaN values.	1. The calculated columns should not be entirely empty or NaN after calculations.	Pass
TC_07	Sufficient Data Points for Rolling Windows	1. Ensure that there are more than 14 data points in the dataset to calculate the 14-period rolling average for DMT_Smoothed.	1. There should be at least 14 data points in the dataset for the rolling average calculation to proceed without error.	Pass

7. OUTPUT SCREEN

Bank Nifty



Bank Nifty With DMT



8. CONCLUSION

8.1 Conclusion

This project successfully developed a Dual Momentum Tracker (DMT) for analyzing the performance of selected banks (HDFC and ICICI) in relation to the Bank Nifty index. By leveraging historical data from Yahoo Finance, the system calculated daily returns and identified significant market trends through local maxima and minima in the DMT. The dual-axis plot effectively visualized the relationship between the Bank Nifty index and the smoothed DMT, providing insights into market dynamics and potential investment opportunities.

8.2 Further Enhancements

Future improvements could include expanding the analysis to incorporate additional banks or financial instruments, enhancing the accuracy of trend detection through advanced smoothing techniques, and integrating machine learning models to predict future market movements based on historical DMT patterns. Additionally, automating the data retrieval and processing pipeline could streamline the analysis, making it more robust and scalable for real-time applications.

9.BIBLIOGRAPHY

9.1 Books References

- Murphy, J. J. (1999) Technical Analysis of the Financial Markets: A Comprehensive Guide to Trading Methods and Applications*. New York Institute of Finance. This book provides foundational knowledge on technical indicators, including momentum indicators, which are integral to this project.

- Pring, M. J. (2002) Technical Analysis Explained: The Successful Investor's Guide to Spotting Investment Trends and Turning Points. McGraw-Hill Education. This book covers in-depth analysis of market cycles and momentum indicators, useful for understanding the concepts applied in the DMT analysis.

9.2 Websites References

* Yahoo Finance.[Yahoo Finance Data](https://finance.yahoo.com). Used for downloading historical stock data for the Bank Nifty index and selected banks (HDFC and ICICI).

*Investopedia(DualMomentumInvesting).[InvestopediaArticle](https://www.investopedia.com/terms/d/dualmomentum.asp). Provided background information on the concept of dual momentum, which inspired the DMT indicator in this project.

9.3 Technical Publication References

- Antonacci, G. (2014).(Dual Momentum Investing) An Innovative Strategy for Higher Returns with Lower Risk. McGraw-Hill Education. This publication introduces the dual momentum concept, which is a core principle used in the project's DMT calculation.

-Sharma, R., & Kumar, S. (2021).(Financial Market Analysis Using Moving Averages and Momentum Indicators). Journal of Financial Engineering, 8(2), 215-230. This paper discusses the application of momentum indicators in financial market analysis.

10.APPENDICES

A. Software Used

- Python 3.10

The primary programming language used for this project.

- Jupyter Notebook

For interactive development and testing of the code.

- yFinance

Python library used to download historical market data.

- NumPy

Used for numerical operations, particularly with arrays.

- Pandas

For data manipulation and analysis, including handling time series data.

- Matplotlib

Used for plotting and visualizing the data and results.

- SciPy

For signal processing, particularly for identifying local maxima and minima.

B. Methodologies Used

- Data Collection

Historical stock data for Bank Nifty and selected banks (HDFC and ICICI) was collected using the yFinance library.

-Dual Momentum Calculation

The Dual Momentum Tracker (DMT) was calculated as the mean of daily returns of the selected banks. A smoothed version of DMT was created using a moving average.

- Fractal Analysis

Local maxima and minima of the smoothed DMT were identified to highlight significant peaks and troughs. A threshold was applied to filter out less significant points.

- Data Visualization

Dual y-axes were used to plot Bank Nifty and the DMT on the same graph, with significant fractal points highlighted

C. Testing Methods Used

- Data Integrity Checks

Ensured that the downloaded data had no missing or erroneous values.

- Threshold Sensitivity Analysis

The threshold for significant changes in DMT was adjusted and tested to ensure that only meaningful peaks and troughs were identified.

- Graphical Validation

Visual inspection of the plotted graphs to verify that the significant maxima and minima align with expected market movements and trends.