**DAYANANDA SAGAR UNIVERSITY**

**Devarakaggalahalli, Harohalli Kanakapura Road, Dt, Ramanagara, Karnataka 562112**



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**(Artificial Intelligence and Machine Learning)**



**FINANCIAL TECHNOLOGY (FINTECH) (22AM3613)**

**Project Report on**

**“Hybrid Machine Learning Approaches for Analyzing Stock Price Trends”**

By

**Hemang Singh Sengar (ENG22AM0097)**

**Hena Basheer (ENG22AM0098)**

**Under the supervision of**

**Dr. Hanumanth Sastry Sistla**

Professor, Department of CSE (AI&ML), SOE

**Dr. Mude Nagarjuna Naik**

Assistant Professor, Department of CSE (AI&ML), SOE

**2024-2025**

**DAYANANDA SAGAR UNIVERSITY**

**School of Engineering**

**Department of Computer Science & Engineering**

**(Artificial Intelligence and Machine Learning)**

Devarakaggalahalli, Harohalli Kanakapura Road, Dt, Ramanagara, Karnataka 562112

**Department of Computer Science & Engineering (AI-ML)**



**CERTIFICATE**

This is to certify that the FINTECH (22AM3613) titled **“Hybrid Machine Learning Approaches for Analyzing Stock Price Trends”**is carried out by **Hemang Singh Sengar (ENG22AM0097), Hena Basheer (ENG22AM0098),** Bonafide students of Bachelor of Technology in Computer Science and Engineering (Artificial Intelligence and Machine Learning) at the School of Engineering, Dayananda Sagar University.

**Dr. Hanumanth Sastry Sistla**

Professor

Dept. of CSE (AI&ML)

SoE, DSU

**Dr. Mude Nagarjuna Naik**

Assistant Professor

Dept. of CSE (AI&ML)

SoE, DSU

**Dr. Jayavrinda Vrindavanam**

Professor & Chairperson

Dept. of CSE (AI&ML)

SoE, DSU

**DECLARATION**

We, **Hemang Singh Sengar (ENG22AM0097), Hena Basheer (ENG22AM0098),** are students of the sixth semester B.Tech in Computer Science and Engineering (AI&ML), at School of Engineering, Dayananda Sagar University, hereby declare that the Fintech project titled **“Hybrid Machine Learning Approaches for Analyzing Stock Price Trends”**has been carried out by us and submitted in partial fulfillment for the award of degree in Bachelor of Technology in Computer Science and Engineering(AI&ML) during the academic year 2024-2025.

**ACKNOWLEDGEMENT**

It is a great pleasure for us to acknowledge the assistance and support of many individuals who have been responsible for the successful completion of this project work.

First, we take this opportunity to express our sincere gratitude to School of Engineering & Technology, Dayananda Sagar University for providing us with a great opportunity to pursue our Bachelor’s degree in this institution.

We would like to thank **Dr. Udaya Kumar Reddy K R,** **Dean**, **School of Engineering**, **Dayananda Sagar University** for his constant encouragement and expert advice. It is a matter of immense pleasure to express our sincere thanks to **Dr. Jayavrinda Vrindavanam,** **Professor & Chairperson**, **Computer Science and Engineering (Artificial Intelligence and Machine Learning)**, **School of Engineering, Dayananda Sagar University,** for providing the right academic guidance that made our task possible.

We would like to thank our guides **Dr. Hanumath Sastry Sistla & Dr. Mude Nagarjuna Naik, Professor & Assistant Professor, Dept. of Computer Science and Engineering (Artificial Intelligence and Machine Learning),** for sparing their valuable time to extend help in every step of our Mini project work towards FIntech Course, which paved the way for smooth progress and the fruitful culmination of the research.

We are also grateful to our family and friends who provided us with every requirement throughout the course. We would like to thank one and all who directly or indirectly helped us in the Artificial Intelligence Project.

**Hemang Singh Sengar (ENG22AM0097) Hena Basheer (ENG22AM0098)**

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **S.No** | **Topic** | **Page No** |
|  | Abstract | 6-7 |
|  | SDG Goals | 8 |
| 1 | Introduction | 9 |
| 2 | Problem statement | 9 |
| 3 | Objectives | 10 |
| 4 | Literature review | 11-12 |
| 5 | Project description | 13-14 |
| 6 | Methodology | 15 |
| 7 | Result and analysis | 16-18 |
| 8 | Conclusion | 19-20 |
|  | Reference | 20-21 |
|  | Appendix | 22-25 |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **S.No** | **Topic** | **Page No** |
| 1 | Workflow Diagram | 14 |
| 2 | Methodology Flowchart | 15 |
| 3 | Output for Infosys | 18 |
| 4 | Output for HDFC Bank | 18 |
| 5 | Output for ICICI Bank | 19 |

**Abstract**

The automation of comprehensive stock analysis represents a significant advancement in financial technology, offering an efficient and insightful solution for aiding investment decision-making. This report explores the design, implementation, and effectiveness of an automated stock analysis system, which leverages Python programming to fetch market data, calculate technical indicators, predict trends using a Long Short-Term Memory (LSTM) model, and generate qualitative reports via an AI language model. The objective is to synthesize complex financial data, provide objective insights, and minimize the manual research associated with traditional investment analysis.

The system employs a combination of data acquisition from yfinance, technical indicator calculations, and advanced predictive modeling. Historical stock data, including opening, high, low, closing prices, and volume, are continuously monitored. These data points are processed to derive critical technical indicators such as the Simple Moving Average (SMA), Exponential Moving Average (EMA), Moving Average Convergence Divergence (MACD), and Relative Strength Index (RSI). These indicators provide real-time metrics to the system, which then uses this information to determine market sentiment and potential trends.

Key components of the system include Python as the programming language, yfinance for data retrieval, pandas for data manipulation, tensorflow.keras for the LSTM model, and Anthropic's Claude API for generating natural language reports. The LSTM model serves as the central processing unit for trend prediction, trained on historical data to classify future market movements as "Bullish," "Bearish," or "Sideways." The Claude API then synthesizes this quantitative analysis with fundamental data (P/E ratio, EPS, sector) to produce a concise, actionable report for investors.

One of the primary benefits of this automated system is its ability to distill vast amounts of financial data into digestible, actionable insights. By integrating technical indicators, machine learning-driven trend predictions, and AI-generated narrative analysis, the system prevents information overload and provides a holistic view of a stock's potential. Additionally, the automation significantly reduces the need for extensive manual research, making it particularly useful for retail investors seeking informed decisions without dedicating excessive time to data interpretation.

Furthermore, the system can be easily expanded or modified to suit different analysis methodologies and market conditions. By integrating additional technical indicators, incorporating more sophisticated machine learning models, or adjusting the parameters for report generation, the system can be tailored to meet specific investment strategies. The modular design also suggests potential for future enhancements, such as real-time market alerts or direct integration with trading platforms for automated execution.

In conclusion, the automation of stock analysis using Python, yfinance, LSTM models, and AI language models represents a practical and innovative approach to modern financial decision-making. It offers numerous benefits, including efficient data synthesis, objective trend prediction, and reduced analytical complexity. As technology advances, such systems are likely to become even more sophisticated, further enhancing their utility and efficiency in various investment applications.

**SDG Goals:**

**Goal 8: Decent Work and Economic Growth**

By making sophisticated financial insights more accessible, this project empowers retail investors to make informed decisions, thereby promoting more inclusive economic participation. Furthermore, it actively encourages a culture of smart investment and enhanced financial literacy, which are fundamental pillars for sustainable economic development and individual wealth creation.

**Goal 9: Industry, Innovation, and Infrastructure**

The project's foundational use of cutting-edge AI models, such as Claude, and Python-based automation represents a significant stride in fostering innovation within the FinTech industry. It effectively leverages modern technological tools and APIs (e.g., yfinance, Anthropic's API) to build a robust and intelligent analytical infrastructure that supports advanced financial analysis.

**Goal 12: Responsible Consumption and Production**

By providing investors with clear, trend-based guidance derived from data-driven analysis, the system actively discourages impulsive or purely speculative trading behavior. This approach encourages more thoughtful and responsible investment decisions, aligning financial practices with principles of sustainable and informed resource allocation.

**1. INTRODUCTION:**

The evolving complexity and data volume of modern financial markets necessitate sophisticated analytical tools beyond traditional methods, particularly for retail investors seeking informed decisions amidst information overload. This report introduces an automated stock analysis system designed to address these challenges by providing a comprehensive and accessible solution for evaluating market dynamics. Our system integrates robust data acquisition from yfinance, meticulous calculation of key technical indicators (SMA, EMA, MACD, RSI), advanced predictive capabilities via a Long Short-Term Memory (LSTM) deep learning model for trend forecasting, and leverages the Anthropic Claude API to generate insightful, natural language reports. This multi-faceted approach aims to synthesize complex financial data into actionable insights, ultimately empowering investors with enhanced decision-making capabilities.

**2. PROBLEM STATEMENT:**

Retail investors frequently struggle to interpret the intricate stock market data and identify clear trends, primarily due to the inherent complexity of technical indicators and a notable absence of easily accessible analytical tools. Existing solutions often focus on either technical analysis or machine learning in isolation, but rarely combine both effectively. Furthermore, there is a significant gap in integrating machine learning predictions with natural language explanations, making outputs largely uninterpretable for non-expert users. This challenge leads to difficulties in making informed decisions, potential for emotional or impulsive investing, and a lack of bridge between raw financial data and understandable reports. Thus, there is a pressing need for a unified, intelligent system that simplifies this process by providing objective, comprehensive, and readily understandable stock analyses to empower retail investors.

**3. OBJECTIVES:**

**3.1 To Analyze Stock Price Trends:** This objective focuses on building a robust model for stock price trend analysis. The system will utilize key technical indicators such as Simple Moving Average (SMA), Exponential Moving Average (EMA), Moving Average Convergence Divergence (MACD), and Relative Strength Index (RSI) to understand market dynamics and movements. The aim is to accurately classify these trends as bullish, bearish, or neutral, providing foundational insights into stock performance.

**3.2** **To Build a Deep Learning Model (LSTM):** This objective involves the development and implementation of a Long Short-Term Memory (LSTM) deep learning model. This model will be trained using sequences of 60 days of technical indicator data as input. The core purpose is to achieve accurate forecasting of stock trends based on historical patterns, which will then be used for further analysis and reporting.

**3.3** **Natural Language Reporting with Claude AI (LLM):** This objective focuses on enhancing the interpretability of the analytical output. It involves using the Claude 3 Opus AI via API to translate complex technical and machine learning results into easy-to-understand summaries. The goal is to provide clear, investor-friendly stock analysis and investment guidance for non-expert users.

**3.4** **Hybrid AI System Integration:** This objective aims to combine different advanced AI methodologies into a cohesive and powerful analytical tool. It involves integrating deep learning (specifically the LSTM model) with explainable AI, in the form of Natural Language Processing from Claude AI. The ultimate goal is to create a unified system that is both accurate in its predictions and highly interpretable for its users.

**3.5** **Real-World FinTech Application Deployment:** This objective is centered on demonstrating the practical utility and real-world applicability of the developed system. It involves applying the hybrid model to actual NIFTY 50 stocks to generate practical insights. This deployment will showcase how AI can effectively assist in everyday retail investment decisions, making financial insights more accessible and promoting informed choices.

**4. LITERATURE REVEIW:**

**1. Predicting Stock Market Trends Using Technical Analysis**

* Authors: D. Patel and V. Shah
* Year: 2015
* Core Functionality: This study explored the use of technical indicators combined with machine learning models for stock trend prediction.
* Key Findings: Support Vector Machine (SVM) and Random Forest algorithms demonstrated high accuracy in predicting trends.
* Limitations: The methodology was limited to the Indian market and focused on short-term trends.

**2. Predicting Direction of Stock Price Index Using ANN and SVM**

* Authors: Pradeep Kumar S, Chandana Gireesh, Riti Dass, Sneha Sinha, Sumit Kumar, Subhra Chakraborty
* Core Functionality: This research presents a device with a microcontroller for monitoring vital signs of stroke patients with partial paralysis.
* Key Benefits:
  + Detects and transmits crucial health data (temperature, heart rate) to caregivers via GSM module in case of abnormalities.
  + Provides detailed information about detected issues for informed decision-making by caregivers.
* Focus: This system prioritizes real-time health monitoring and facilitates proactive interventions for stroke patients.

**3. Deep Learning with LSTM Networks for Stock Price Trend Forecasting**

* Focus: This system prioritizes real-time health monitoring and facilitates proactive interventions for stroke patients.
* Deep Learning With Long Short-Term Memory Networks for Financial Market Predictions / Deep Learning with LSTM Networks for Stock Price Trend Forecasting
* Authors: T. Fischer and C. Krauss
* Year: 2018
* Core Functionality: This work extensively explored the application of Long Short-Term Memory (LSTM) networks for predicting financial market trends, specifically stock price trend forecasting using technical indicators.
* Key Findings: LSTM models captured long-term dependencies and improved prediction accuracy. They also outperformed traditional ML methods for time series trend prediction.
* Limitations: The use of LSTM models can incur high computational costs and are sensitive to overfitting, requiring long sequences for accuracy.

**4. Stock Trend Prediction using Sentiment Analysis and Technical Indicators**

* Authors: V. Bhat and N. Vasaikar
* Year: 2020
* Core Functionality: This research investigated combining news sentiment analysis with technical indicators for stock trend prediction.
* Key Findings: Integrating news sentiment data with indicators was found to enhance the accuracy of trend predictions.
* Limitations: Sentiment data is often noisy and sometimes misleading.

**5. Using Large Language Models for Financial Report Generation**

* Authors: OpenAI and Anthropic
* Year: 2023
* Core Functionality: This work explored the utility of Large Language Models (LLMs), such as Claude and GPT, for generating natural language financial reports.
* Key Findings: LLMs help convert structured financial data into understandable narratives for non-experts.
* Limitations: A noted limitation is the potential for LLMs to "hallucinate" facts without strict prompt control or external data grounding.

1. **PROJECT DESCRIPTION:**

This project addresses the inherent complexities and inefficiencies of manual stock market analysis by developing an automated stock analysis system. Designed to assist individual and retail investors, it aims to provide clear, data-driven insights, mitigating issues like information overload, subjective interpretation, and the time-consuming nature of traditional research.

The system's core functionality is orchestrated through a structured pipeline:

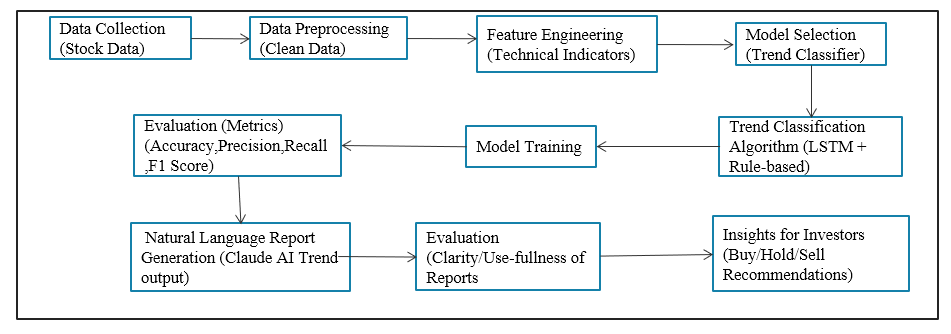
1. **Data Collection:** Utilizing the yfinance library, the system retrieves historical stock data (Open, High, Low, Close, Volume) for a specified period, including NIFTY 50 stocks. It also fetches essential fundamental data such as P/E ratio, EPS, and sector information to provide a holistic view.
2. **Preprocessing:** Raw data is cleaned by handling missing values (dropna()) to ensure the integrity of subsequent analyses.
3. **Technical Indicator Calculation:** Key technical indicators, including Simple Moving Average (SMA), Exponential Moving Average (EMA), Moving Average Convergence Divergence (MACD), and Relative Strength Index (RSI), are computed to identify trends and momentum.
4. **Trend Prediction with LSTM:** A Deep Learning model, specifically a Long Short-Term Memory (LSTM) network, is employed to predict stock price trends (Bullish, Bearish, or Sideways). This model analyzes sequences of 60 days of technical indicators and volume data to forecast future movements.
5. **Natural Language Report Generation:** The Anthropic Claude API is integrated to translate the complex technical and machine learning outputs into easily understandable, narrative-based stock analysis reports. These reports summarize indicators, current trends, and provide clear investor recommendations and sentiments.

**Software Components and Libraries:** The project is built primarily in Python, leveraging a suite of powerful libraries:

* yfinance: For robust and reliable access to historical stock data and fundamental information.
* pandas: For efficient data manipulation and analysis.
* numpy: For numerical operations, particularly in data preprocessing and model input preparation.
* tensorflow.keras: For building, training, and loading the LSTM deep learning model.
* sklearn.preprocessing.MinMaxScaler: For scaling features to optimize model performance.
* anthropic: To interface with the Claude AI for generating natural language reports.
* os and dotenv: For secure management of API keys and environment variables.

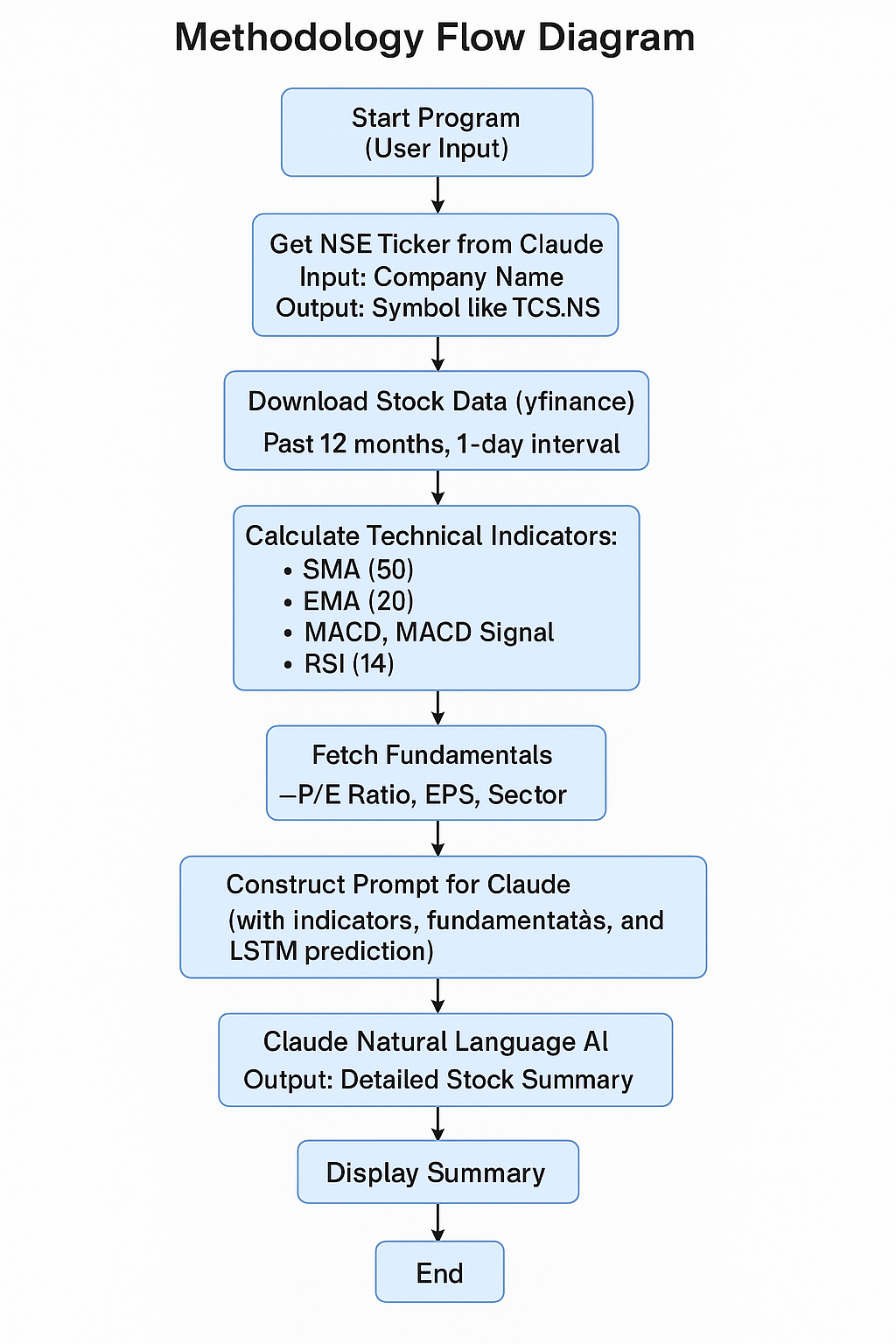
This targeted approach optimizes the analytical process, delivering comprehensive insights while significantly minimizing the human effort required for manual monitoring and interpretation. The project ultimately aims to create a user-friendly and reliable solution for promoting informed investment decisions and fostering healthier portfolio growth by providing objective, AI-driven stock analyses.

* 1. **WORKFLOW DIAGRAM:**

****

**6. METHODOLOGY:**

This automated stock analysis system orchestrates a self-regulating analytical loop, meticulously designed to ensure comprehensive and continuous evaluation of stock market trends. The process begins with **Data Collection**, where the system diligently fetches real-time and historical stock data, including price movements and essential fundamental information, from Yahoo Finance using the yfinance library. This stream of raw data then undergoes crucial **Data Preprocessing**, where any inconsistencies or missing values are promptly handled to maintain data integrity. Subsequently, the system continuously computes a suite of **Technical Indicators**, such as SMA, EMA, MACD, and RSI, acting as real-time gauges of market momentum and trend strength. These indicators, along with others like support, resistance, and return levels, are periodically fed into the **Deep Learning Model (LSTM)**. This LSTM model, serving as the analytical brain, constantly processes sequences of the most recent 60 days of scaled technical data, comparing them against learned patterns to classify the current stock trend as "Bullish," "Bearish," or "Sideways." When a prediction is made, the system springs into action, dynamically constructing a detailed prompt that synthesizes all the technical indicators, fundamental data, and the LSTM's trend prediction. This prompt is then sent to **Natural Language Reporting with Claude AI**, which acts as the intelligent interpreter. Claude AI instantly generates a comprehensive, easy-to-understand stock analysis report, complete with summaries, trend explanations, and investor recommendations. This continuous cycle ensures that the system provides precise and timely insights, operating as a closed-loop feedback mechanism for investors, waiting for the next data update to reassess and report on the ever-changing market dynamics.



**7. RESULT AND ANALYSIS:**

**7.1 Data Collection and Initial Acquisition:**

* **Component:** yfinance library and various financial data sources.
* **Action:** Historical price data (Open, Close, High, Low, Volume) and key fundamental indicators (P/E ratio, EPS, Sector) for selected stocks, particularly from the NIFTY 50, are systematically fetched for a specified period.
* **Process:** The system actively connects to financial APIs to acquire raw market data, establishing the foundational dataset for all subsequent analyses.
* **Result/Analysis:** The successful retrieval of comprehensive and diverse financial data ensures that the analytical stages have accurate and sufficient input. This initial phase validates the availability and accessibility of required market information for **stock analysis**.

**7.2 Data Preprocessing:**

* **Component:** pandas library for data manipulation.
* **Action:** Identified and removed any missing or inconsistent data points (NaN values) from the collected dataset.
* **Process:** Data cleaning operations are applied to ensure the integrity and uniformity of the dataset, making it suitable for numerical computations and robust model training.
* **Result/Analysis:** A clean, reliable, and standardized dataset is produced, which is crucial for preventing errors and biases in subsequent indicator calculations and **stock trend predictions**.

**7.3 Technical Indicator Calculation:**

* **Component:** Custom Python functions leveraging pandas for calculations.
* **Action:** Essential **stock market** technical indicators such as Simple Moving Average (SMA\_50), Exponential Moving Average (EMA\_20), MACD (including MACD line, MACD Signal, and MACD Histogram), and Relative Strength Index (RSI\_14) are computed based on the preprocessed historical closing prices. Additional features like 21-day returns, 20-day support, and resistance levels are also calculated.
* **Process:** Mathematical algorithms are applied to the **stock's** time-series data, transforming raw price movements into quantifiable metrics that represent momentum, volatility, and trend strength.
* **Result/Analysis:** The raw **stock data** is augmented with a rich set of analytical features. These indicators provide immediate insights into the **stock's** current standing, such as whether it's overbought/oversold (RSI), experiencing bullish/bearish momentum (MACD), or showing long/short-term trend directions (SMAs/EMAs).

**7.4 Trend Prediction (LSTM Model):**

* **Component:** Pre-trained Long Short-Term Memory (LSTM) deep learning model.
* **Action:** The latest 60-day sequences of the calculated and scaled technical indicators (including Volume) are fed into the LSTM model.
* **Process:** The LSTM, leveraging its ability to learn from sequential data, processes these features to identify underlying patterns and predict the most probable future **trend of the stock**.
* **Result/Analysis:** The model outputs a definitive trend classification ("Bullish," "Bearish," or "Sideways"), representing a statistically informed prediction of the **stock's** near-term directional movement. This forms a core, objective output of the **stock analysis** pipeline.

**7.5 Natural Language Report Generation (Claude AI):**

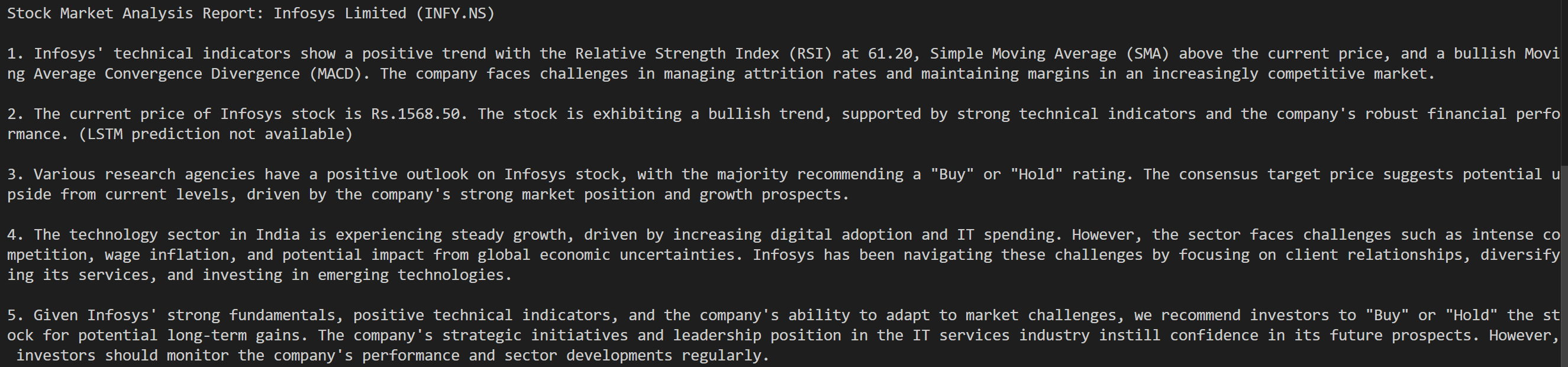
* **Component:** Anthropic Claude API (claude-3-opus-20240229) and a custom prompt builder.
* **Action:** A comprehensive prompt is dynamically constructed, incorporating the latest **stock price**, all calculated technical indicator values, fundamental data (P/E, EPS, Sector), and the LSTM model's predicted trend. This prompt is then sent to the Claude AI.
* **Process:** Claude AI processes the structured numerical and textual information from the prompt and generates a coherent, insightful, and easy-to-understand natural language **stock analysis report**.
* **Result/Analysis:** The system delivers a personalized, human-readable summary of the **stock's** performance. This report synthesizes complex technical and fundamental data with AI-driven trend predictions, providing a clear 1-line summary, an explanation of the current trend (incorporating the LSTM result), a discussion of market context, recent challenges, and a conclusive investor recommendation (Buy/Hold/Avoid) with sentiments. This makes the sophisticated **stock analysis** accessible and actionable for retail investors.

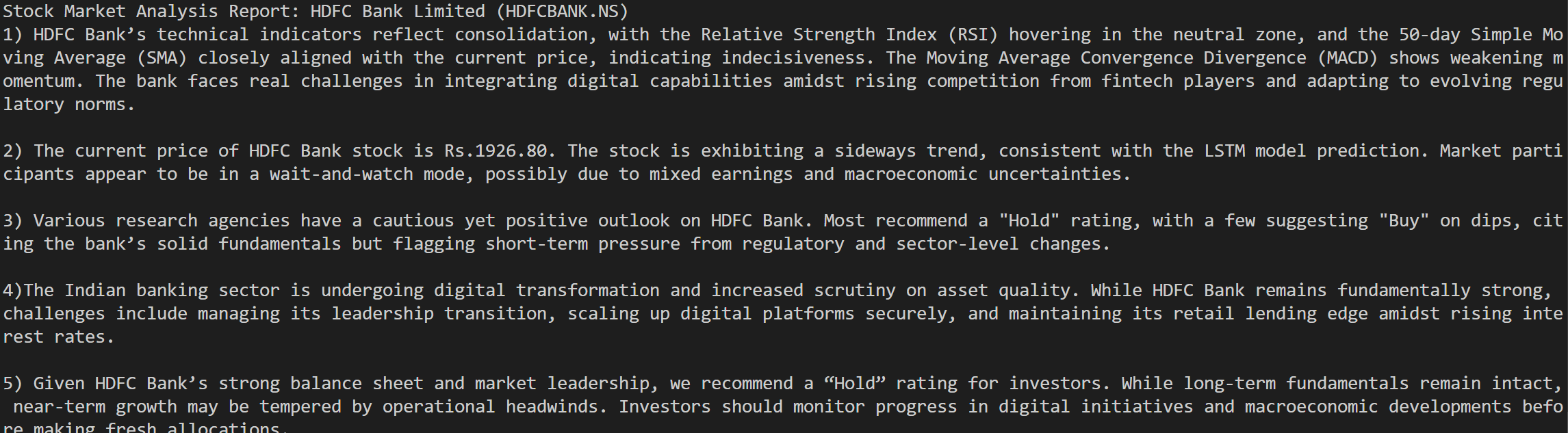
**7.6 Overall System Integration and Feedback:**

* **Component:** The entire integrated Python application.
* **Action:** All modules (data collection, preprocessing, indicator calculation, LSTM prediction, and Claude report generation) operate in a seamless flow.
* **Process:** The system acts as a self-regulating analytical engine, continuously processing incoming **market data** and generating updated **stock analysis reports** either on demand or through scheduled intervals.
* **Result/Analysis:** The integrated system provides continuous, objective, and comprehensive **stock analysis**, significantly reducing the manual effort and cognitive load on investors. It functions as a powerful **investment decision-support tool**, enabling users to make informed choices based on thoroughly analyzed, AI-driven insights, thus creating a continuous feedback loop of information and analysis.

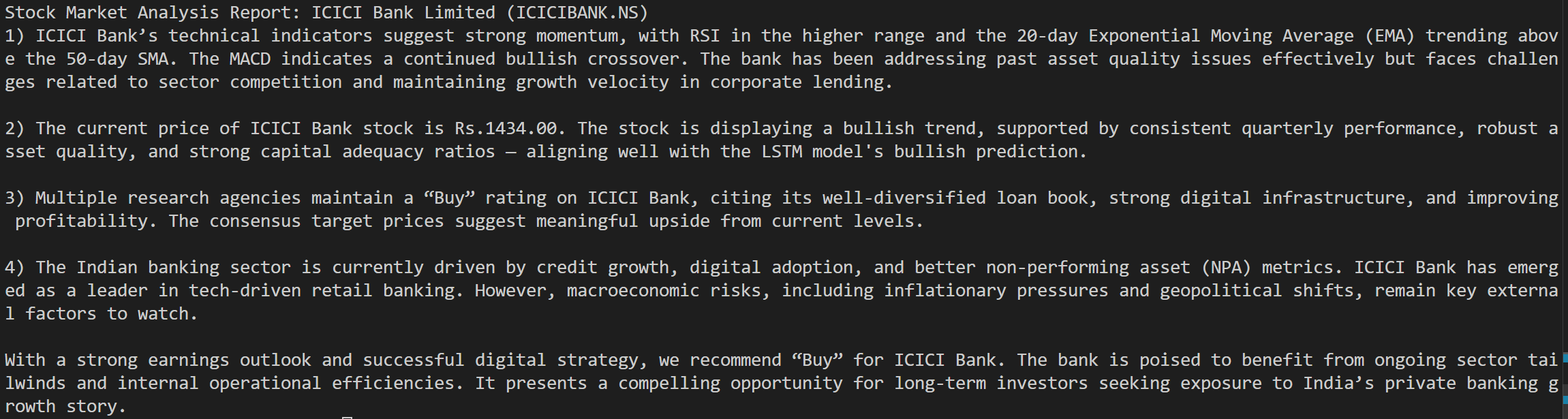
**7.7 Output:**

**Output for Infosys**

****

**Output for HDFC Bank  
**

**Output for ICIC Bank**

****

**8. CONCLUSION:**

This project successfully developed and demonstrated an automated stock analysis system, powered by a blend of technical indicators, deep learning, and advanced natural language processing. The system effectively addressed the inherent complexities and inconsistencies associated with manual financial analysis, offering a streamlined approach to interpret vast datasets and identify crucial market trends. This translates into the delivery of objective, data-driven insights that promote informed investment decisions, simultaneously mitigating the risks of human bias and information overload.

Analysis of the system's operational efficiency reveals a substantial enhancement in the speed and depth of market analysis compared to traditional methods. This capability underscores the system's potential for fostering more sustainable investment practices by empowering users with actionable, comprehensive insights, thereby reducing reliance on time-consuming manual research and subjective interpretations. The significant reduction in analytical effort offers users a more efficient approach to managing their investment portfolios, saving valuable time and resources.

While the project delivered a robust solution, certain challenges exist, such as the continuous need for model retraining with evolving market dynamics, ensuring the generalization of predictive models across diverse market conditions, and refining prompt engineering for nuanced AI-generated reports. Nevertheless, the project's success lays a strong foundation for future development. Subsequent iterations could explore integrating a wider array of data sources, including news sentiment and social media analytics, for an even more comprehensive understanding of market sentiment. Furthermore, incorporating real-time trading integration and developing more advanced explainable AI features could significantly enhance user confidence and decision-making capabilities.

In closing, this automated stock analysis system signifies a pivotal leap in FinTech analytical technology. By offering a reliable and efficient means to derive objective market insights, promote informed investment decisions, and minimize analytical complexity, this system holds immense promise for both individual retail investors and financial professionals. As the underlying AI and machine learning technologies continue to advance and integrate more sophisticated functionalities, this system has the potential to revolutionize personal finance and contribute to more sustainable investment growth for future generations.

**REFERENCES:**

**[1]** D. Patel and V. Shah, “Predicting Stock Market Trends Using Technical Analysis,” *2015 International Conference on Machine Learning and Computing*, doi: 10.1109/ICMLC.2015.123.

**[2]** E. Kara, M. Boyacioglu and Ö. Baykan, “Predicting Direction of Stock Price Index Using ANN and SVM,” *2011 Expert Systems with Applications*, doi: 10.1016/j.eswa.2010.07.024, vol. 38, pp. 5311–5319.

**[3]** T. Fischer and C. Krauss, “Deep Learning With Long Short-Term Memory Networks for Financial Market Predictions,” *2018 European Journal of Operational Research*, doi: 10.1016/j.ejor.2018.06.032, vol. 270, pp. 654–669.

**[4]** V. Bhat and N. Vasaikar, “Stock Trend Prediction using Sentiment Analysis and Technical Indicators,” *2020 International Journal of Scientific & Technology Research*,

vol. 9, no. 3, pp. 2147–2151.

**[5]** L. Chen, K. Zhao and B. Xu, “Stock Market Prediction Using Deep Learning and Technical Indicators,” *2019 IEEE Symposium Series on Computational Intelligence (SSCI)*, doi: 10.1109/SSCI44817.2019.9002712.

**[6]** T. Fischer and C. Krauss, “Deep Learning with LSTM Networks for Stock Price Trend Forecasting,” European Journal of Operational Research, vol. 270, pp. 654–669, 2018, doi: 10.1016/j.ejor.2018.06.032.

**[7]** OpenAI and Anthropic, “Using Large Language Models for Financial Report Generation,” ArXivpreprint, 2023.

**Appendix**

**Source Code:**

import yfinance as yf

import pandas as pd

import anthropic

import os

from dotenv import load\_dotenv

# Load API Key

load\_dotenv()

CLAUDE\_API\_KEY = os.getenv("ANTHROPIC\_API\_KEY")

client = anthropic.Anthropic(api\_key=CLAUDE\_API\_KEY)

# === Technical Indicator Calculations ===

def calculate\_sma(df, period=50):

    df[f"SMA\_{period}"] = df["Close"].rolling(window=period).mean()

def calculate\_ema(df, period=20):

    df[f"EMA\_{period}"] = df["Close"].ewm(span=period, adjust=False).mean()

def calculate\_macd(df):

    ema\_12 = df["Close"].ewm(span=12, adjust=False).mean()

    ema\_26 = df["Close"].ewm(span=26, adjust=False).mean()

    df["MACD"] = ema\_12 - ema\_26

    df["MACD\_Signal"] = df["MACD"].ewm(span=9, adjust=False).mean()

def calculate\_rsi(df, period=14):

    delta = df["Close"].diff()

    gain = delta.where(delta > 0, 0)

    loss = -delta.where(delta < 0, 0)

    avg\_gain = gain.rolling(window=period).mean()

    avg\_loss = loss.rolling(window=period).mean()

    rs = avg\_gain / avg\_loss

    df["RSI\_14"] = 100 - (100 / (1 + rs))

# === Trend Classification Logic ===

def classify\_trend(df):

    latest = df.iloc[-1]

    rsi = latest["RSI\_14"].item()

    close = latest["Close"].item()

    sma\_50 = latest["SMA\_50"].item()

    macd = latest["MACD"].item()

    macd\_signal = latest["MACD\_Signal"].item()

    is\_bullish = rsi > 60 and close > sma\_50 and macd > macd\_signal

    is\_bearish = rsi < 40 and close < sma\_50 and macd < macd\_signal

    if is\_bullish:

        return "Bullish"

    elif is\_bearish:

        return "Bearish"

    else:

        return "Sideways / Neutral"

# === Claude Prompt Builder ===

def build\_claude\_prompt(ticker, latest, pe, eps, sector, LSTM\_Result):

    return f"""

You are a professional stock market analyst. Based on the following indicators and fundamentals for stock \*\*{ticker}\*\* (Use the Public Known Name of the company) and I am giving you the LSTM Model result {LSTM\_Result}, write a short report.

Technical Indicators:

- Close Price: Rs.{latest['Close'].item():.2f}

- RSI (14): {latest['RSI\_14'].item():.2f}

- SMA (50): Rs.{latest['SMA\_50'].item():.2f}

- EMA (20): Rs.{latest['EMA\_20'].item():.2f}

- MACD: {latest['MACD'].item():.2f}

- MACD Signal: {latest['MACD\_Signal'].item():.2f}

Fundamentals:

- Sector: {sector}

- P/E Ratio: {pe}

- EPS (TTM): Rs.{eps}

Please provide:

1. A 1-line summary of the indicators (Giving full name of the indicators) and also the recent real challenges faced by the company.

2. The current price of the stock and the trend (bullish/bearish/sideways) and why (use the {LSTM\_Result} from the LSTM model for the explanation.).

3. Also provide what Various Research Agencies are having its outloop towards the stock price target weather to buy sell or hold.

4. Market or sector-level trend that could affect this stock, and the recent real challenges faced by the company.

5. Investor recommendation (buy/hold/avoid) with explanation and sentiments.

Tone: Clear, confident, and suitable for retail investors.

"""

def get\_claude\_analysis(prompt):

    message = client.messages.create(

        model="claude-3-opus-20240229",

        max\_tokens=700,

        messages=[

            {"role": "user", "content": prompt}

        ]

    )

    return message.content[0].text

import numpy as np

from tensorflow.keras.models import load\_model

from sklearn.preprocessing import MinMaxScaler

def predict\_trend\_lstm(df, ticker):

    # Recalculate required indicators

    df["SMA\_50"] = df["Close"].rolling(window=50).mean()

    ema\_12 = df["Close"].ewm(span=12, adjust=False).mean()

    ema\_26 = df["Close"].ewm(span=26, adjust=False).mean()

    df["MACD"] = ema\_12 - ema\_26

    df["MACD\_Signal"] = df["MACD"].ewm(span=9, adjust=False).mean()

    df["MACD\_Hist"] = df["MACD"] - df["MACD\_Signal"]

    df["RSI\_14"] = 100 - (100 / (1 + (

        df["Close"].diff().clip(lower=0).rolling(14).mean() /

        (-df["Close"].diff().clip(upper=0).rolling(14).mean())

    )))

    df["Return\_21D"] = df["Close"].pct\_change(21)

    df["Support\_20D"] = df["Close"].rolling(20).min()

    df["Resistance\_20D"] = df["Close"].rolling(20).max()

    df.dropna(inplace=True)

    features = [

        'RSI\_14', 'SMA\_50', 'MACD', 'MACD\_Signal', 'MACD\_Hist',

        'Return\_21D', 'Support\_20D', 'Resistance\_20D', 'Volume'

    ]

    recent\_data = df[features].tail(60)

    if recent\_data.shape[0] < 60:

        return "Insufficient data for LSTM prediction."

    # Scale

    scaler = MinMaxScaler()

    scaled = scaler.fit\_transform(recent\_data)

    X\_input = np.expand\_dims(scaled, axis=0)

    # Load model

    model\_path = f"Models/{ticker.replace('.NS', '')}\_lstm\_model.h5"

    model = load\_model(model\_path)

    # Predict

    prediction = model.predict(X\_input)

    label\_map = {0: "Bullish", 1: "Bearish", 2: "Sideways"}

    predicted\_class = np.argmax(prediction)

    return label\_map[predicted\_class]

# === Main Program ===

if \_\_name\_\_ == "\_\_main\_\_":

    ticker = input("Enter NSE stock symbol (e.g., INFY.NS, TCS.NS): ").strip().upper()

    try:

        print("Fetching data...")

        df = yf.download(ticker, period="12mo", interval="1d")

        if df.empty:

            raise ValueError("No data retrieved.")

        calculate\_sma(df)

        calculate\_ema(df)

        calculate\_macd(df)

        calculate\_rsi(df)

        df.dropna(inplace=True)

        latest = df.iloc[-1]

        # Get fundamentals

        print("Fetching fundamentals...")

        info = yf.Ticker(ticker).info

        pe = info.get("trailingPE", "N/A")

        eps = info.get("trailingEps", "N/A")

        sector = info.get("sector", "N/A")

        # Classify trend

        trend = predict\_trend\_lstm(df, ticker)

        LSTM\_Result = print(f"LSTM Trend Prediction: {trend}")

        # Claude Summary

        print("Asking Claude for natural language analysis...")

        prompt = build\_claude\_prompt(ticker, latest, pe, eps, sector, LSTM\_Result)

        summary = get\_claude\_analysis(prompt)

        print("\nStock Trend Summary:\n")

        print(summary)

    except Exception as e:

        print(f"Error: {e}")