AI STOCK TRADING ASSISTANT: PERSONALIZED BUY-SELL CALLS FOR INVESTORS

NAVEEN KUMAR J

Research Proposal

AUGUST 2023

Abstract

In today's interconnected world, stock market trading profoundly impacts individuals from all walks of life. Unfortunately, a significant number of people lack the essential knowledge to trade effectively, leading them to rely on costly financial experts. This often results in frustration and financial losses. In response to this challenge, a groundbreaking AI-powered solution is proposed to provide personalized recommendations specifically tailored to the Nifty50 index that match individual needs and risk preferences. This research focuses on improving Nifty50 index prediction models. Existing models predict daily closing prices without considering investors' backgrounds. The gap lies in the absence of tailored advice for individual investors. To fill this gap, an AI bot will gather information about investors' finances, abilities, and interests. Using advanced techniques like BERT and LSTM, the AI bot will analyze news sentiment and use technical indicators to predict Nifty50 index movements. Drawing from historical data spanning the Nifty50 index's trajectory, the model is poised to yield accurate predictions for the imminent days, furnishing projected values encompassing Open, High, Low, and Close prices for the next trading session. Nifty50 Historic Datasets from 2010 to 2023 and Nifty50 news Headlines Datasets from 2014 to 2023 shall form the bedrock of this research endeavor. The AI bot will categorize investments into low, medium, or high-risk preferences to give personalized recommendations. The project's outcome is an AI bot that provides investors with tailored buy-sell advice based on their risk profile and financial capability. By combining AI, sentiment analysis, and technical indicators, this solution empowers individuals to make informed investment decisions, reducing risks and increasing investment confidence. This research has the potential to democratize stock trading, enabling everyone to trade smarter and independently.

Table of Contents

List of Figures

List of Abbreviations

ADX: Average Directional Index

AI: Artificial Intelligence

API: Application Programming Interface

ASP .NET MVC: Active Server Pages .NET Model-View-Controller

BERT: Bidirectional Encoder Representations from Transformers

CCI: Commodity Channel Index

EMA: Exponential Moving Average

GPU: Graphics Processing Unit

GRU: Gated Recurrent Unit

IDE: Integrated Development Environment

LSTM: Long Short-Term Memory

MACD: Moving Average Convergence Divergence

MAE: Mean Absolute Error

MAPE: Mean Absolute Percentage Error

MSE: Mean Square Error

OHLC: Open, High, Low, Close

OpenAI: Open Artificial Intelligence

OpenAI GPT-3: OpenAI Generative Pre-trained Transformer 3

PSAR: Parabolic Stop and Reverse

RNN: Recurrent Neural Network

RSI: Relative Strength Index

RMSE: Root Mean Square Error

SMA: Simple Moving Average

SQL: Structured Query Language

VS Code: Visual Studio Code

1. Background

In the sprawling theatre of global finance, the stock market takes center stage as a captivating drama that unfolds each day. It's a kingdom where fortunes are made and lost, where economic shifts can sway the fates of individuals and entire nations. The allure of this domain is magnetic, drawing people from all corners of the world from the seasoned Wall Street veterans to the eager rookies looking to strike gold. The stock market is more than just a financial instrument, it's a living, breathing entity that mirrors the pulse of the global economy.

In the intricate realm of the stock market, numbers tell an intriguing tale. Each day, billions of dollars flow through the veins of the market, shaping the destinies of countless investors. Consider this in 2021 alone, global stock markets recorded an average daily trading volume of over $178 billion (World Federation of Exchanges, 2022). This financial orchestra involves millions of participants, from institutional giants to individual traders, each seeking to harness the power of market fluctuations.

Every morning, as the sun rises over financial districts, traders gather with a sense of anticipation and trepidation. Numbers flash on screens, ticker symbols move in dizzying patterns, and the frenetic energy of buying and selling fills the air. It's a high-stakes game of strategy, fueled by data, instincts, and nerves of steel. Each trade holds the promise of profits or the Specter of losses, and the market's mood can swing from exuberance to panic in a matter of minutes.

Yet, amidst this bustling landscape, the challenges are palpable. Research suggests that nearly 80% of retail traders incur losses in the stock market (Barber & Odean, 2013). The reasons are multifaceted complex market dynamics, information overload, and the perennial struggle to make informed decisions in the face of uncertainty. It's no surprise that many turn to financial advisors or experts for guidance, despite the associated cost.

The proposed research addresses a pressing issue in stock market trading by taking inspiration from an array of related studies. (Lamba et al., 2022) used historical data prices to predict Nifty Junior (CNX Nifty) index value and (Sisodia et al., 2022) employs ten years of historical stock price data for Nifty50 Index, spanning from December 2011 to December 2021. However, it is notable that these studies predominantly concentrate on the closing price as the target variable, thus overlooking the significance of encompassing essential parameters such as Open, High, and Low. Incorporating all four OHLC values in predictions holds pivotal importance, particularly for traders engaging in scalping and intraday trading, as it augments the accuracy of forecasting trading ranges for the subsequent day. (Shashank V et al., 2023) added some lights on using machine learning approaches and technical indicators to predict the stock market predictions. They used multiple indicators like Willliams%R, RSI etc. But these studies employ technical indicators separately from news sentiments, and while they provide valuable insights, they do not extend to risk-oriented recommendations or personalized buy-sell calls.

Recurrent Neural Networks (RNN) have demonstrated their efficacy in efficiently predicting time series data. However, the challenges of vanishing and exploding gradients have prompted the emergence of Long Short-Term Memory (LSTM) architectures by (Sonkiya et al., 2021). Due to its substantial parameter count, LSTM requires more time for training, as it needs to learn both short-term and long-term dependencies effectively. Moreover, when considering the optimal choice between LSTM and Gated Recurrent Unit (GRU) for stock prediction on time series data, LSTM generally shines as it exhibits a better capability in capturing intricate temporal patterns and dependencies.

However, while numerous research efforts have focused on the development of prediction models for stock market movements, a significant gap remains in terms of translating these predictions into actionable insights for end-users. Most existing studies primarily emphasize the accuracy of their prediction models, but the practical utility and impact of these predictions for end-users are often left unexplored.

But what if there were a solution that bridged this gap, equipping traders with personalized insights and recommendations? Imagine harnessing the prowess of artificial intelligence to decode market sentiment, analyze historical trends, and provide real-time guidance. This very notion forms the bedrock of our research endeavour – an AI-powered assistant tailored to the Nifty50 index. Just as medical breakthroughs are transforming the battle against diseases, our aim is to revolutionize stock trading, democratizing access to informed decisions and levelling the playing field.

As we delve into the world of stock market prediction, we embark on a journey fueled by data, innovation, and a vision for a more accessible and empowered trading landscape. The statistics are compelling, but the potential for transformation is even more alluring. This study seeks to empower end-users to make strategic decisions aligned with their risk profiles and financial capabilities.

2. Related Work

The literature review critically examines prior research efforts that pave the way for our innovative solution in the realm of stock market prediction and AI-driven decision-making. By delving into related studies, we establish the context, identify gaps, and draw insights that inform our approach's development and distinctiveness. To begin with the dataset (Pant et al., 2023) research aims to offer predictive insights into the Nifty 50 index values, leveraging historical data spanning eight years.

The mentioned indicators are well known and based on price movements, volume of the shares traded and moving averages.(Kumbhare et al., 2023) paper proposes an algorithmic trading strategy utilizing technical indicators, namely the Average Directional Index (ADX), Super trend, and Fibonacci Pivot Points, alongside LSTM neural networks for enhanced accuracy. The ADX, Super trend, and Fibonacci Pivot Points indicators are calculated to identify trends, support and resistance levels.(Jayaraman et al., 2021) uses still more indicators like p SAR Trend, ADX, RSI, MACD, MACD Histogram, CCI Oscillator, Williams %R Indicators, and 10 Days SMA-NIFTY 50 to add values during the feature extraction. In both the papers However, the challenge lies in the fact that stock market movements are not solely driven by historical patterns and technical signals. Sudden and unforeseen events, coupled with breaking news, have the potential to dramatically alter market trajectories, rendering indicator-based predictions less effective in such dynamic scenarios.

(Manasi et al., 2023) and (Goswami et al., 2022) convey about Preprocessing steps, such as Data Normalization through the Min-Max Scaling Method, enhance data quality for training and validation. Also (Lamba et al., 2022) says the transfer function exploration in their work aligns with our methodology's emphasis on LSTM modelling and technical indicators for predicting OHLC values.

(Goswami et al., 2022) focuses on predicting stock prices of five NIFTY 50 shares using a Stacked Long Short-Term Memory (LSTM) model. The purpose is to analyze the impact of the number of epochs on the LSTM model's training performance. The study concludes that the number of epochs significantly impacts model performance during training. While some stocks show better performance with 20 epochs, others perform better with 10 epochs.

(Shah et al., 2022) conducted a comparative study between Linear Regression and Long Short-Term Memory (LSTM) models for predicting Indian Stock Market Index Nifty 50 time-series data. Their investigation aimed to discern the performance disparity of these models in handling time-series data. Employing a supervised approach with data partitioning for Linear Regression and harnessing LSTM's sequential data capabilities, the study found that the LSTM model's design, featuring input, hidden, dense, and output layers, effectively captures temporal patterns. Results exhibited LSTM's superior predictive capabilities over Linear Regression, emphasizing its suitability for time-series data due to its adeptness in capturing intricate temporal dependencies, a key advantage over Linear Regression more suited to non-time series scenarios.(Guo, 2020) also confirms the effectiveness of incorporating news sentiment analysis in stock price prediction using LSTM neural networks will be significant.

(Minnoor and G, 2021) works on the sentiment analysis of tweets with the hashtag "#Nifty50" is incorporated as an input parameter. enhancing the model's market sentiment capturing ability. our study aims to extend this concept to news headlines of nifty50 datasets. While (Mantravadi et al., 2023) envisions sentiment analysis from twitter data for enhanced predictions. (Sonkiya et al., 2021) presents the evolution of sentiment analysis models, introducing FinBert, a tailored version of BERT, fine-tuned for financial sentiment analysis. Nonetheless, relying solely on the Nifty50 hashtag may lack accuracy due to diverse public opinions. To address this, our research incorporates sentiment analysis from reputable Nifty50 news headlines, offering a more reliable insight into market sentiment.

(Sisodia et al., 2022) reports promising results with an average accuracy of approximately 83% for selected stocks, evaluated using metrics like RMSE, MSE, MAE, and MAPE. Meanwhile, (Shashank V et al., 2023) highlights a maximum accuracy of around 73% achieved by a KNN model for next day's price change prediction. (Lamba et al., 2022) and (Lamba et al., 2021) showcase an impressive 98% accuracy in predicting Nifty values through neural network models. While we acknowledge accuracy's significance, our focus is not solely on optimization but rather on providing reliable and actionable insights and recommendations to traders.

(Jethva et al., 2022) introduces a Python chatbot as a key component of the project, enhancing user interaction and accessibility. This chatbot operates as an AI-driven platform, allowing users to engage in real-time conversations and receive personalized stock predictions. Integrated within the web application, the chatbot leverages machine learning techniques to analyze user inputs and provide tailored responses. By offering a seamless interface for users to query and receive stock-related information, the chatbot enhances the accessibility and user-friendliness of the stock prediction system. (Sonkiya et al., 2021) envisions future integration of reinforcement learning to empower the stock market bot with informed trading decisions. Real-time news analysis and instantaneous stock prices would optimize accuracy, potentially revolutionizing intraday trading effectiveness. Innovative approach presents by (Wei, 2017) on utilizing the BOT FRAMEWORK for autonomous learning in higher education. This framework, based on ASP.NET MVC, fosters personalized intelligent chatbot development across diverse channels, enhancing user engagement through Microsoft's technology stack. (Sharma et al., 2019) paper delivers how the part of bot framework being used in Visual studio code and Eclipse as IDE. The proposed model will be using few ideas from this paper but differs with the implementation since we are using different azure services.

However, drawing inspiration from (Jethva et al., 2022) and (Wei, 2017) papers , the proposed approach will be developing the bot framework within Visual Studio Code offers a versatile technology integration beyond Python-based chatbots. This integration enables seamless incorporation of tools and APIs for advanced functionality, complemented by a modular structure allowing the addition of features without disrupting the codebase. This holistic approach integrates various languages, libraries, and APIs, forming a robust foundation for our transformative AI-powered stock trading bot. (Kovačević, 2023) explains how ChatGPT is utilized in his paper. The Proposed model will be consuming Azure AI Services, the OpenAI GPT-3 model which is higher and more advanced version compared to that.

Drawing inspiration from extensive literature reviews and recognizing the critical gaps in current stock market prediction approaches, my proposed paper presents an innovative and transformative solution. In response to the lack of personalized recommendations tailored to individual investors' needs and risk preferences, proposing an AI-powered platform for the Nifty50 index. By harnessing the potential of Deep Learning techniques like LSTM and sentiment analysis using FinBert, alongside the incorporation of technical indicators, this research seeks to provide accurate predictions and insightful buy-sell recommendations.

3. Research Questions

The research questions are designed to address the core focus of this study and encompass:

- How does integrating sentiment analysis from reputable Nifty50 news headlines enhance the accuracy of stock market predictions, addressing limitations of indicator-based models.
- To what extent can an AI-driven platform leverage Deep Learning, sentiment analysis, and technical indicators for personalized buy-sell recommendations, fostering independent decision-making.
- How does integrating an AI-powered chatbot in Visual Studio Code and Azure services revolutionize stock trading accessibility, engagement, and prediction accuracy?

4. Aim and Objectives

This research endeavors to revolutionize stock market trading by introducing an AI-driven solution that tailors Nifty50 index recommendations to individual preferences and risk profiles. Through advanced AI techniques, sentiment analysis, and technical indicators, the aim is to empower investors with personalized insights, fostering confident and independent decision-making while mitigating financial risks.

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

- To Implement Deep Learning Techniques (LSTM) on historical Nifty50 data to derive accurate OHLC values for next trading session, enhancing predictive outcomes.
- To Extract news sentiments on Nifty50 Headlines data, apply FinBert and get outputs in the form of (0 and 1) embeddings and seamlessly integrate them into the existing dataset for comprehensive analysis.
- To Calculate technical indicators manually and seamlessly incorporate them into the dataset, augmenting the predictive model's precision.
- To Develop Personalized Risk-based Recommendation system using Generative AI Models (Openai Azure Services) to offer the buy sell calls that cater to each investor's unique profile.

5. Significance of the Study

Imagine the stock market as a global game affecting everyone's money. But many people don't know all the rules, so they rely on expensive experts. Our new idea is like a helpful robot using smart techniques to give personalized advice about the game. This makes it easier for people to decide what to do with their money and where to invest. This idea could change everything, making the game fairer for everyone. Regular people like you and me could play too, with better chances of winning.

This special robot is for a part of the game called the Nifty50. It uses smart Machine learning techniques to understand news and historic numbers, giving better advice. It doesn't matter if you're from our country or far away; this idea could change how people play the game around the world. Using this robot's advice, people might make choices about their money on their own, feel safer, and maybe win more. So, this research starts a big change in how people use the stock market game.

Here's the exciting part: if this robot idea works well, it could have big effects not only here but in other countries too. It might help people in different places play the game, making it fair and fun for everyone. This research could make the stock market game better for people everywhere, making the stock market a better game to play.

6. Scope of the Study

The scope is strategically defined to achieve well-defined goals within available resources, ensuring feasibility and benefiting diverse users, even those with limited technical expertise.

6.1 In scope

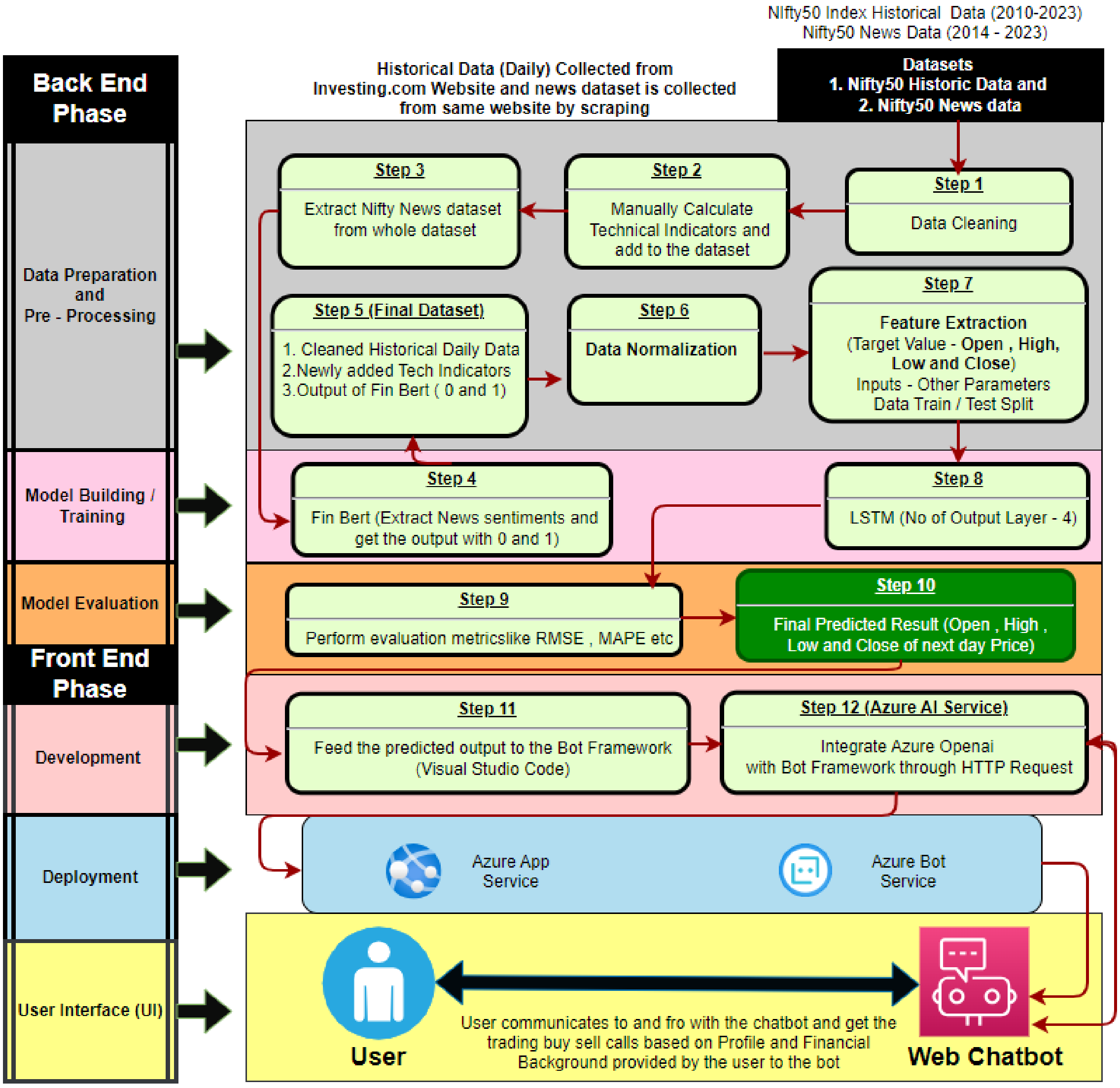
- Focusing on daily datasets and predicted OHLC values provides immediate insights for daily trading, benefiting traders engaged in activities like scalping and intraday trading.
- The AI chatbot's recommendation feature addresses trader's personalized guidance needs.
- The Primary emphasis is on generating OHLC predictions while prioritizing practical insights rather than excessive optimization of prediction accuracy.

6.2 Out of scope

- The research does not explore predictions beyond daily datasets, deliberately omitting the extended timeframes like weekly and monthly data and intricate optimizations.
- It also does not involve the creation of a speech recognition-based chatbot which is a future Scope of Development.

7. Research Methodology

The research methodology employed in this study serves as the guiding framework to achieve the outlined objectives. Through a mutual integration of state-of-the-art AI methodologies, sentiment analysis techniques, and comprehensive technical indicators, this approach is meticulously crafted to pave the way for the development of a robust predictive model tailored to the intricate dynamics of the Nifty50 index. The methodological journey encompasses data preparation, cleaning, and feature extraction, culminating in the utilization of advanced machine learning techniques like Long Short-Term Memory (LSTM) networks and Fin Bert to extract the news sentiment analysis. Rigorous evaluation through established metrics ensures the model's accuracy and effectiveness in providing valuable insights for traders and investors alike.



NIfty50 Index Historical Data (2010-2023)
Nifty50 News Data (2014 - 2023)
Back End Phase
Historical Data (Daily) Collected from Investing.com Website and news dataset is collected from same website by scraping
Datasets
1. Nifty50 Historic Data and
2. Nifty50 News data
Data Preparation and Pre - Processing
Step 3
Extract Nifty News dataset from whole dataset
Step 2
Manually Calculate Technical Indicators and add to the dataset
Step 1
Data Cleaning
Step 5 (Final Dataset)
1. Cleaned Historical Daily Data
2.Newly added Tech Indicators
3.Output of Fin Bert (0 and 1)
Step 6
Data Normalization
Step 7
Feature Extraction
(Target Value - Open , High, Low and Close)
Inputs - Other Parameters
Data Train / Test Split
Model Building / Training
Step 4
Fin Bert (Extract News sentiments and get the output with 0 and 1)
Step 8
LSTM (No of Output Layer - 4)
Model Evaluation
Step 9
Perform evaluation metricslike RMSE , MAPE etc
Step 10
Final Predicted Result (Open , High , Low and Close of next day Price)
Front End Phase
Development
Step 11
Feed the predicted output to the Bot Framework (Visual Studio Code)
Step 12 (Azure AI Service)
Integrate Azure Openai with Bot Framework through HTTP Request
Deployment
Azure App Service
Azure Bot Service
User Interface (UI)
User
User communicates to and fro with the chatbot and get the trading buy sell calls based on Profile and Financial Background provided by the user to the bot
Web Chatbot

Figure 7a Overall Architecture of the Proposed Model

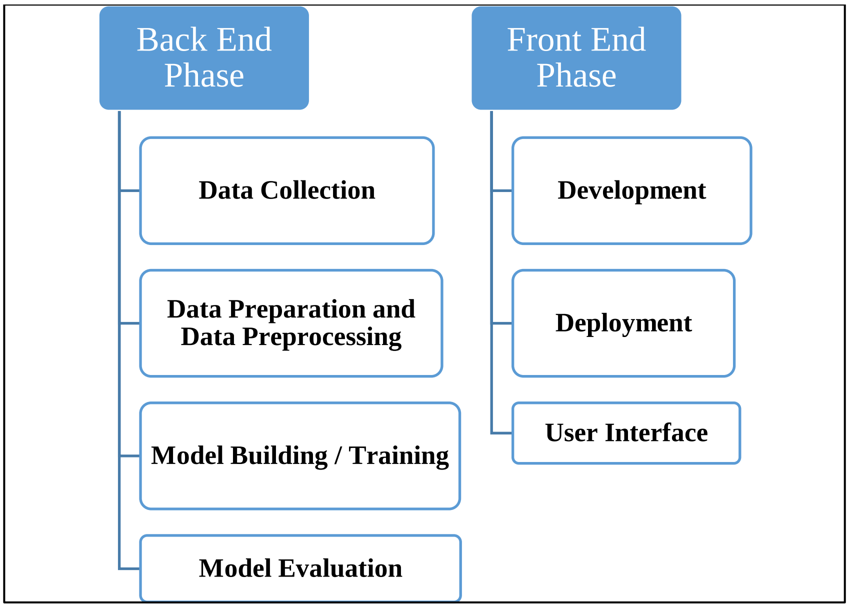


Figure 7b System Workflow Diagram

7.1 Back End Phase

7.1.1 Data Collection

In the initial phase of the back-end process, the focus is on sourcing and gathering the fundamental datasets required for the predictive model.

- **Historical Dataset** - Our research commences with the creation of a comprehensive historical dataset derived from Investing.com. This dataset encompasses crucial attributes such as date, closing price, opening price, highest and lowest trading prices, volume, and percentage change on a daily basis for the Nifty50 index (2010-2023).
- **News Dataset** - We employ web scraping techniques using Parsehub free open-source tool to curate a dataset of Nifty50 index-related news headlines from Investing.com. This dataset encompasses news headlines, publication dates, and publisher names, spanning the years 2014 to 2023.

7.1.2 Data Preparation and Data Pre-processing

7.1.2.1 Data Cleaning

Data Cleaning is a fundamental step in ensuring the reliability of our dataset. We carefully review the Nifty50 historical daily dataset and Nifty50 news dataset to address any missing or erroneous entries. Simultaneously, we scrutinize the news dataset, addressing any noise or irregularities that might affect sentiment analysis accuracy. Through this process, we strive to enhance the dataset's quality, minimizing any potential distortions in subsequent analysis.

7.1.2.2 Addition of Technical Indicators

To empower our dataset with valuable insights, we carefully compute pivotal technical indicators. These include Moving Average Convergence Divergence (MACD), Exponential Moving Average (EMA), Relative Strength Index (RSI), Bollinger Bands, and support and resistance levels. (Kumbhare et al., 2023) showcase these calculations enrich the dataset with delicate market trends, amplifying its analytical depth and relevance.

7.1.2.3 Addition of Fin-Bert Outputs

To infuse our dataset with even more insightful signals, we integrate outputs generated by Fin-Bert. This entails appending the Fin-Bert scores, represented as 0 or 1 denoting bearish or bullish sentiments, respectively. These scores, derived from analyzing news headlines, contribute a valuable layer of sentiment-based information to our dataset, enhancing its predictive capacity. The incorporation of these outputs lays the groundwork for our subsequent model training and evaluation phases.

7.1.2.4 Data Normalization

Data normalization is a crucial process that ensures uniformity and comparability among various features in our dataset. By applying the Min-Max Scaling method, we transform the values of different features to a standardized range, typically between 0 and 1 which is proposed by (Karim and Ahmed, 2021). This normalization procedure removes the potential bias introduced by varying scales, allowing us to analyze and interpret the data consistently. In our finalized dataset, encompassing historical stock data, technical indicators, and Fin-Bert outputs, this normalization guarantees that each feature contributes equitably to the subsequent analysis and model development.

7.1.2.5 Feature Extraction

Feature extraction involves the process of selecting and identifying the most relevant attributes from our comprehensive dataset to construct a compact and informative representation. In our context, this entails choosing specific aspects of the data that hold significant predictive power for our goal of Nifty50 index value prediction. Features like historical price data, technical indicators, and sentiment outputs generated by Fin-Bert are carefully selected based on their potential to influence the target variable – the Nifty50 index values. By focusing on these key attributes, we create a streamlined dataset that captures essential information while minimizing redundancy, setting the stage for effective model training and evaluation.

7.1.2.6 Data Split -Train and Test

In the final phase of data preparation, we partition the dataset into distinct train and test subsets. The training subset facilitates the model's learning process by identifying underlying patterns, while the test subset serves as an independent benchmark for evaluating the model's performance on new data. This partitioning guarantees a comprehensive assessment of the model's accuracy and adaptability.

7.1.3 Model Building / Training

7.1.3.1 Fin Bert

In this phase, we leverage the capabilities of FinBert, a specialized language model designed to understand financial language and context. FinBert processes textual data, such as news headlines, and assigns sentiment labels, usually "bearish" or "bullish," denoting negative or positive sentiments respectively. This sentiment analysis serves as a key input, enriching our model's comprehension of the market's emotional dynamics proved by (Sonkiya et al., 2021).

7.1.3.2 LSTM

Our chosen architectural tool is the Long Short-Term Memory (LSTM) neural network. LSTM is expert at handling sequential data reiterated by (Beniwal et al., 2023) and (Bharti et al., 2022) making it an ideal choice for capturing the temporal dependencies inherent in financial time series. This network is designed to retain and utilize information over varying time intervals, enabling us to identify complex patterns in stock price movements. The Final Output layer of LSTM will be 4 each corresponding to Open, High, Low, and Close (OHLC) values of the next trading day. we're combining linguistic and analytical skills to create a robust model primed to predict the Nifty50 index's movements.

7.1.4 Model Evaluation

To gauge the accuracy of our predictive model, we turn to key metrics. (Fathali et al., 2022) and (Goswami et al., 2022) proves some key metrics as follows : Root Mean Squared Error

(RMSE) serves as a compass, measuring the average size of prediction errors. Additionally, Mean Absolute Percentage Error (MAPE) provides insights into the magnitude of errors relative to actual values. Then comes Mean Squared Error (MSE) metric which quantifies the average squared difference between predicted and actual values, highlighting the magnitude of prediction errors. A lower MSE signifies closer alignment between predictions and reality. By incorporating MSE alongside RMSE and MAPE, our assessment arsenal encompasses a comprehensive view of prediction accuracy, crucial for empowering traders with reliable insights for navigating the stock market landscape.

7.2 Front End Phase

7.2.1 Development

The climax of our backend efforts sets the stage for the development phase, where the fruits of predictive modeling converge with modern software development. The final predicted Open, High, Low, and Close (OHLC) values, generated by our LSTM model, are seamlessly integrated into Visual Studio Code (VS Code). This integration can be achieved through endpoints, facilitating the flow of data between our predictive model and the coding environment.

7.2.1.1 Creation of Bot Framework

With predictions at our disposal, we embark on crafting a user-friendly interface for traders to engage with. Within VS Code, we construct a bot framework that acts as an interactive platform, offering personalized buy-sell recommendations. This AI-powered bot serves as a bridge, providing traders with actionable insights and facilitating informed decision-making. The bot's functionality will be rigorously tested using the Bot Emulator, ensuring seamless communication and accurate recommendation delivery.

7.2.1.2 OpenAI GPT – 3 AI Generative Models

To elevate the sophistication of our bot's recommendations, we harness the power of AI generative models. Azure AI Services the OpenAI GPT-3 model, becomes a cornerstone. By integrating the AI model using Azure services through HTTP Request, our bot transcends scripted responses and gains the ability to generate human-like textual interactions. This synergy of predictive models and AI generative capabilities shapes the user experience, bridging the gap between data-driven insights and intuitive human interaction.

7.2.2 Deployment:

As we transition from development to deployment (Transforming Concepts into Reality), two pivotal components come into play: the Azure App Service Plan and the Azure Bot Service. These services collectively enable the operationalization of our AI-driven trading assistant, facilitating its accessibility to users.(Tellez et al., 2022) performs bot deployment using docker and other techniques , while the Proposed model employs distinct methods by invoking separate services. Please find those separate azure services below.

7.2.2.1 Azure App Service Plan

This crucial infrastructure forms the foundation of our deployment process. The Azure App Service Plan provides a managed hosting environment, ensuring seamless scalability, security, and availability of our application. By leveraging this service, we guarantee that our AI-powered trading assistant is readily accessible to traders across the globe, without concerns of downtime or performance bottlenecks.

7.2.2.1 Azure Bot Service

Building upon the bot framework created earlier the Azure Bot Service acts as a conduit between our AI models and end-users. This service facilitates the integration of our bot with various channels such as web chat, enabling traders to interact with the assistant through familiar interfaces. The Azure Bot Service streamlines the deployment of our trading assistant, ensuring its availability across multiple platforms and devices.

7.2.3 User Interface

The User Interface (UI) phase, representing the culmination of our research, facilitates dynamic interactions with traders. Upon initiation, users engage in dialogue with our AI-driven assistant, sharing financial details, risk tolerance, and investment capacity. Leveraging the predictive model and user input, the assistant calculates personalized investment strategies. For instance, if a user plans to invest $10,000, the assistant divides it across low, moderate, and high-risk categories based on OHLC predictions and risk analysis, offering insights into potential gains and losses. These real-time insights bridge the gap between technology and informed trading decisions.

8. Requirements Resources

8.1 Software Requirements

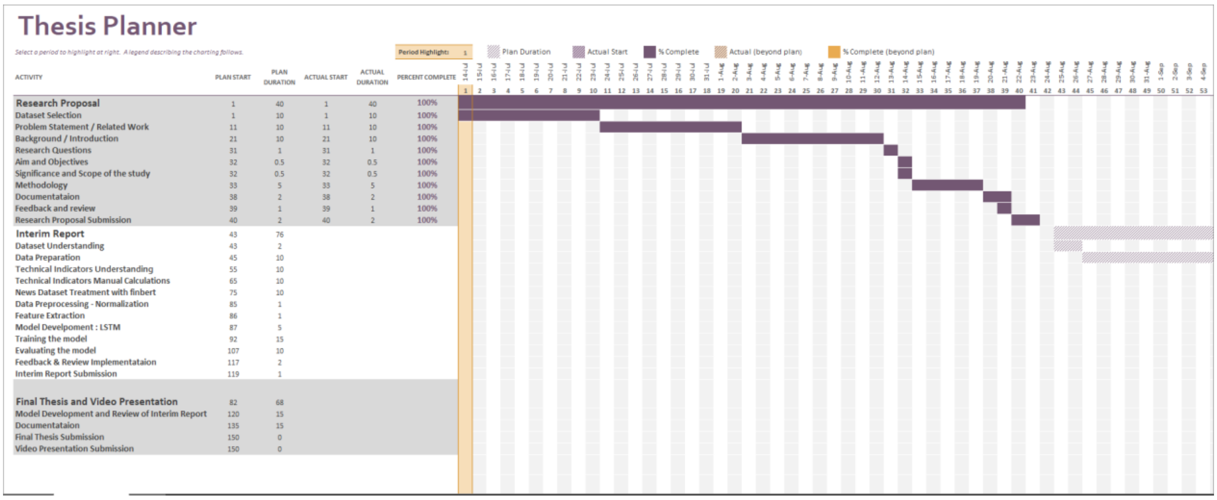
S.NO	Software	Purpose
1	Visual Studio Code	Development environment for coding
2	Google Colab	Cloud-based Jupyter notebook for machine learning
3	Jupyter Notebook	Interactive notebook for data analysis and code
4	Bot Framework	Development of AI chatbot
5	Bot Emulator	Testing and debugging the chatbot
6	Azure App Service Plan	Hosting web applications and chatbots
7	Azure Bot Service	Deploying and managing chatbots on Azure
8	Web Chatbot	Enabling chatbot interaction on websites
9	Python	Programming language for data analysis and AI
10	TensorFlow	Deep learning framework for AI models
11	Azure OpenAI Service	Integration of OpenAI services
12	Other Python Libraries like Pandas, NumPy, Matplotlib	Additional libraries used for data manipulations, analysis, and visualization
13	Database - SQL server 2019	Storage and management of structured data
14	SSMS – 18.12.1	SQL Server Management Studio for database management

8.2 Hardware Requirements

S.NO	Hardware	Purpose
1	Laptop or Desktop computer	Development and execution of code
2	Good Internet access	Access to online resources and cloud platforms
3	Publicly available GPU (e.g., Google Colab, AWS GPU instances)	Enhance performance for model training
4	Local GPU hardware (if available)	Accelerate deep learning model training

9. Research Plan

The research plan encompasses a Gantt chart that delineates a comprehensive timeline for various stages of the project, including data collection, model development, validation, experimentation, and analysis. This structured plan ensures efficient progress and timely achievement of research milestones.

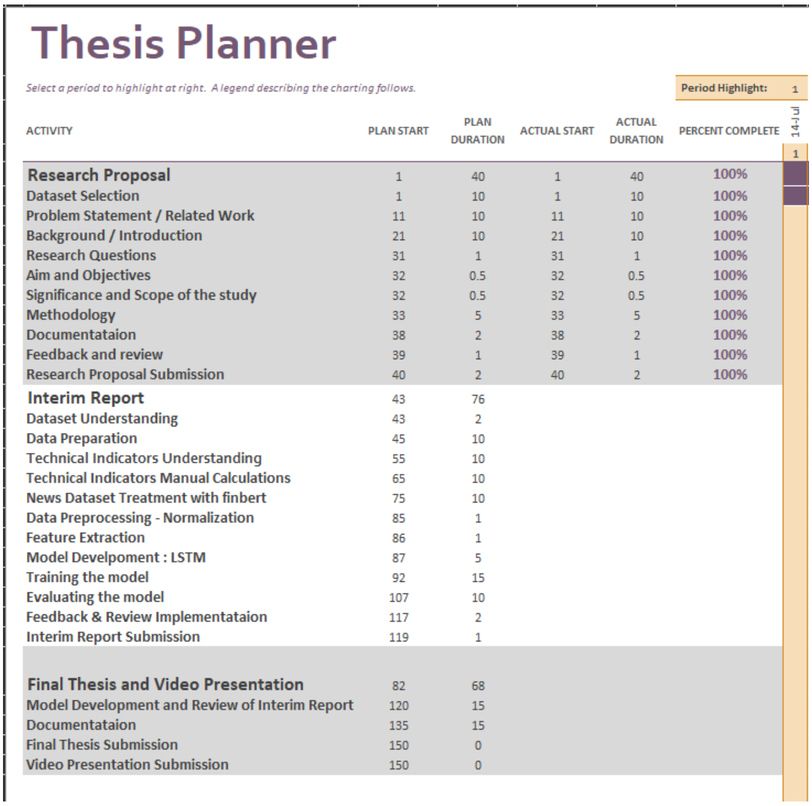


Thesis Planner

Select a period to highlight at right. A legend describing the charting follows.

ACTIVITY	PLAN START	PLAN DURATION	ACTUAL START	ACTUAL DURATION	PERCENT COMPLETE
Research Proposal	1	40	1	40	100%
Dataset Selection	1	10	1	10	100%
Problem Statement / Related Work	11	10	11	10	100%
Background / Introduction	21	10	21	10	100%
Research Questions	31	1	31	1	100%
Aim and Objectives	32	0.5	32	0.5	100%
Significance and Scope of the study	32	0.5	32	0.5	100%
Methodology	33	5	33	5	100%
Documentataion	38	2	38	2	100%
Feedback and review	39	1	39	1	100%
Research Proposal Submission	40	2	40	2	100%
Interim Report	43	76			
Dataset Understanding	43	2			
Data Preparation	45	10			
Technical Indicators Understanding	55	10			
Technical Indicators Manual Calculations	65	10			
News Dataset Treatment with finbert	75	10			
Data Preprocessing - Normalization	85	1			
Feature Extraction	86	1			
Model Develpoment : LSTM	87	5			
Training the model	92	15			
Evaluating the model	107	10			
Feedback & Review Implementataion	117	2			
Interim Report Submission	119	1			
Final Thesis and Video Presentation	82	68			
Model Development and Review of Interim Report	120	15			
Documentataion	135	15			
Final Thesis Submission	150	0			
Video Presentation Submission	150	0			

Figure 9a Gantt Chart


Period Highlight: 1
14-Jul
1

Thesis Planner

Select a period to highlight at right. A legend describing the charting follows.

ACTIVITY	PLAN START	PLAN DURATION	ACTUAL START	ACTUAL DURATION	PERCENT COMPLETE
Research Proposal	1	40	1	40	100%
Dataset Selection	1	10	1	10	100%
Problem Statement / Related Work	11	10	11	10	100%
Background / Introduction	21	10	21	10	100%
Research Questions	31	1	31	1	100%
Aim and Objectives	32	0.5	32	0.5	100%
Significance and Scope of the study	32	0.5	32	0.5	100%
Methodology	33	5	33	5	100%
Documentataion	38	2	38	2	100%
Feedback and review	39	1	39	1	100%
Research Proposal Submission	40	2	40	2	100%
Interim Report	43	76			
Dataset Understanding	43	2			
Data Preparation	45	10			
Technical Indicators Understanding	55	10			
Technical Indicators Manual Calculations	65	10			
News Dataset Treatment with finbert	75	10			
Data Preprocessing - Normalization	85	1			
Feature Extraction	86	1			
Model Develpoment : LSTM	87	5			
Training the model	92	15			
Evaluating the model	107	10			
Feedback & Review Implementataion	117	2			
Interim Report Submission	119	1			
Final Thesis and Video Presentation	82	68			
Model Development and Review of Interim Report	120	15			
Documentataion	135	15			
Final Thesis Submission	150	0			
Video Presentation Submission	150	0			

Figure 9b Gantt Chart – Activity View

References

Beniwal, M., Singh, A. and Kumar, N., (2023) Predicting Next Quarter Nifty 50 Price using Genetic Algorithm and Support Vector Regression. In: *2023 2nd International Conference on Applied Artificial Intelligence and Computing (ICAAIC)*. pp.631–635.

Bharti, S.K., Tratiya, P. and Gupta, R.K., (2022) Stock Market Price Prediction through News Sentiment Analysis & Ensemble Learning. In: *2022 IEEE 2nd International Symposium on Sustainable Energy, Signal Processing and Cyber Security (iSSSC)*. pp.1–5.

Fathali, Z., Kodia, Z. and Ben Said, L., (2022) Stock Market Prediction of NIFTY 50 Index Applying Machine Learning Techniques. *Applied Artificial Intelligence*, [online] 361, p.2111134. Available at: https://doi.org/10.1080/08839514.2022.2111134.

Goswami, S., Sagar, A.K., Nand, P. and Khalaf, O.I., (2022) Time Series Analysis Using Stacked LSTM Model for Indian Stock Market. In: *2022 IEEE IAS Global Conference on Emerging Technologies (GlobConET)*. pp.399–405.

Guo, Y., (2020) Stock Price Prediction Based on LSTM Neural Network: the Effectiveness of News Sentiment Analysis. In: *2020 2nd International Conference on Economic Management and Model Engineering (ICEMME)*. pp.1018–1024.

Jayaraman, G., Venkatachalam, M. and M, T., (2021) Predicting Daily Closing Value of Indian Stock Index Through Artificial Neural Network. In: *2021 International Conference on Decision Aid Sciences and Application (DASA)*. pp.828–837.

Jethva, C., Dudani, S., Malik, E., Sonje, M. and Tanna, G., (2022) Stock Analyzer and Bot using Machine Learning. In: *2022 IEEE Region 10 Symposium (TENSYMP)*. pp.1–6.

Karim, M.E. and Ahmed, S., (2021) A Deep Learning-Based Approach for Stock Price Prediction Using Bidirectional Gated Recurrent Unit and Bidirectional Long Short Term Memory Model. In: *2021 2nd Global Conference for Advancement in Technology (GCAT)*. pp.1–8.

Kovačević, D., (2023) Use of ChatGPT in ESP Teaching Process. In: *2023 22nd International Symposium INFOTEH-JAHORINA (INFOTEH)*. pp.1–5.

Kumbhare, P., Kolhe, L., Dani, S., Fandade, P. and Theng, D., (2023) Algorithmic Trading Strategy Using Technical Indicators. In: *2023 11th International Conference on Emerging Trends in Engineering & Technology - Signal and Information Processing (ICETET - SIP)*. pp.1–6.

Lamba, V., Hooda, S., Solanki, V., Bhardwaj, V., Lilhore, U.K. and Khullar, V., (2022) Nifty Junior (CNX

Nifty) Value Prediction by Applying Depth Psychology Approach in Machine Learning. In: *2022 10th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)*. pp.1–4.

Manasi, H., Kumar, A., Kushwah, V. and Agrawal, P., (2023) A comparative study of the various optimizations applied on an LSTM for the application of Stock Price Predictions. In: *2023 Third International Conference on Secure Cyber Computing and Communication (ICSCCC)*. pp.684–689.

Mantravadi, B.S., Karedla, A., Blessitta, G., Krishna, C.R. and Abitha, C., (2023) Prediction of Stock Prices based on Historical Data using Linear Regression Model through LSTM Algorithm. In: *2023 International Conference on Advancement in Computation & Computer Technologies (InCACCT)*. pp.96–100.

Minnoor, N. and G, S., (2021) Nifty Price Prediction from Nifty SGX using Machine Learning, Neural Networks and Sentiment Analysis. In: *2021 7th International Conference on Computer and Communications (ICCC)*. pp.1256–1260.

Pant, K., Rawat, R. and Pant, A., (2023) Forecasting of Nifty 50 Index Volatility Using Seasonal ARIMA Model. In: *2023 International Conference on Advancement in Computation & Computer Technologies (InCACCT)*. pp.271–274.

Sharma, V.S., Mehra, R., Kaulgud, V. and Podder, S., (2019) A Smart Advisor for Software Delivery - A Bot Framework for Awareness, Alerts and Advice. In: *2019 IEEE/ACM 1st International Workshop on Bots in Software Engineering (BotSE)*. pp.22–23.

Shashank V, N.P., Ramanan, V.S., Hudson D, A.L. and Sheik Abdullah, A., (2023) An Efficient Decision Support System for Stock Market Prediction using Technical Indicators as Features by Machine Learning Approaches. In: *2023 4th International Conference for Emerging Technology (INCET)*. pp.1–6.

Sisodia, P.S., Gupta, A., Kumar, Y. and Ameta, G.K., (2022) Stock Market Analysis and Prediction for Nifty50 using LSTM Deep Learning Approach. In: *2022 2nd International Conference on Innovative Practices in Technology and Management (ICIPTM)*. pp.156–161.

Sonkiya, P., Bajpai, V. and Bansal, A., (2021) Stock price prediction using BERT and GAN. [online] Available at: https://arxiv.org/abs/2107.09055v1 [Accessed 1 Jul. 2023].

Tellez, N., Serra, J., Ebreso, U., Opara, K., Kumar, Y., Li, J.J. and Morreale, P., (2022) An Assure AI Bot (AAAI bot). In: *2022 International Symposium on Networks, Computers and Communications (ISNCC)*. pp.1–5.