Stock Price Analysis and Automated Insight Generation Using Technical Indicators

By – Vasav Srivastava and Utkarsh Bhagat

Introduction

Stock market prediction has attracted extensive attention from investors, researchers, and financial analysts. While machine learning and deep learning have gained popularity recently, technical indicators like Simple Moving Average (SMA), Relative Strength Index (RSI), and Moving Average Convergence Divergence (MACD) have been traditionally used to support trading decisions.

This project focuses on developing an automated system to fetch real-world stock data, calculate major technical indicators, visualize patterns, and generate financial insights using a Large Language Model (LLM).

Research Question: Can the integration of traditional technical indicators with automated language models accelerate and enhance stock analysis for investors?

Significance: Automating technical analysis democratizes financial tools for non-expert investors and saves significant time for professional analysts, supporting better decision-making.

Related Work

Technical indicators have been foundational tools for decades. Key literature highlights include:

- SMA (Simple Moving Average): Used to smooth price data, identify trends, and generate trading signals (Murphy, Technical Analysis of the Financial Markets).
- RSI (Relative Strength Index): Introduced by J. Welles Wilder in 1978, RSI measures the speed and change of price movements to identify overbought or oversold conditions.
- MACD (Moving Average Convergence Divergence): Developed by Gerald Appel, MACD detects momentum changes by comparing short-term and long-term EMAs.

Studies such as Patel et al. (2015) explored machine learning approaches but emphasized that hybrid models combining technical analysis and AI yield better results.

Gap Identified: Traditional technical analysis is manual, slow, and subjective. There is a need for automated interpretation to improve speed, consistency, and accessibility.

Proposed Methodology

The method involves two main parts:

1. Numerical Calculation of Indicators using Python:
    - SMA: SMA_n(t) = (P(t) + P(t-1) + ... + P(t-n+1)) / n
    - RSI: RSI = 100 - (100 / (1 + RS)) where RS = (Average Gain) / (Average Loss)
    - MACD: MACD = EMA_short(t) - EMA_long(t) Signal Line = EMA_MACD(9 periods)
2. Automated Insight Generation:
    - Use EleutherAI/gpt-neo-2.7B from Hugging Face to interpret the indicators and produce short financial analysis.

Tools & Libraries:

- yfinance for fetching data
- pandas, numpy for calculations
- matplotlib for visualization
- transformers for text generation

Install Necessary Libraries

```
!pip install yfinance pandas numpy matplotlib transformers
```

Import Libraries

```
import yfinance as yf
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from transformers import pipeline
```

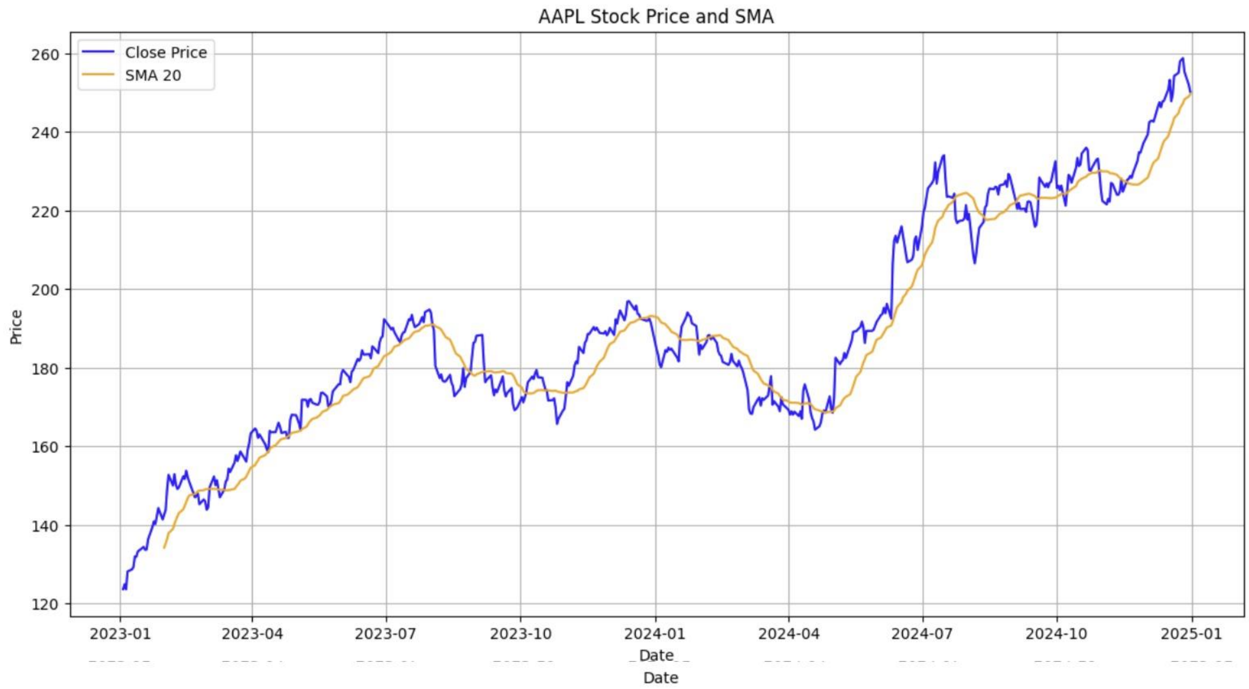
Function Definition

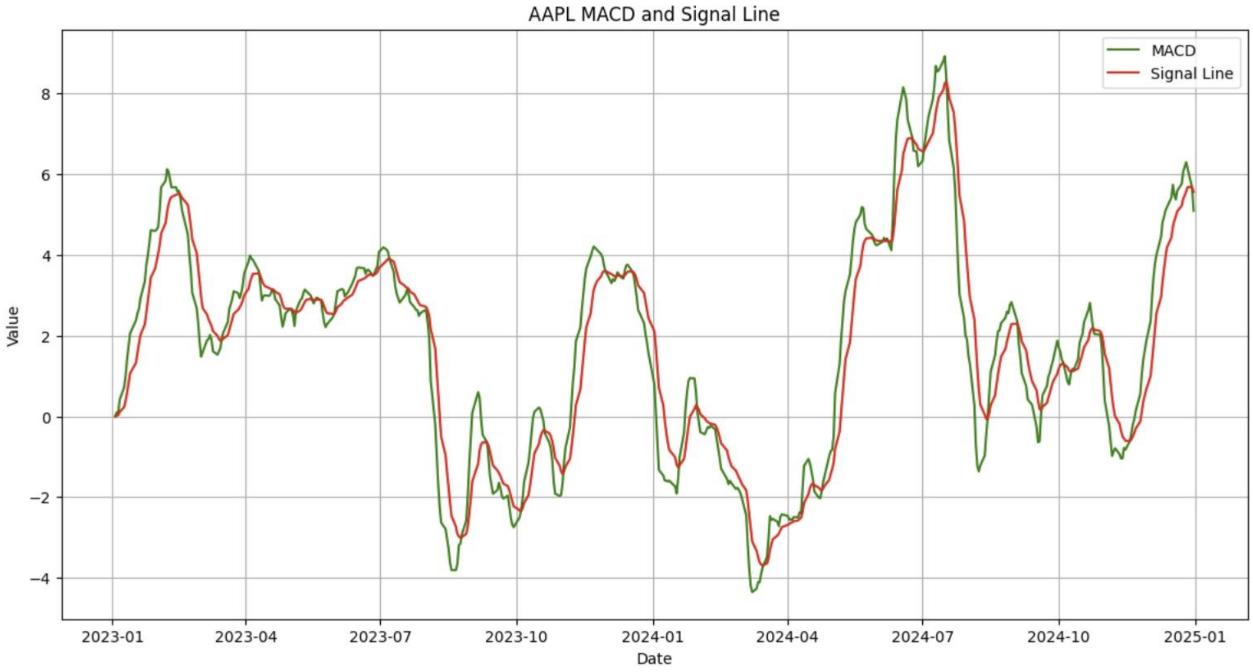
```
# Function to calculate RSI
def calculate_rsi(data, window=14):
    delta = data.diff(1)
    gain = delta.where(delta > 0, 0)
    loss = -delta.where(delta < 0, 0)

    avg_gain = gain.rolling(window=window, min_periods=1).mean()
    avg_loss = loss.rolling(window=window, min_periods=1).mean()

    rs = avg_gain / avg_loss
    rsi = 100 - (100 / (1 + rs))
    return rsi

# Function to calculate MACD
def calculate_macd(data, short_window=12, long_window=26, signal_window=9)
:
    short_ema = data.ewm(span=short_window, adjust=False).mean()
    long_ema = data.ewm(span=long_window, adjust=False).mean()
    macd = short_ema - long_ema
    signal = macd.ewm(span=signal_window, adjust=False).mean()
    return macd, signal

# Fetch stock data
def fetch_stock_data(ticker, start_date, end_date):
    data = yf.download(ticker, start=start_date, end=end_date)
    return data

# Analyze a single stock
def analyze_stock(ticker, start_date, end_date, llm):
    data = fetch_stock_data(ticker, start_date, end_date)

    data['SMA_20'] = data['Close'].rolling(window=20).mean()
    data['RSI'] = calculate_rsi(data['Close'])
    data['MACD'], data['Signal_Line'] = calculate_macd(data['Close'])

    plt.figure(figsize=(14, 7))
    plt.plot(data['Close'], label='Close Price', color='blue')
    plt.plot(data['SMA_20'], label='SMA 20', color='orange')
    plt.title(f'{ticker} Stock Price and SMA')
    plt.xlabel('Date')
    plt.ylabel('Price')
    plt.legend()
    plt.grid()
    plt.show()

    plt.figure(figsize=(14, 7))
    plt.plot(data['MACD'], label='MACD', color='green')
    plt.plot(data['Signal_Line'], label='Signal Line', color='red')
    plt.title(f'{ticker} MACD and Signal Line')
    plt.xlabel('Date')
    plt.ylabel('Value')
    plt.legend()
    plt.grid()
    plt.show()

    plt.figure(figsize=(14, 7))
    plt.plot(data['RSI'], label='RSI', color='purple')
    plt.axhline(70, color='red', linestyle='--', label='Overbought (70)')
    plt.axhline(30, color='green', linestyle='--', label='Oversold (30)')
    plt.title(f'{ticker} RSI')
    plt.xlabel('Date')
    plt.ylabel('RSI Value')
    plt.legend()
    plt.grid()
    plt.show()

    rsi_value = data['RSI'].iloc[-1]
    macd_value = data['MACD'].iloc[-1]
    signal_value = data['Signal_Line'].iloc[-1]

    prompt = (
        f"The stock's RSI is {rsi_value:.2f}, the MACD is {macd_value:.2f}, and the Signal Line is {signal_value:.2f}. "
        "Based on these indicators, provide a detailed financial analysis and recommendation for this stock."
    )
    insights = llm(prompt, max_length=100, truncation=True)
    generated_text = insights[0]['generated_text']

    cleaned_text = generated_text.split("recommendation")[0] + "recommendation."
    print(f"\nGenerated Insights for {ticker}:")
    print(cleaned_text)
```

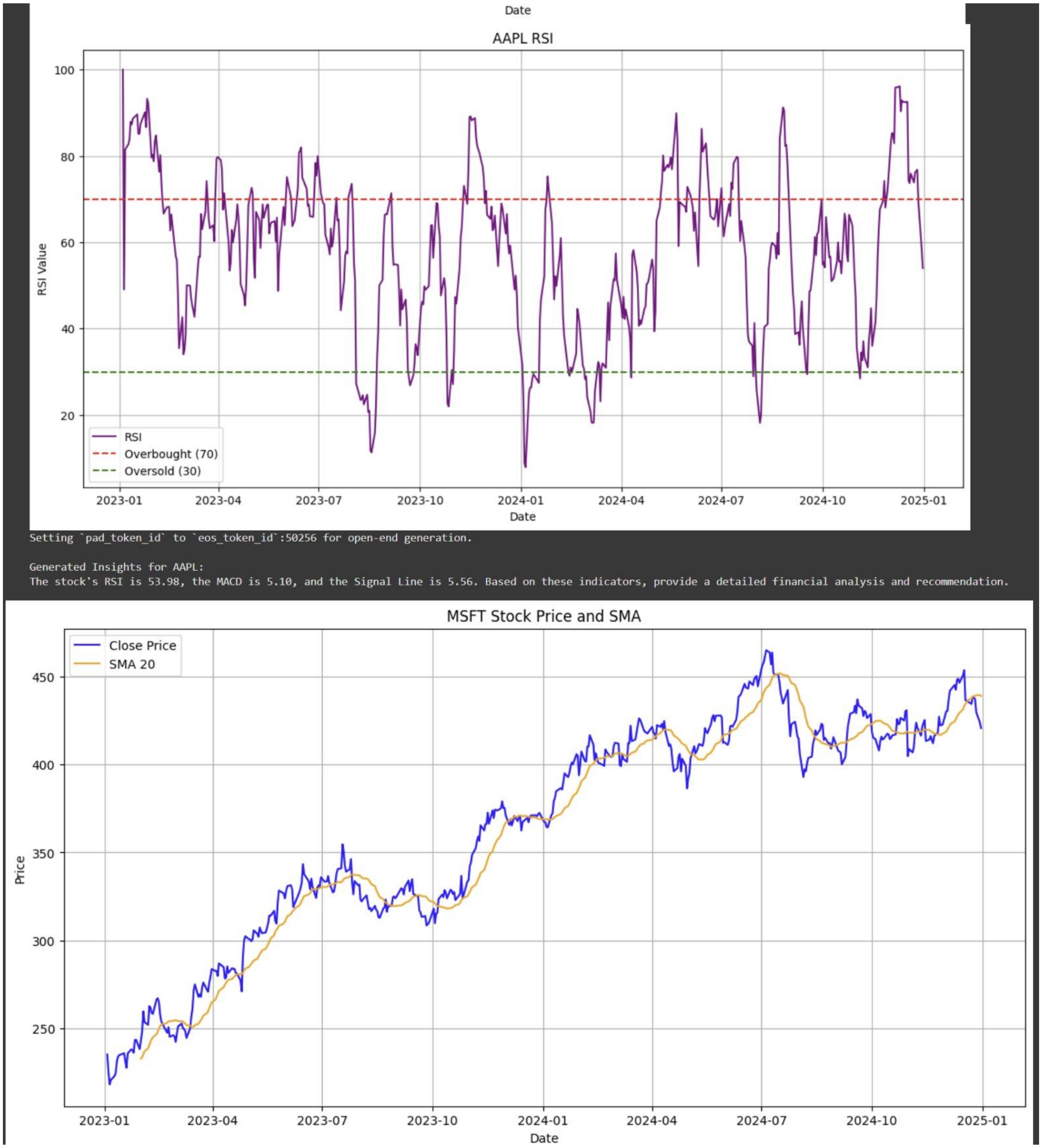
Main Function

```
def analyze_top_nasdaq_stocks():
    nasdaq_stocks = ["AAPL", "MSFT", "GOOGL", "AMZN", "NVDA", "TSLA", "META", "PEP", "AVGO", "COST"]
    start_date = "2023-01-01"
    end_date = "2025-01-01"

    llm = pipeline("text-generation", model="EleutherAI/gpt-neo-2.7B")

    for ticker in nasdaq_stocks:
        print(f"\nAnalyzing {ticker}...")
        analyze_stock(ticker, start_date, end_date, llm)

analyze_top_nasdaq_stocks()
```

Experiment Setup and Result Discussion

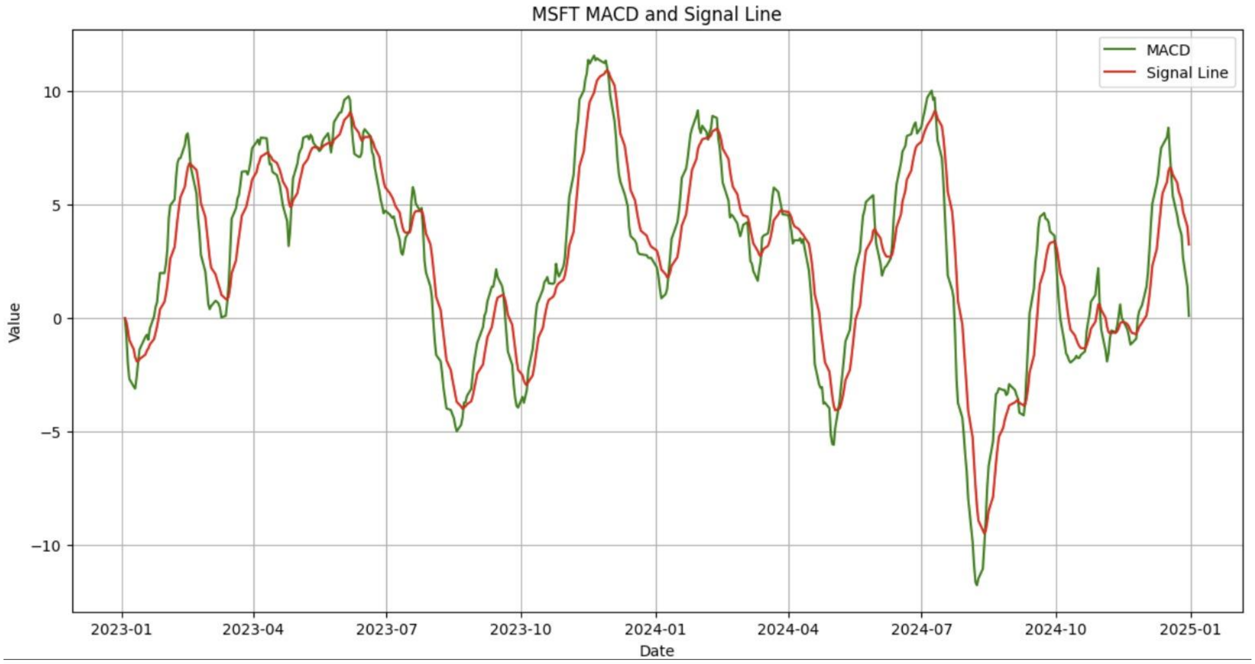

AAPL Stock Price and SMA
Close Price
SMA 20
Price
260
240
220
200
180
160
140
120
2023-01
2023-04
2023-07
2023-10
2024-01
2024-04
2024-07
2024-10
2025-01
Date
Date

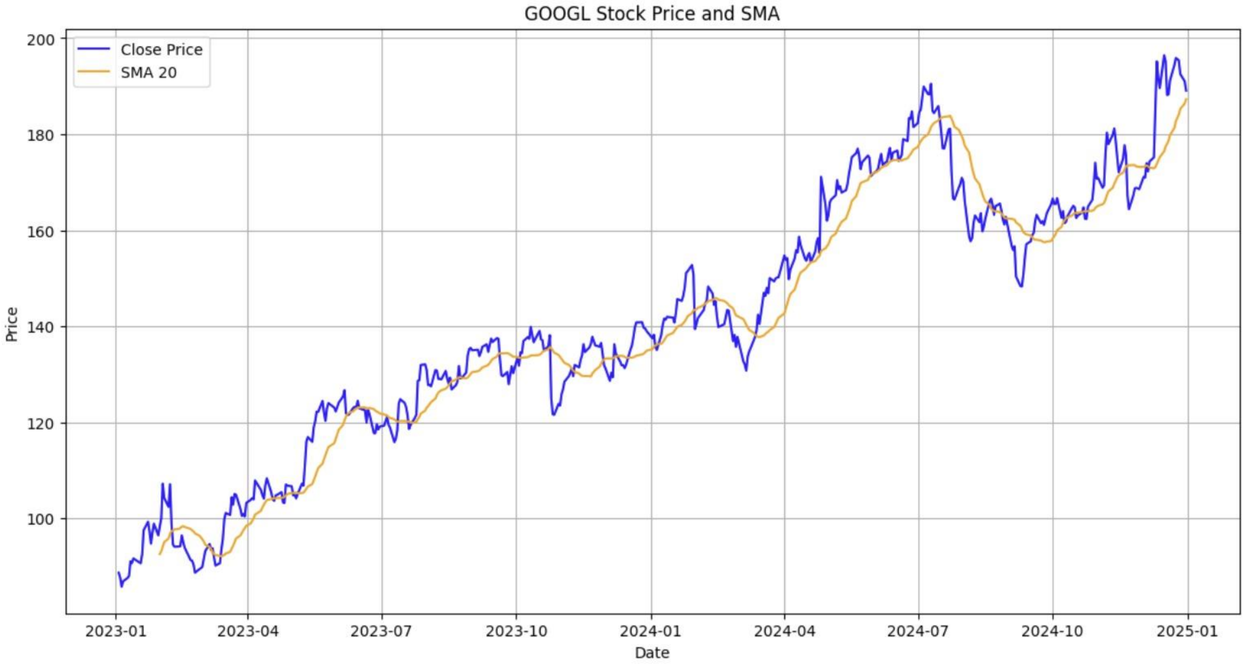

AAPL MACD and Signal Line
MACD
Signal Line
Value
8
6
4
2
0
−2
−4
2023-01
2023-04
2023-07
2023-10
2024-01
2024-04
2024-07
2024-10
2025-01
Date

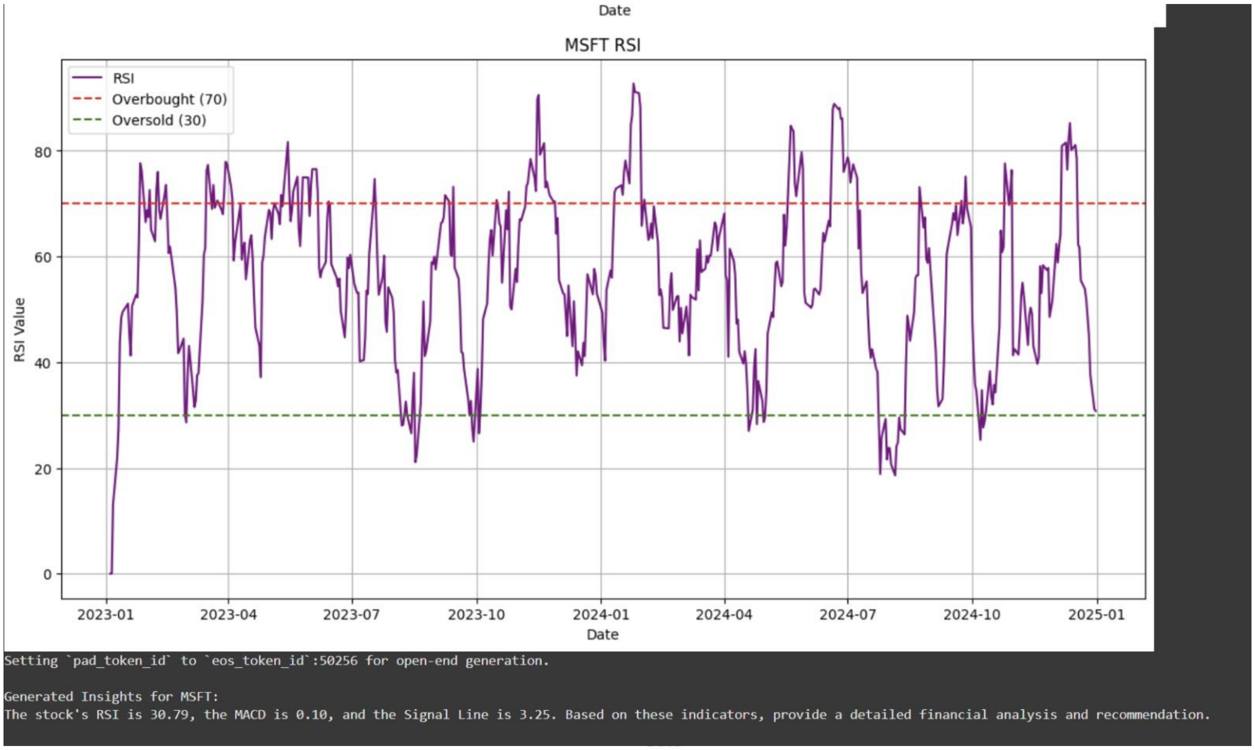


```
Setting `pad_token_id` to `eos_token_id`:50256 for open-end generation.

Generated Insights for AAPL:
The stock's RSI is 53.98, the MACD is 5.10, and the Signal Line is 5.56. Based on these indicators, provide a detailed financial analysis and recommendation.
```

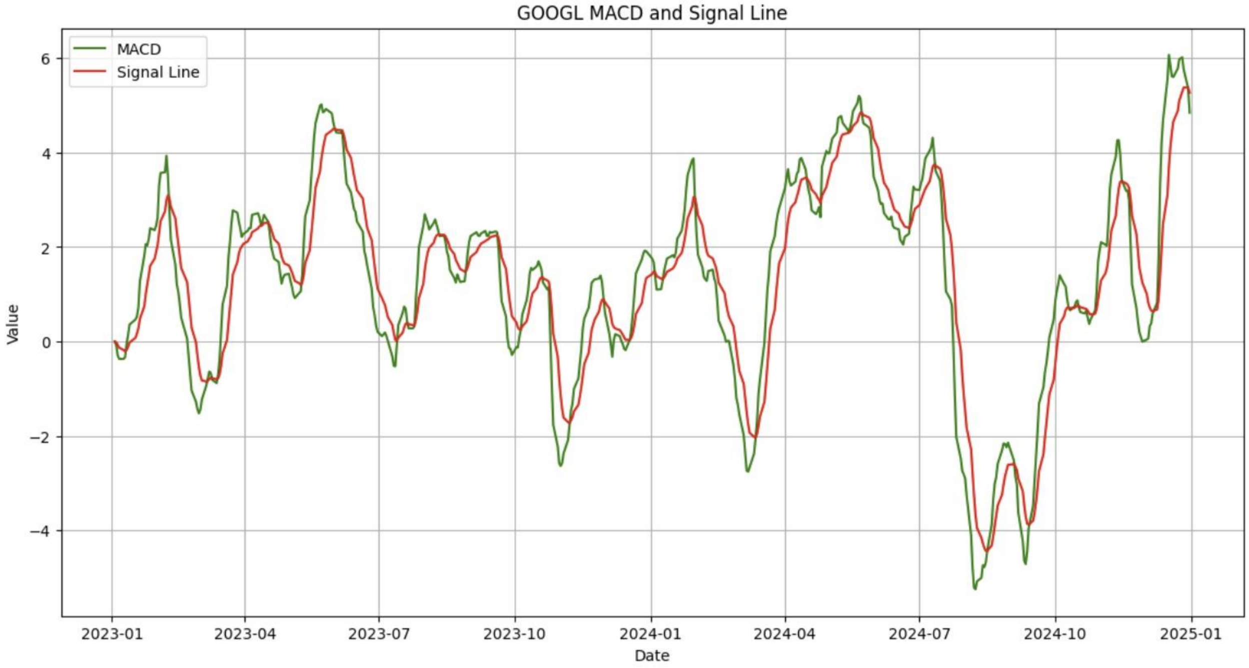
Date

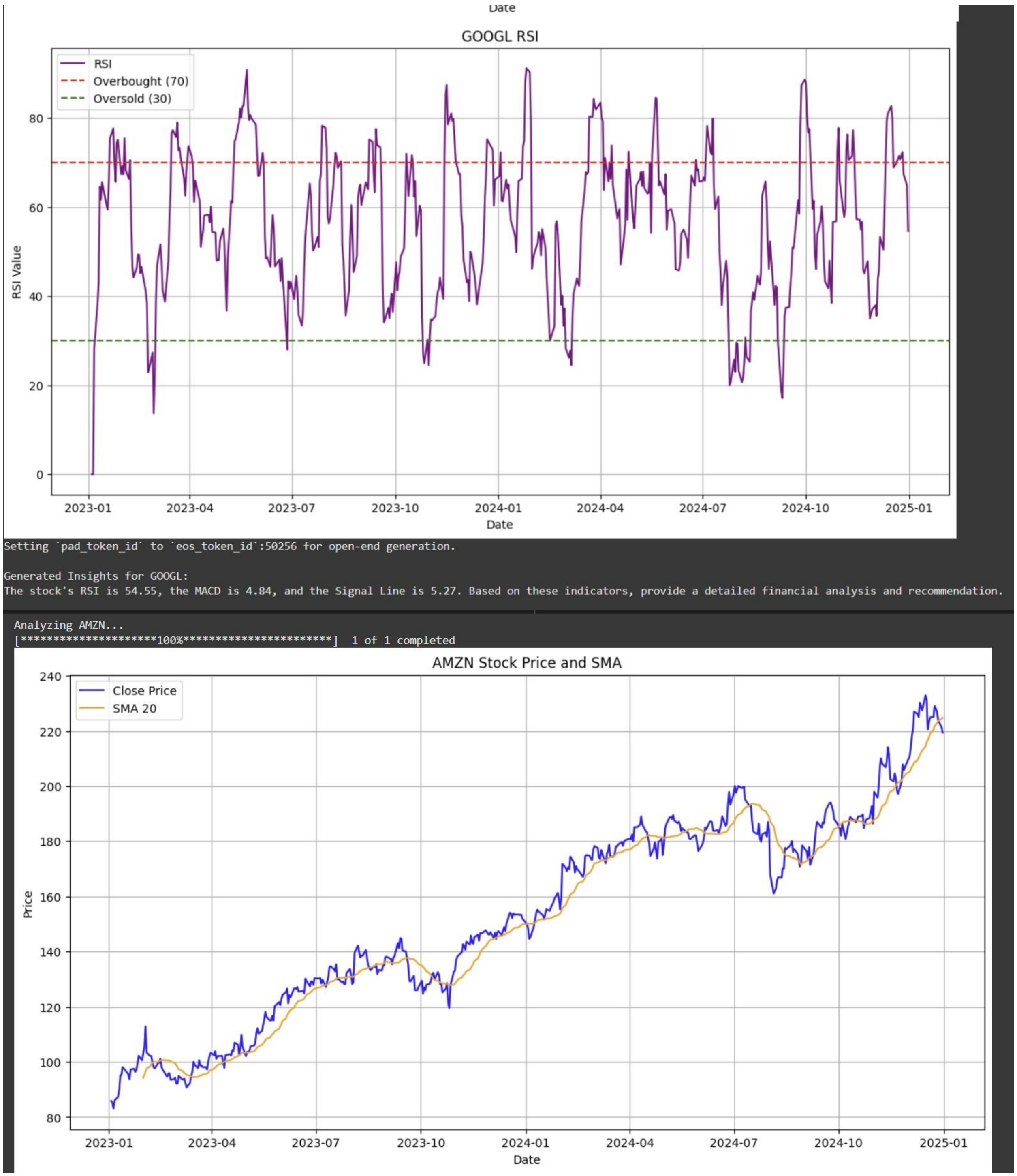

MSFT MACD and Signal Line


GOOGL Stock Price and SMA



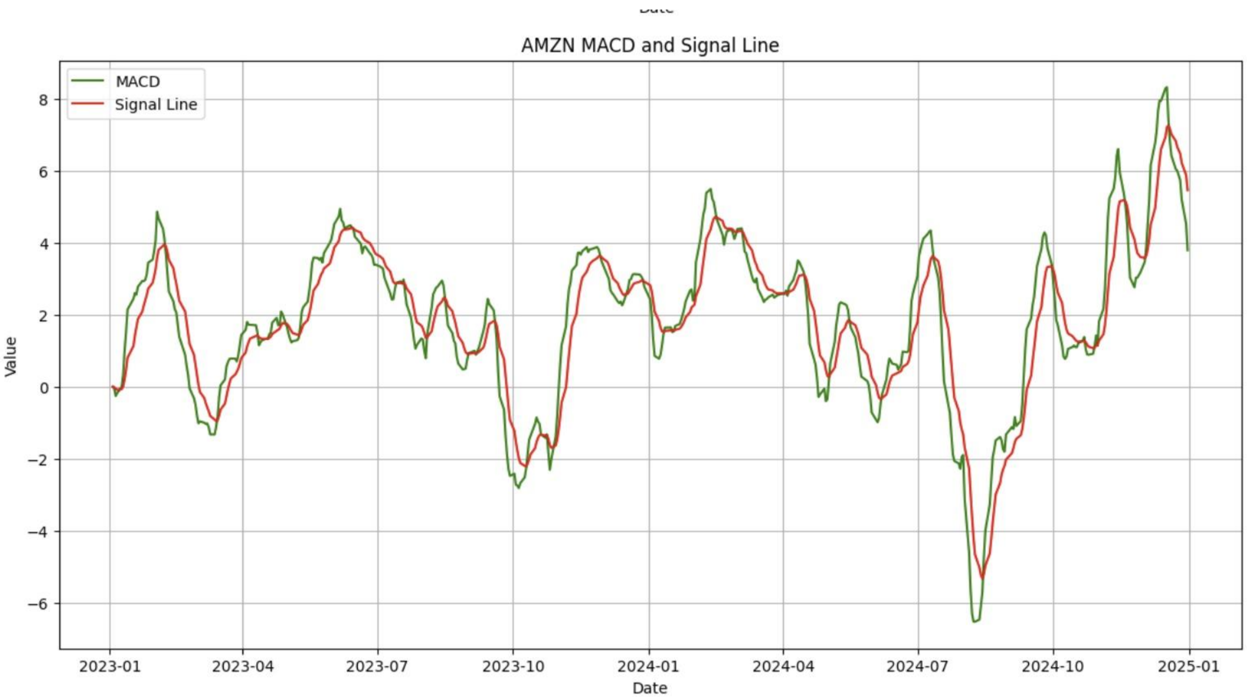
```
Setting `pad_token_id` to `eos_token_id`:50256 for open-end generation.

Generated Insights for MSFT:
The stock's RSI is 30.79, the MACD is 0.10, and the Signal Line is 3.25. Based on these indicators, provide a detailed financial analysis and recommendation.
```

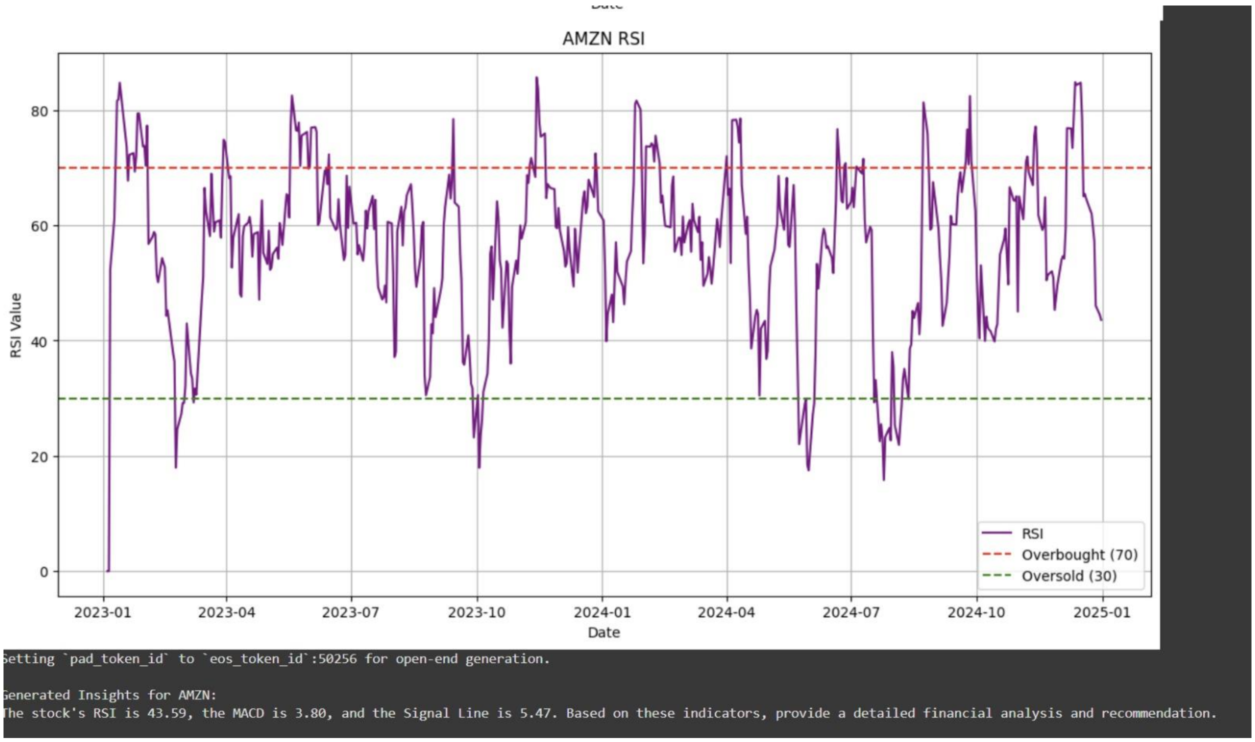




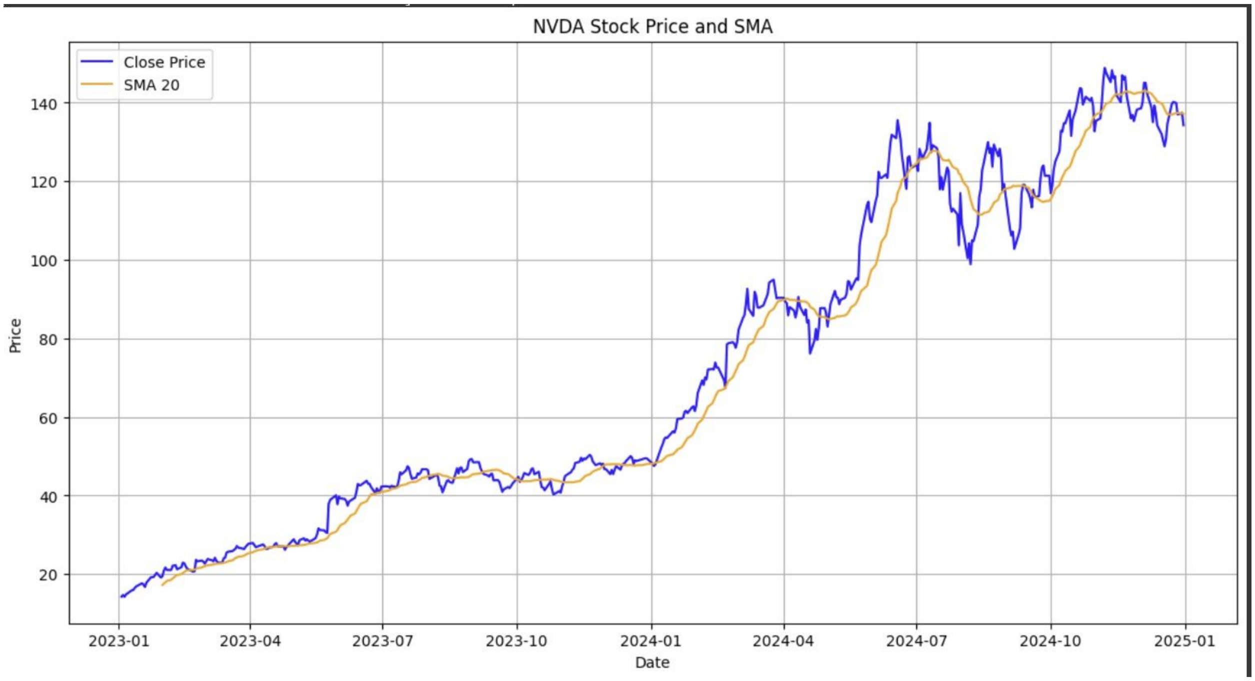
```
Setting `pad_token_id` to `eos_token_id`:50256 for open-end generation.

Generated Insights for GOOGL:
The stock's RSI is 54.55, the MACD is 4.84, and the Signal Line is 5.27. Based on these indicators, provide a detailed financial analysis and recommendation.
```

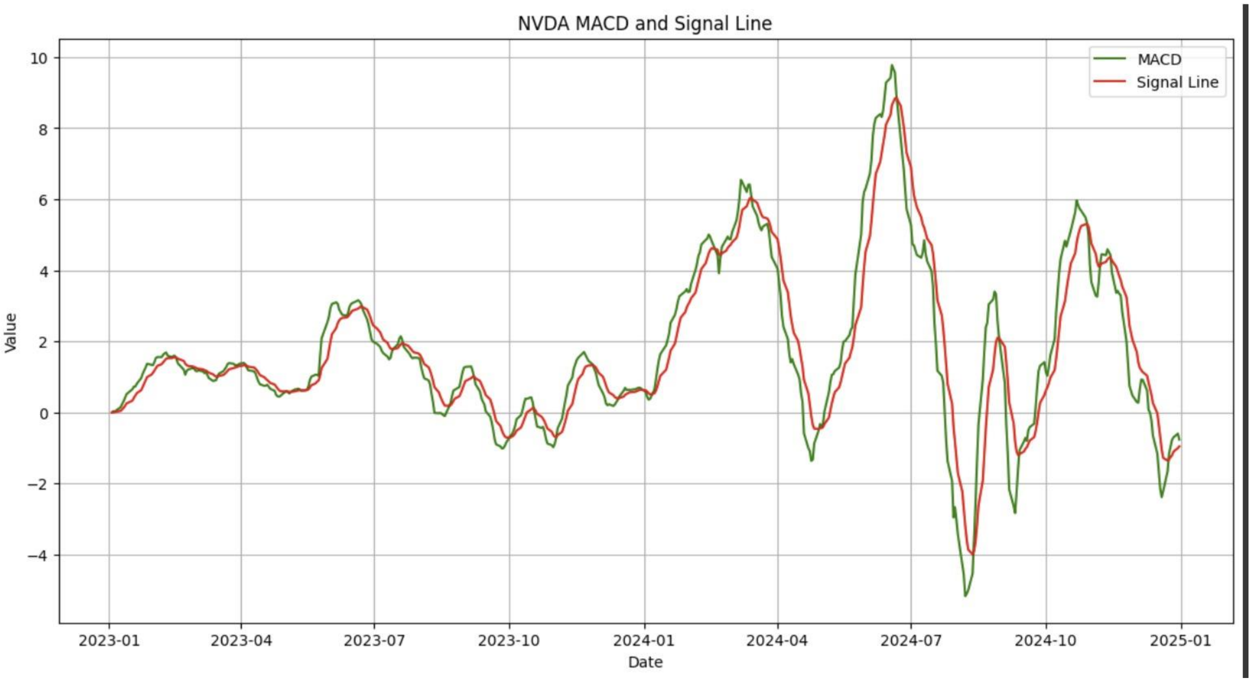
```
 Analyzing AMZN...
 [*********************100%***********************]  1 of 1 completed
```

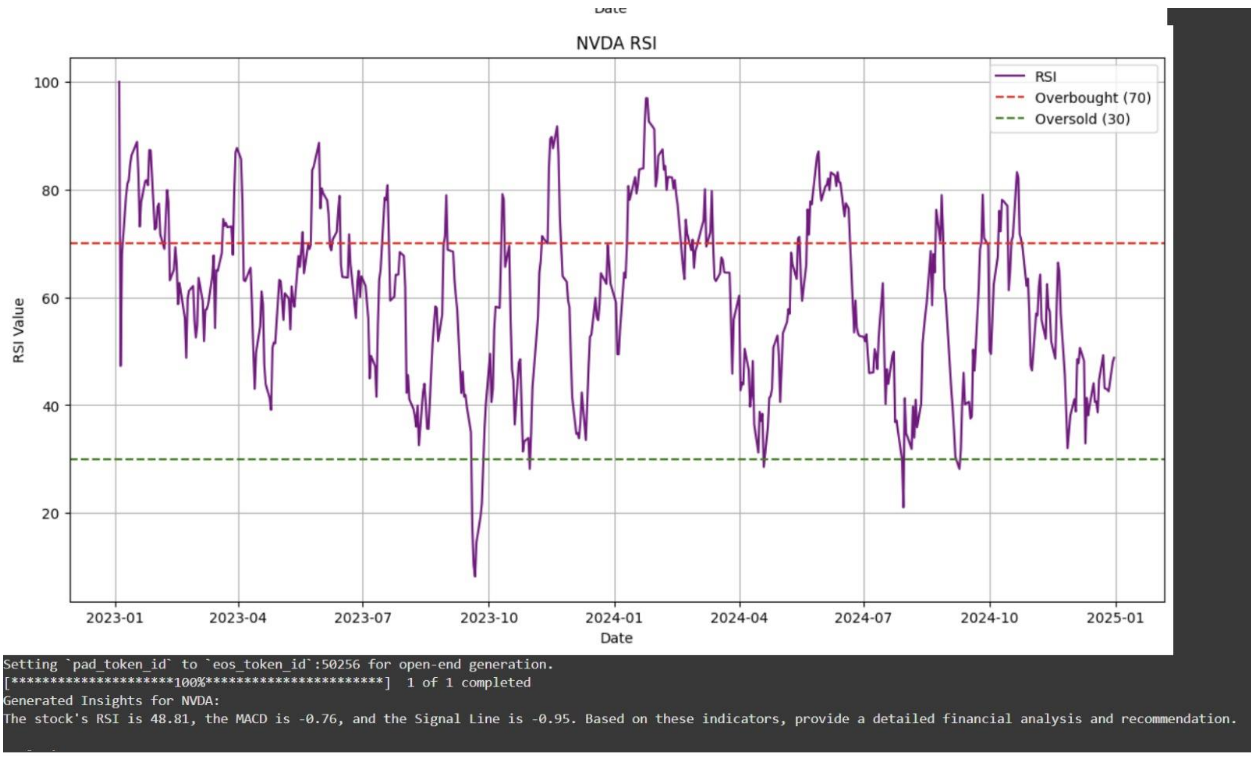




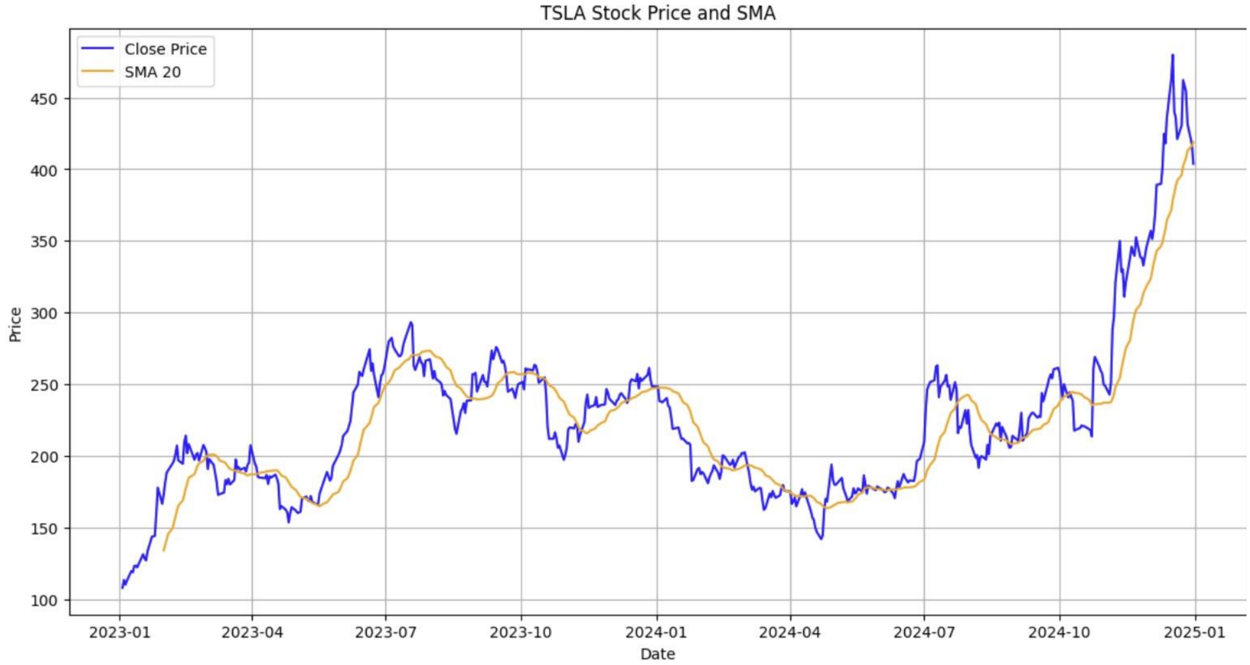
```
Setting `pad_token_id` to `eos_token_id`:50256 for open-end generation.

Generated Insights for AMZN:
The stock's RSI is 43.59, the MACD is 3.80, and the Signal Line is 5.47. Based on these indicators, provide a detailed financial analysis and recommendation.
```

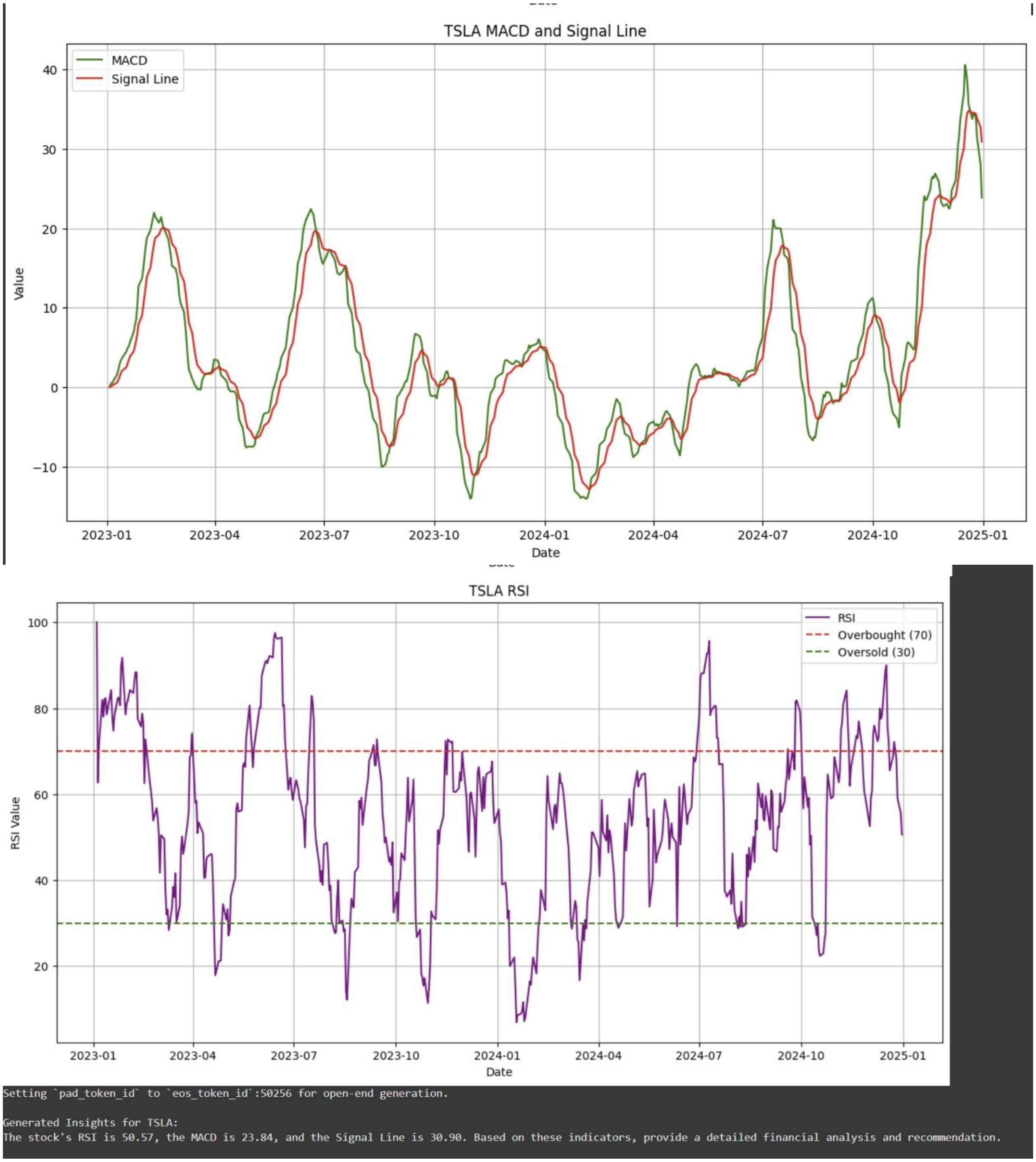

NVDA Stock Price and SMA
Close Price
SMA 20
Price
140
120
100
80
60
40
20
Date
2023-01
2023-04
2023-07
2023-10
2024-01
2024-04
2024-07
2024-10
2025-01


NVDA MACD and Signal Line
MACD
Signal Line
Value
10
8
6
4
2
0
−2
−4
Date
2023-01
2023-04
2023-07
2023-10
2024-01
2024-04
2024-07
2024-10
2025-01



```
Setting `pad_token_id` to `eos_token_id`:50256 for open-end generation.
[*********************100%***********************]  1 of 1 completed
Generated Insights for NVDA:
The stock's RSI is 48.81, the MACD is -0.76, and the Signal Line is -0.95. Based on these indicators, provide a detailed financial analysis and recommendation.
```





```
Setting `pad_token_id` to `eos_token_id`:50256 for open-end generation.

Generated Insights for TSLA:
The stock's RSI is 50.57, the MACD is 23.84, and the Signal Line is 30.90. Based on these indicators, provide a detailed financial analysis and recommendation.
```

Datasets:

- Stocks: AAPL, MSFT, GOOGL, AMZN, NVDA, TSLA, META, PEP, AVGO, COST
- Source: Yahoo Finance
- Date Range: January 1, 2023 – January 1, 2025

Observations:

- Visualizations of each stock.
- Insights generated through LLM.
- Identified overbought, oversold, and crossover signals.

Comparison

Manual vs Automated Analysis:

- Manual analysis is slow and subjective.
- Automated insights are quicker, more consistent.

Comparison with past studies:

- Matches traditional methods in simplicity.
- Offers instant, low-bias interpretations.

Conclusion

Combining traditional technical indicators with LLMs creates a powerful, fast, and accessible stock analysis tool.

Future Work:

- Include fundamental data analysis.
- Fine-tune LLMs on financial datasets.
- Expand to commodities and forex markets.

References

- Murphy, J. J. (1999). Technical Analysis of the Financial Markets.
- Wilder, J. W. (1978). New Concepts in Technical Trading Systems.
- Patel et al. (2015). Predicting stock and stock price index movement.
- Investopedia on SMA, RSI, MACD.
- HuggingFace Transformers documentation.
- Yahoo Finance via yfinance API.