Predicting Stock Prices

Using Time-Series Models

**Yejin Hwang Hung-Yu Lin Andrews Amoah Abdullah Bdeir**

Texas A&M University-Corpus Christi

6300 Ocean Dr, Corpus Christi, TX 78412

{[yhwang, hlin5, aamoah, abdeir}@islander.tamucc.edu](mailto:yhwang,%20hlin5,%20aamoah,%20abdeir%7d@islander.tamucc.edu)

May 5, 2025

# Abstract

Accurate forecasting of stock prices remains a major challenge due to the volatile and nonlinear nature of financial markets. This project evaluates the predictive performance of three models—ARIMA (a traditional statistical method), TimesFM (a pretrained transformer by Google), and Chronos-T5 (a token-based transformer from Amazon)—on short-term stock price prediction. Using 5-day forecasts for three major stocks: Apple (AAPL), Tesla (TSLA), and Nvidia (NVDA), we compare model outputs against actual market behavior. Our results show that TimesFM excels in capturing short-term dynamics, while Chronos-T5 demonstrates better generalization over multiple days. The study highlights the strengths and trade-offs of traditional and AI-based forecasting approaches in real-world financial applications.

# 1. Introduction

Forecasting stock prices is a critical task in financial analytics, with far-reaching implications for investors, hedge funds, and automated trading systems. However, the unpredictable nature of financial markets driven by both internal trends and external events such as earnings announcements or political decisions makes accurate forecasting particularly challenging. Traditional statistical models often fall short in capturing the complex temporal dependencies and nonlinearities inherent in financial time-series data[1].

The recent advancement of deep learning, especially transformer-based models, offers new tools for capturing intricate temporal patterns across various domains, including finance. This project aims to investigate whether these state-of-the-art models can outperform traditional approaches in the context of short-term stock price prediction.

We address the following research question : **Can pretrained transformer-based models provide more accurate and generalizable forecasts of stock prices compared to a classical statistical model like ARIMA?** To explore this, we focus on predicting 5-day closing prices for three high-profile and volatile stocks: Apple (AAPL), Tesla (TSLA), and Nvidia (NVDA).

We use historical daily stock data (Open, High, Low, Close, Volume) from 2023 to late 2024 to predict the closing prices over a 5-day horizon in December 2024. And compare the following models: **ARIMA** – a widely used statistical approach for time-series forecasting. **TimesFM** – a Google-developed decoder-only transformer pretrained on a variety of time-series data. **Chronos-T5** – a language-model-inspired forecasting model that tokenizes time-series data for scalable sequence prediction.

# 2. Related Work

Stock price prediction has been widely studied using both traditional statistical models and modern deep learning approaches. Among statistical models, ARIMA (AutoRegressive Integrated Moving Average) remains foundational due to its interpretability and robustness in stable environments. It captures linear trends and seasonality via autoregressive terms, differencing, and moving averages. Although ARIMA has proven effective in short-term series[2], its performance tends to degrade in highly volatile or nonlinear financial markets.

In contrast, recent Transformer-based models have demonstrated strong capabilities in sequence modeling, particularly for financial time-series. TimesFM [3] is a decoder-only Transformer trained on diverse real-world and synthetic data, enabling zero-shot forecasting across domains. It uses patch-wise encoding and attention mechanisms to extract local and positional patterns in temporal sequences. Chronos-T5[4] reframes forecasting as a language modeling task by quantizing real-valued inputs into tokens. This token-based decoding enables long-range temporal reasoning and probabilistic predictions.

Other deep learning architectures such as Autoformer[5], Stockformer[6] and the Temporal Fusion Transformer[7] have also been effective in handling financial signals and multi-horizon forecasting, especially when interpretability and uncertainty quantification are required.

Our work contributes to this literature by conducting a unified evaluation of three paradigms—ARIMA, TimesFM, and Chronos-T5—on the same 5-day forecasting task across multiple high-volatility stocks. This design allows us to systematically compare performance across interpretability, adaptability, and forecasting accuracy.

# 3. Dataset and Features

In this study, we utilized historical daily stock data for three major technology companies: Apple (AAPL), Tesla (TSLA), and Nvidia (NVDA). The data was obtained from Yahoo Finance using the yfinance Python API. The training period spans from January 1, 2023, to November 1, 2024, while the testing period covers the short-term forecasting window from November 1 to November 9, 2024. Each record includes standard financial indicators such as opening price, highest and lowest price of the day, closing price, and trading volume. Among these, the closing price was selected as the sole predictive target due to its stability and wide use in financial analysis and modeling.

To ensure consistency and comparability across models, all three forecasting methods were applied to the same data under a unified preprocessing pipeline. For the ARIMA model, differencing was applied to the closing price series to achieve stationarity. The optimal parameters for the ARIMA configuration (p, d, q) were estimated using autocorrelation and partial autocorrelation plots. For TimesFM, the time series was normalized and then segmented into overlapping input-output sequences to match the patch-based input format expected by the pretrained transformer model. No fine-tuning was performed, as the TimesFM model was used in a zero-shot manner via the Hugging Face interface. Chronos-T5, in contrast, required the time-series to be scaled and discretized through quantization. The sequence was then tokenized and processed using a T5-style decoder, and the output tokens were subsequently dequantized back into continuous numeric values.

This consistent design allowed for a fair comparison between statistical and transformer-based models, ensuring that the same raw inputs were interpreted and transformed only as required by the model-specific architecture.

## **4. Methods**

This project compares three distinct time-series forecasting approaches: a statistical model (ARIMA), a pretrained decoder-only transformer (TimesFM), and a token-based transformer model (Chronos-T5). Each model was applied individually to predict the next 5 days of closing prices for AAPL, TSLA, and NVDA.

### **4.1 ARIMA (AutoRegressive Integrated Moving Average)**

ARIMA is a classical statistical model widely utilized for time series forecasting. It predicts future values based on past observations and residual errors. It integrates three primary components: Autoregression (AR), Differencing (I), and Moving Average (MA). Autoregression leverages the dependencies between current and past observations by regressing the current value on its previous values. Differencing is used to transform a non-stationary time series into a stationary series, which is critical for ARIMA modeling. The Moving Average component corrects predictions based on past forecast errors. Mathematically, an ARIMA(p, d, q) model is represented as:

* **AR (p):** Autoregression—uses lagged observations to model future values.
* **I (d):** Differencing—removes trend and seasonality to make the series stationary.
* **MA (q):** Moving average—models the error as a linear combination of past errors.

**Implementation Details:**

* The statsmodels library was used for model fitting.
* ACF and PACF plots helped estimate the optimal (p,d,q) values.
* The `auto\_arima()` function from the `pmdarima` package was additionally used to automate parameter selection, based on AIC minimization.
* The model was trained separately for each stock.
* Forecasts were generated recursively for the 5-day test window.

### **4.2 TimesFM-2.0-200M (Google Research)**

TimesFM is a state-of-the-art deep learning model developed by Google Research, specifically tailored for time-series forecasting. This model is a transformer-based, decoder-only foundation model pretrained on a wide variety of time-series data (e.g., economic indicators, Google Trends, weather data). It generalizes across domains and supports zero-shot inference[3].

**Key Components:**

* **Input Patch Encoding:** Time-series segments are split into input-output windows (e.g., 24-input steps to predict 5 output steps).
* **Positional Encoding:** Temporal order is preserved using learned or sinusoidal positional embeddings.
* **Self-Attention:** Enables the model to focus on relevant parts of the input window for each prediction step.

**Implementation Details:**

* Used Hugging Face’s timesfm pipeline with TimesFM-1.0-200M pretrained checkpoint.
* No fine-tuning was applied; the model was used in inference-only mode.
* Input patches were constructed using rolling windows over the normalized closing price series.

A diagram of a network

AI-generated content may be incorrect.

### **4.3 Chronos-T5 (Amazon)**

Chronos-T5 treats time-series forecasting as a language modeling problem. It discretizes numeric input sequences into tokens and leverages the T5 (Text-to-Text Transfer Transformer) framework for autoregressive generation[8].

**Key Components:**

* **Normalization and Quantization:** Converts raw values into discrete tokens using scaling and binning.
* **Tokenized Forecasting:** Feeds tokenized input sequences into a transformer decoder to predict future tokens.
* **Dequantization:** Converts predicted tokens back to numerical values for evaluation.

**Implementation Details:**

* Employed Hugging Face’s amazon/chronos-t5-small checkpoint.
* The AutoModelForCausalLM and AutoTokenizer interfaces were used to encode and decode the time-series.
* Forecasts were sampled from the model’s probability distribution and converted back to real-valued outputs.

Chronos-T5-small, introduced by Amazon[4], utilizes a novel approach to time series forecasting by transforming numerical series into discrete tokens. Initially, historical time series data undergoes mean scaling and quantization, resulting in context tokens that represent various segments of the series. These context tokens are then input into a Time Series Language Model, which employs cross-entropy loss during training to learn the probabilistic relationships between tokens. During inference, the model generates probabilistic forecasts by sampling tokens, subsequently dequantizing and unscaling them to reconstruct the forecasted numerical values. This approach effectively models uncertainty and complex temporal dynamics, facilitating robust and accurate probabilistic forecasts.

A diagram of a training and information

AI-generated content may be incorrect.

## **5. Experiments / Results / Discussion**

To evaluate the performance of the three forecasting models—ARIMA, TimesFM, and Chronos-T5—we conducted short-term (5-day) stock price predictions for Apple (AAPL), Tesla (TSLA), and Nvidia (NVDA). All models were trained on historical data from January 2023 to November 2024, and evaluated on their ability to forecast the closing prices from November 1 to November 5, 2024. Model performance was assessed using Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE).

**Figure 1** visualizes the forecasting results for each model and stock. ARIMA produced stable forecasts that aligned with general price trends but failed to respond to abrupt changes, especially during the U.S. election week for Tesla. TimesFM demonstrated high responsiveness in the first 2–3 days, effectively capturing local dynamics, but its accuracy declined in later time steps due to limited contextual range from patch-based encoding. Chronos-T5 generated smoother, probabilistic outputs that captured overall trajectory more reliably over the full 5-day horizon, particularly excelling in volatile periods like Tesla’s election spike—though it sometimes underestimated sharp peaks.

A screenshot of a graph

AI-generated content may be incorrect.

**Figure 1.** Forecasting performance of ARIMA, TimesFM, and Chronos-T5 models across three different stocks. Each row shows the full historical stock trend on the left and 5-day zoom-in of the predicted vs actual prices on the right.

**Table 1** presents the quantitative performance comparison. Chronos-T5 achieved the lowest RMSE on TSLA, indicating strong generalization in volatile settings, while TimesFM yielded the lowest MAE for AAPL, showing strength in short-term precision. ARIMA remained a reasonable baseline, particularly where interpretability and computational simplicity are preferred.

These findings suggest that TimesFM is best suited for near-term forecasting with fast trend adaptation, whereas Chronos-T5 offers better performance over multi-day horizons in volatile conditions. Although all models struggled to fully capture exogenous shocks, their relative strengths provide valuable insights for hybrid or ensemble-based approaches.

A table of numbers and letters

AI-generated content may be incorrect.

**Table 1.** Forecasting performance of ARIMA, TimesFM, and Chronos-T5 across different stocks.

### **6. Conclusion / Future Work**

**6.1 Conclusion**

This project explored the effectiveness of three time-series forecasting models—ARIMA, TimesFM, and Chronos-T5—for predicting 5-day stock price movements of Apple (AAPL), Tesla (TSLA), and Nvidia (NVDA). Each model represents a distinct methodological paradigm: ARIMA as a classical statistical model, TimesFM as a pretrained patch-based transformer, and Chronos-T5 as a token-based sequence model adapted from natural language processing.

Our findings highlight the strengths and limitations of each approach. ARIMA performed adequately in stable or trend-driven contexts but struggled with volatility and rapid market shifts. TimesFM showed superior short-term responsiveness, capturing immediate fluctuations in price with impressive fidelity, particularly during the first three days of forecasting. Chronos-T5 offered the best long-range forecasting performance and handled abrupt changes in market conditions with greater robustness, though it occasionally smoothed out high-frequency signals.

Importantly, all models exhibited decreased performance in the presence of unpredictable external events, such as Tesla's sharp price movements during the U.S. election week. This underscores the challenges of financial forecasting and the importance of integrating exogenous variables in future work.

Looking forward, several avenues can enhance forecasting accuracy and robustness. First, ensemble techniques that combine Times FM’s short-term precision with Chronos-T5’s long-range consistency may provide improved hybrid solutions. Second, integrating external sentiment data—such as news headlines, social media activity, and analyst reports—could enable models to anticipate regime changes. Third, incorporating technical indicators and macroeconomic signals may improve input richness and model generalization. Finally, while this study used transformer models in a zero-shot setting, domain-specific fine-tuning could unlock further performance gains, especially in volatile sectors like technology stocks.

By unifying classical and modern approaches in a comparative framework, this project provides a foundation for developing next-generation financial forecasting systems that are both accurate and adaptable.

This comparative study serves as a foundation for further research in building robust, hybrid time-series forecasting models for real-world financial applications.

# 6.2 Future work

**1.** **Model Assembling**  
We propose combining TimesFM and Chronos-T5 in a weighted ensemble framework. TimesFM can capture early time step patterns, while Chronos-T5 may extrapolate long-term trends effectively.

**2.** **Sentiment Integration**  
Future work will consider integrating sentiment scores derived from financial news articles or social media to enhance context-aware forecasting capabilities.

**3.** **Multivariate Features**  
Incorporating additional features such as technical indicators (e.g., RSI, MACD) or macroeconomic variables (e.g., interest rates) is expected to improve model performance.

**4.** **Extended Evaluation**  
We aim to expand the evaluation window to cover multiple weeks, and assess predictive accuracy using RMSE, MAE, and SMAPE to ensure robust statistical validation.

**5.** **Fine-Tuning Transformers**  
Future research will explore fine-tuning Chronos-T5 and TimesFM on domain-specific datasets within the financial sector, leveraging transfer learning strategies to improve performance.

# 7. Contributions

**Yejin Hwang**: Led the project’s experimental pipeline and analysis. Implemented and evaluated all three forecasting models (ARIMA, TimesFM, Chronos-T5), performed end-to-end data preprocessing, and generated all visualizations and evaluation tables. Also led the interpretation of results, created the final presentation, and contributed significantly to the report content.

**Hung-Yu Lin**: Focused on model development and experimentation. Fine-tuned TimesFM and Chronos-T5 via Hugging Face, adjusted hyperparameters, and prepared formatted outputs. Also managed the structure of the final report and edited all figures.

# ****Andrew Amoah****: Conducted the literature review and compiled the related work and reference section. Assisted with presentation delivery and supported the interpretation and explanation of results during team discussions.

# ****Abdullah Bdeir****: Participated in early team meetings and reviewed draft versions of the report.

**Reference:**

[1]. Sezer, Omer Berat, Mehmet Ugur Gudelek, and Ahmet Murat Ozbayoglu, Financial Time Series Forecasting with Deep Learning : A Systematic Literature Review: 2005-2019. arXiv [cs.LG], 2019.

[2]. Dhaduk, H., Stock market forecasting using time series analysis with ARIMA model. Analytics Vidhya, 2021.

[3]. Das, Abhimanyu, Weihao Kong, Rajat Sen, and Yichen Zhou, A decoder-only foundation model for time-series forecasting. arXiv [cs.CL], 2024.

[4]. Ansari, Abdul Fatir, Lorenzo Stella, Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, Jasper Zschiegner, Danielle C. Maddix, Hao Wang, Michael W. Mahoney, Kari Torkkola, Andrew Gordon Wilson, Michael Bohlke-Schneider, and Yuyang Wang, Chronos: Learning the Language of Time Series. arXiv [cs.LG], 2024.

[5]. Wu, Haixu, Jiehui Xu, Jianmin Wang, and Mingsheng Long, Autoformer: Decomposition Transformers with Auto-Correlation for Long-Term Series Forecasting. arXiv [cs.LG], 2022.

[6]. Bao, Wuzhida, Yuting Cao, Yin Yang, Hangjun Che, Junjian Huang, and Shiping Wen, Data-driven stock forecasting models based on neural networks: A review. Information Fusion, 2025. 113: p. 102616.

[7]. Lim, Bryan, Sercan Ö Arık, Nicolas Loeff, and Tomas Pfister, Temporal Fusion Transformers for interpretable multi-horizon time series forecasting. International Journal of Forecasting, 2021. 37(4): p. 1748-1764.

[8]. Raffel, Colin, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu, Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. Journal of Machine Learning Research, 2020. 21(140): p. 1-67.