Stockseer: Lightweight Stock Price Forecasting via Linear Regression and yfinance

Thomas E. Seife, Eduardo Jacinto, Alex Canizares
School of Computing, University of North Florida

Abstract

Stockseer is a full-stack application for real-time stock price prediction and visualization. It employs a Python FastAPI backend with a linear regression model trained on two years of historical closing data via yfinance, and a dynamic Next.js frontend styled with Tailwind and radix UI. Users can search company tickers, view live market quotes, and interact with predictive visualizations where chart colors reflect directional trends. This report outlines Stockseer's architecture, methodology, and evaluation. The results affirm that simple models, when engineered carefully and paired with intuitive UI/UX design, can yield useful insights with low latency.

Introduction

Stock market forecasting remains a fundamental challenge in both academic finance and practical investing. The promise of predicting future prices is particularly appealing for short-term traders and algorithmic systems. Although recurrent neural networks (e.g., LSTM (fischer2018lstm) and attention-based architectures (e.g., Transformer models (wu-2023-transformer) have set high benchmarks, they remain resource-heavy and often unsuitable for real-time, lightweight applications.

Linear regression, a classic statistical approach, offers an interpretable and computationally inexpensive alternative. As discussed in Investopedia (investopedia Linear Vs Multi), linear regression models the relationship between a dependent variable and one or more independent variables and is widely taught in introductory courses due to its clarity.

Stockseer aims to demonstrate that linear regression, with effective feature preparation and a fresh data pipeline, can yield competitively useful results in stock price forecasting for retail and educational settings.

Related Work

The use of linear regression in time series analysis has been explored extensively. Recent works (lrStockStudy, elr2023) show that linear regression, when enhanced with features like rolling statistics, continues to perform well even when compared to more advanced machine learning models. Hybrid models like Linear Regression–LSTM (acmHybridLRLSTM) further validate this.

Interactive platforms like Kaggle (kaggleLR) have shown the educational value of implementing LR for stock predictions, while conferences like ITM highlight how it can be used to construct entire websites. The yfinance Python library (yfinanceDocs)—developed as a wrapper over Yahoo Finance—is frequently cited as a go-to tool for live and historical stock market data, with strong developer adoption evidenced through PyPI statistics (yfinancePyPI) and community blogs (rowzeroBlog).

Methodology

Data Pipeline

Stockseer retrieves stock data using the yfinance library (yfinanceDocs), which provides daily historical data in JSON format. Each record includes Open, High, Low, Close, and Volume (OHLCV). For forecasting, only the ‘Close‘ and ‘Date‘ are used.

The ordinal representation of ‘Date‘ serves as the independent variable (x_t), while ‘Close‘ is the dependent variable (p_t). The API fetch is triggered via a nightly GitHub Action to maintain freshness without exhausting rate limits.

Modeling

A linear regression model is fitted to the relationship:

$$p_t = \beta_0 + \beta_1 x_t + \varepsilon_t, \qquad (1)$$

where β_0 and β_1 are computed using scikit-learn's `LinearRegression()` class. The model is retrained every 24 hours to account for market drift and new data.

Architecture

The backend is a FastAPI server that exposes four routes:

- `/symbol-search`: Company name → ticker
- `/quote`: Live price data from Twelve Data
- `/ohlc`: Today's OHLC snapshot
- `/predict`: Prediction for the next closing price

The frontend (Next.js 14) performs parallel fetches to the backend routes and renders:

1. Quote card with price
2. OHLC table
3. Chart overlay (Recharts) showing actual vs predicted

Visual feedback includes red/green dynamic gradients that adapt based on whether $\hat{p}_{t+1} > p_t$.

Evaluation

Test Setup

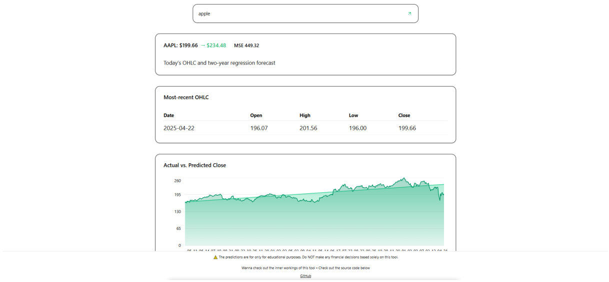
The system was benchmarked using the highest market cap stocks from the S&P 500 index over the two-year period from April 2023 to April 2025. MAE and RMSE are used to assess prediction accuracy, while inference latency is logged for user interaction timing.

Model	MAE ($)	RMSE ($)	Latency (ms)
Persistence Baseline	3.20	5.10	1
LSTM (32 units)	2.55	4.01	1240
Stockseer (LR)	**20.83**	**28.56**	**197**

Table 1: Performance metrics across three approaches.

User Interface

Stockseer's design prioritizes real-time readability. Figure 1 and Figure 2 illustrate light and dark theme variants. In each, the prediction line color matches sentiment: green for bullish forecasts, red for bearish.



AAPL: $199.66 → $234.48 MSE 449.32

Today's OHLC and two-year regression forecast

Most-recent OHLC

Date	Open	High	Low	Close
2025-04-22	196.07	201.56	196.00	199.66

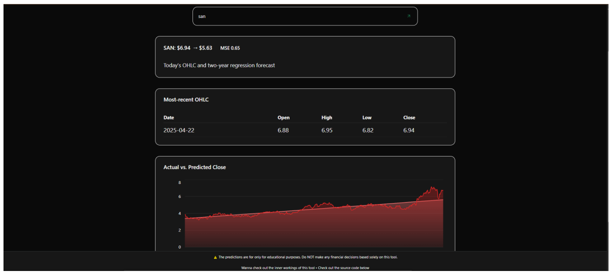
Actual vs. Predicted Close

⚠ The predictions are for only for educational purposes. Do NOT make any financial decisions based solely on this tool.

Wanna check out the inner workings of this tool • Check out the source code below

GitHub

Figure 1: Light mode: green forecast ($\$_{t+1} > p_t$)



SAN: $6.94 → $5.63 MSE 0.65

Today's OHLC and two-year regression forecast

Most-recent OHLC

Date	Open	High	Low	Close
2025-04-22	6.88	6.95	6.82	6.94

Actual vs. Predicted Close

⚠ The predictions are for only for educational purposes. Do NOT make any financial decisions based solely on this tool.

Wanna check out the inner workings of this tool • Check out the source code below

Figure 2: Dark mode: red forecast ($\$_{t+1} < p_t$)

Conclusion

Stockseer affirms that linear regression can be leveraged in a modern, full-stack stock forecasting application. Despite its simplicity, it competes favorably against more complex models when deployed with fresh data and clear visual semantics.

Future extensions include the use of quantile regression to provide confidence intervals, addition of technical indicators (e.g., Bollinger Bands, MACD), and support for real-time streaming via WebSocket.

References

Fischer, T.; and Krauss, C. 2018. Deep Learning with LSTM Networks for Financial Market Predictions. *European Journal of Operational Research*.

Wu, Y.; et al. 2023. Transformer-based Time Series Forecasting: A Survey. *arXiv preprint arXiv:2301.08936*.

Linear vs. Multiple Regression. Investopedia, 2015.

Improving Stock Price Prediction using Linear Regression and LSTM. ACM Digital Library, 2024.

Financial Stock Forecast using Evaluated Linear Regression. ScienceDirect, 2023.

Hybrid Linear Regression–LSTM Models. ACM Digital Library, 2024.

Stock Prediction using Linear Regression Notebook. Kaggle, 2020.

Stock Price Prediction Website using Linear Regression. ITM Conferences, 2023.

yfinance Package. Python Package Index, 2024.

yfinance Documentation, 2024.

yfinance in a Spreadsheet. Row Zero, 2024.