JOURNAL OF
EMERGING INVESTIGATORS

Article

Stock price prediction: Long short-term memory vs. Autoformer and time series foundation model

Grant Lau[1], Steven Basart[2]
[1] Aragon High School, San Mateo, California
[2] University of Chicago, Chicago, Illinois

SUMMARY
Predicting future closing prices is important for investors because it can help them decide whether they want to invest in a stock and can provide insights into the total profits they might earn. We hypothesized that a long short-term memory (LSTM) model's architecture, coupled with its exclusive training on historical stock data, would result in better predictions of a stock's future closing prices compared to newer models that applied transfer learning. To test this, we analyzed the accuracy of a LSTM against two influential transfer learning models for stock price prediction: the Autoformer model and the time series foundation model (TimesFM). Each model received historical stock data for both Apple and Amazon and used it to predict future closing prices over a three-month period. After calculating the mean absolute error for each model, we found that the LSTM performed best when predicting Apple stock prices, and TimesFM performed better when forecasting Amazon stock prices. The Autoformer was the least accurate, possibly because it was trained on traffic data before stock data. By comparing the performance of these models, we have provided valuable insights for future research on stock price prediction models and show that domain-specific training can be superior to transfer learning for predicting stock prices in certain instances.

INTRODUCTION
Forecasting stock prices is an essential part of the finance industry. However, making accurate predictions of stock price movement is very difficult due to numerous factors, including release of earnings reports, the state of the stock market, investor sentiment, and changes in executive positions (1-4). Earnings reports and changes in executive positions may be expected to have a straightforward effect on stock prices, but this is not always the case. For example, beating expected earnings reports, which is perceived as positive, can sometimes negatively change a stock's price (5). An example of this can be seen in October 2024, when PayPal's stock dropped over 5% despite exceeding expectations for their third quarter earnings (6). Furthermore, in February 2024, changing chief executive officers caused Snowflake's stock to drop 20%, demonstrating the unpredictability of stock movements (7).

When people invest in stocks, they tend to use common resources to make informed decisions. Traditionally, investors use stock indicators, such as moving average convergence/divergence (MACD) and exponential moving average (EMA), to determine whether they should buy or sell a stock (8, 9). The MACD indicator has two lines: the MACD and signal line. When the MACD line crosses above the signal line, this is seen as a sign to buy. When the opposite occurs, it is seen as a sign to sell. The EMA indicator is a type of moving average indicator that helps investors visualize short or long-term trends. When a stock's price is above the EMA, it could signal a potential uptrend, and conversely, when the price is below the EMA, it could indicate a downtrend (8, 9). In addition to stock indicators, Wall Street analysts release predictions to the public on expected market changes. However, when predicting stock target prices over the span of a year, only 24% of predictions by human analysts were found to be accurate (10). To improve stock forecasting, new machine learning models are under development.

Time series forecasting, a method of using historical data to predict future trends and patterns, is a popular area of research being explored using machine learning (ML) models. Many ML and deep learning models have been developed for time series predictions, forecasting future values based on patterns in previously observed data points. Long short-term memory (LSTM) models are an example of a time series forecasting method that have already been leveraged for predicting stock prices (11-13). LSTMs work by evaluating each data point in a sequence individually and use context gained to update their memory, retaining important information and discarding irrelevant details (14). Previous studies have combined LSTMs with convolutional neural networks (CNNs) to predict stock prices with varying amounts of success (11, 13). Alternatively, transformers are models used to predict stock prices that rely on relevance filters to consider long-range dependencies and derive relationships between inputs and outputs (15-17).

Recently, studies have been published that compared the performance of LSTMs against transformers for stock price prediction (17,18). One study applied a LSTM, recurrent neural network (RNN), CNN, and transformer model to predict the prices of worldwide stock market indices such as CSI 300, S&P 500, Hang Seng Index, and Nikkei 225 (17). They used the daily closing prices of the indices from January 1, 2010, to December 31, 2020. In all cases, the transformer performed the best (17). Furthermore, in a second study, an LSTM, RNN, and transformer model were trained and tested on predicting the prices of Capgemini, Infosys, IBM Consulting, NTT Data, and Cognizant (18). This second study utilized historical closing stock prices from January 2, 2018, to June 3, 2024, and concluded that the transformer also performed the best,

https://doi.org/10.59720/24-228

noting that it required higher computational resources (18). This may stem from the transformer's longer training time, self-attention mechanism, positional encoding feature, and a better ability to model long-range dependencies (15, 18). Compared to LSTMs, which individually evaluate data points in a sequence and may lose information over longer sequences, the transformer's self-attention mechanism and positional encoding feature allow it to process all data points in a sequence simultaneously and learn dependencies between them, regardless of their distance (14, 15). This helps the model avoid the limitations of sequential processing faced by LSTMs, but comes at the cost of increased computational resources (18).

Despite the success of transformers for stock price prediction, models such as the Autoformer and Google Research's time series foundation model (TimesFM) have seen little to no testing for stock price prediction (19-21). Introduced in 2021, the Autoformer model was based on a transformer but is unique in its auto-correlation mechanism, which Rasul *et al*. demonstrated to be effective for time series forecasting (19, 22). On the other hand, TimesFM, introduced in April 2024, is special because it does not require dataset-specific training (20). To the best of our knowledge, there are no existing studies that have leveraged TimesFM, and only a study by Koa *et al*. has used the Autoformer to predict stock prices (23).

In the study by Koa *et al*., a hierarchical variational autoencoder (VAE) is combined with diffusion probabilistic techniques for stock price prediction, and the results are compared against those of an autoregressive integrated moving average (ARIMA), numerical-based attention (NBA), a vanilla VAE, a VAE with added adversarial perturbations to its input sequence, and the Autoformer model (23). The models are trained and tested on public US stock data across three different periods from January 2014 to January 2023. The models were evaluated using the average mean squared error and standard deviation across five different runs. In their findings, the Autoformer demonstrated competitive performance, outperforming models such as ARIMA, NBA, and the vanilla VAE, though it was ultimately surpassed by the hierarchical VAE combined with diffusion techniques (23). It achieved an average MSE ranging from 0.8625 ± 0.0378 to 1.2066 ± 0.0501 across different datasets and time periods, reflecting robust predictive capability under the tested conditions (23).

In this paper, we investigated the accuracy of two recent transformer models that apply transfer learning and compared them to a generic LSTM model for stock price prediction using two well-known stocks: Apple and Amazon (24). We hypothesized that the LSTM's architecture, coupled with its exclusive training on historical stock data, would result in better predictions of a stock's future closing prices compared to newer models, such as the Autoformer and TimesFM, that applied transfer learning. We found that the LSTM model only provided the most accurate stock predictions for Apple, demonstrating that the effectiveness of domain-specific training and the LSTM's memory architecture in predicting a stock's future closing prices is case dependent.

RESULTS

We sought to predict future stock prices using recent transfer learning Transformer models and a LSTM. Each model was trained on historical stock data from the corresponding company, ranging from the initial public offering (IPO) date to January 2024, before predicting the daily closing price. For the Autoformer, we had the option to provide time features in addition to the time series values it was trained to predict (daily closing price) (25). These time features provided extra context for our model by including the following data: the daily opening price, daily high, daily low, adjusted close, and stock trading volume. TimesFM and the LSTM were solely given the daily closing. Each model predicted a stock's future prices until June 2024. Then, we calculated the mean absolute errors (MAEs) produced using Scikit-learn's built-in library and compared them for each respective stock.

The Autoformer was the least accurate prediction model, with MAEs of 1.0083631695×10^{7} and 9.978526579×10^{6} for Apple and Amazon, respectively (**Figure 1**). These MAEs were over 392,000 times greater than the MAEs of the other two models. The LSTM achieved the most accurate results for predicting Apple's stock prices. We tested this model on varying numbers of epochs, where each epoch represents a complete pass through the entire dataset. This allowed the model to learn patterns in the data and better understand it. When ran with 100 epochs, it scored a MAE of 7.346 for Apple and followed an accurate trajectory given how it aligned with the same

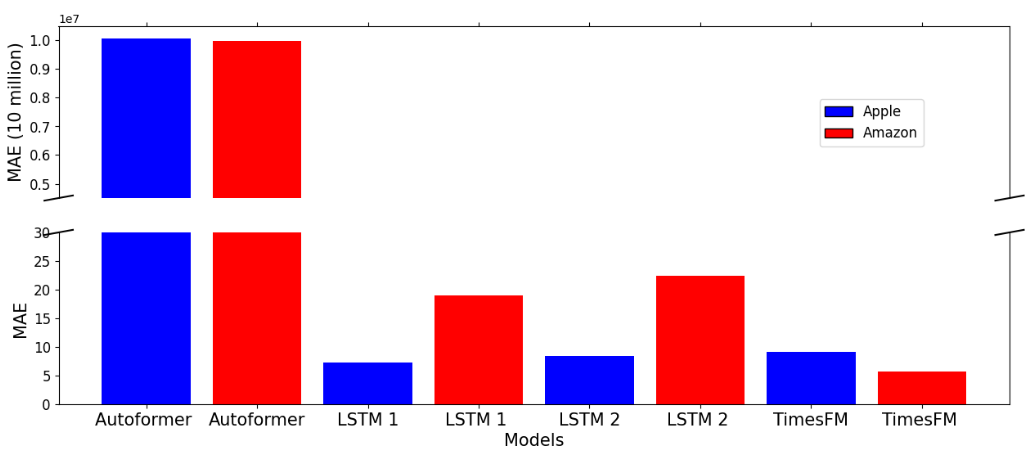

Figure 1: Mean absolute errors scored by the Autoformer, LSTM, and TimesFM. The red bars are for Amazon price predictions and the blue bars are for Apple price predictions. LSTM 1 is the LSTM trained on 100 epochs and LSTM 2 is the model trained on 200 epochs.

https://doi.org/10.59720/24-228

general direction (**Figure 2**). For Amazon, it scored a MAE of 19.046, which followed the correct trajectory but is not as close to the actual values (**Figure 2**). After we increased the number of epochs to 200, the MAE for Apple and Amazon rose to 8.420 and 22.462, respectively. As with 100 epochs, the forecasted trajectory for Apple more accurately followed the stock's true trajectory than with Amazon (**Figure 2**).

Google's TimesFM, which does not have an epoch parameter, initially scored a MAE of 11.169 for Apple and 25.704 for Amazon. When re-run after hyperparameter-tuning, the model scored a MAE of 9.105 for Apple and 5.696 for Amazon, making it the most accurate for predicting Amazon's stock prices. However, the TimesFM trajectories did not accurately reflect the dip or rise in price of the actual value of Apple's stock (**Figure 2**).

Apple's stock, and TimesFM demonstrated the best results for Amazon.

We believe the MAEs achieved by the LSTM and TimesFM are relatively good when compared to other studies. Prior studies have demonstrated strong performance using LSTM-based models (11, 12, 26). The model used by Kim *et al.* was trained and tested on adjusted closing price and trading volume data from the SPDR S&P 500 ETF data, ranging from October 14, 2016, to October 16, 2017 (11). Using this data, the model predicted the price five minutes into the future based on the previous 30 minutes of data (11). The model used by Selvin *et al.* was given timestamp, transaction ID, price, and trading volume information from 1721 stocks on the National Stock Exchange of India (12). The dataset used ranged from July 2014 to June 2015, and the model was trained to predict

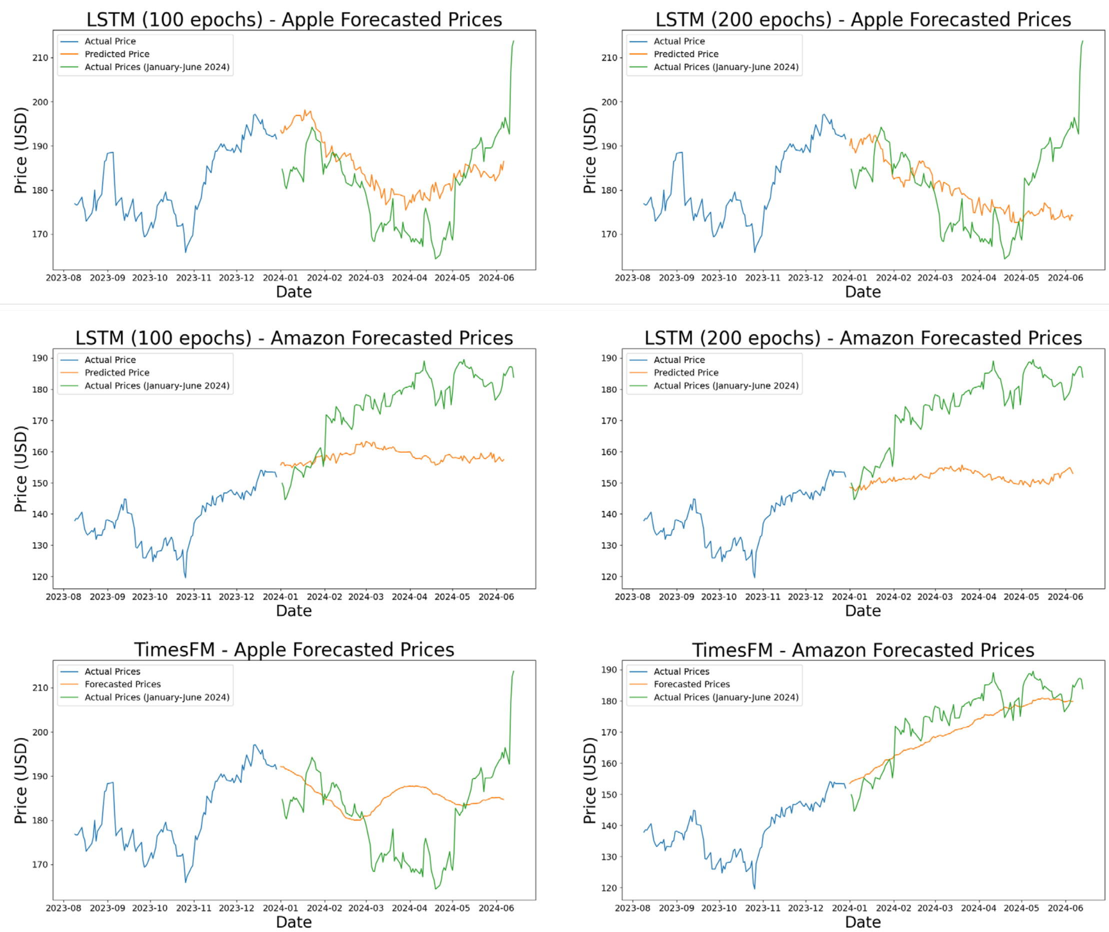

Figure 2: LSTM and TimesFM model predictions for Apple and Amazon stock prices. The blue line represents the actual historical stock prices used to train the model (up to the prediction range). The green line shows the actual stock prices during the prediction period, and the orange line displays the model's predicted prices for the same period. The LSTM model was trained using both 100 and 200 epochs with a univariate time series of Apple and Amazon stock prices. TimesFM does not require dataset-specific training, instead leveraging a pre-trained architecture to make predictions directly from the provided historical stock data. Results represent a single run of the model; replicates were not performed in this experiment.

DISCUSSION

We hypothesized that a LSTM's architecture and domain-specific training would outperform transfer learning models, Autoformer and TimesFM, when predicting a stock's future closing prices. We found that the Autoformer had the least accurate performance, the LSTM performed the best for prices 10 minutes ahead based on the previous 90 minutes of data (12). Finally, Gülmez's LSTM variants were trained and tested on a dataset comprising the prices of stocks from the Dow Jones Industrial Average index, spanning from January 1, 2018, to January 1, 2023 (26). The models were programmed to predict the price 1 day into the future based

https://doi.org/10.59720/24-228

on the previous 20 days (26). To compare our models to other studies that evaluate their models using percent error, we calculated the mean absolute percent error (MAPE) for our LSTM and TimesFM models. We then multiply the MAPEs by 100 to calculate the MAPE percentage. While MAPE percentage and percent error are not identical metrics, they are closely related, making MAPE percentage a suitable and comparable alternative for this evaluation. The performances of the models from each study are summarized in **Table 1**.

There are many potential reasons for the Autoformer's inaccurate results. Firstly, the version used in the study was pre-trained on traffic data from an open-source company called Hugging Face to predict future traffic jams (22). Despite the Autoformer's transfer learning abilities, we believe the volatile behavior of stock prices in the technology sector is too different compared to traffic data, which tends to be more predictable (27). Because of this, the Autoformer likely did not understand how these new values would impact the values it was trying to predict. Moreover, when feeding data to the Autoformer, the data had to be fed in tensors of a specific shape, which prevented 550 rows or 5.25% of Apple stock data and 749 rows or 11.18% of Amazon stock data from being accessed by this model. This resulted in a gap and large price jump in the data that may have also contributed to the model's poor performance.

Unlike both transformer models, the LSTM was the only one trained solely on the selected stock's data, which may be the reason why it performed the best out of the three when predicting future stock prices for Apple. However, since the LSTM only performed the best when predicting future Apple stock prices, the LSTM's performance may be company-dependent. While this model was not as accurate as those in previous studies, we still obtained promising results considering the study's limitations and its three-month timeframe. The higher accuracy observed in prior studies may be attributed to the inclusion of additional information being provided to their models (e.g., trading volume) and the use of substantially shorter prediction horizons—such as minutes or a single day—compared to the six month window used in this study (11, 12, 26). Improving the accuracy of models with longer prediction periods is an area where future research may be able to improve upon. The TimesFM model does not require dataset-specific training, which allowed for immediate input of stock data. Its ability to leverage patterns learned from other types of time series data likely contributed to its accurate predictions.

To our knowledge, this study is the first to test the TimesFM model and evaluate its ability to predict stock values. Additionally, all experiments were conducted within a Google Colab environment, supporting ease of replication and accessibility for others in both the research and non-research communities. This study demonstrates the accessibility of capable time series models that the public can leverage, though the performance of these models should be improved before seriously considering using them for personal trading. Nevertheless, we believe that the results of our approaches using the LSTM and TimesFM models demonstrate that combining domain-specific training with the LSTM's memory architecture does not yield better predictions of a stock's future closing prices compared to transfer learning models. As such, our findings do not support our original hypothesis.

In the future, some steps can be taken to improve the accuracy of these models. For the Autoformer model, restructuring the Autoformer's data loader to not require strict tensor shapes will optimize the process of feeding the model with training and testing data for datasets beyond stock values. Additionally, because each model was only provided historical stock data to make its predictions, factoring in common stock indicators used by human analysts, such as MACD and EMA, and including other factors that affect a stock's price may produce better results (8, 9). Furthermore, more factors such as the state of the stock market and investor sentiment should be considered for more accurate predictions. Improving the accuracy of models with longer prediction horizons is another area where future research could expand on. In addition, more in-depth experimentation with hyperparameters such as per core batch size and the number of layers of TimesFM may improve accuracy. Ultimately, the success of each model came down to how each model was trained. Future research could also work on optimizing the transfer learning strategies of the Autoformer model to enhance its predictive capabilities. We set out to evaluate the effectiveness of pairing the LSTM's memory-based architecture with exclusive training on historical stock data, hypothesizing that this approach would result in superior predictions compared to transfer learning models. However, our results did not support our hypothesis. We found that the LSTM performed the best when predicting future stock prices for Apple, TimesFM performed best for Amazon stock prices, and the Autoformer performed the

Study	Model	Data	MAPE (%)	Notes
This study	LSTM	Trained and tested on AAPL and AMZN (IPO-Jan 2024)	AAPL: 4.00 AMZN: 11.7	Trained on closing price; predicts 6 months ahead
This study	TimesFM	Trained and tested on AAPL and AMZN (IPO-Jan 2024)	AAPL: 5.13 AMZN: 3.24	Trained on closing price; predicts 6 months ahead
Kim, Taewook, and Ha Young Kim (2019)	ST-LSTM	SPDR S&P 500 ETF data (97,474 records from October 14, 2016 to October 16, 2017)	3.11	Trained on adjusted closing price and trading volume; predicts 5 minutes ahead based on previous 30 minutes of data
Selvin *et al.* (2017)	LSTM	1721 NSE stocks (June 2014-2015); evaluated Infosys, TCS, and Cipla	Infosys: 4.18 TCS: 7.82 Cipla: 3.94	Trained on timestamp, transaction ID, price, volume; predicts 10 minutes ahead based on previous 90 minutes
Gülmez (2023)	LSTM (ARO, GA, 1D, 2D, 3D variants)	DJIA index stocks (January 2018-2023)	**Apple:** Artificial Rabbits Optimization algorithm: 2.5 Genetic Algorithm: 2.6 1D: 3.7 2D: 3.2 3D: 4.2	Trained on prices; predicts 1 day ahead based on previous 20 days

Table 1: Comparison of Stock Price Prediction Models Using LSTM and TimesFM. Comparison of various studies that applied machine learning models for stock price prediction. The first column lists the study being referenced, while the second column identifies the predictive model used in each study. The third column describes the dataset, including the specific stocks analyzed and the time span of the data. The fourth column reports model performance in terms of mean absolute percentage error (MAPE). The fifth column includes additional notes, such as the input features used and the prediction horizon.

worst out of the three. These findings suggest that neither training a model specifically on stock data nor applying transfer learning guarantees better performance.

MATERIALS AND METHODS

For each experiment, we used Google Colab to take advantage of its graphics processing unit for handling computations. Additionally, each model predicted the daily closing price of the selected stock. The Autoformer used an Apple stock dataset from Hugging Face and extracted stock data for Amazon through Yahoo Finance. For the LSTM and TimesFM models, stock data of both Apple and Amazon were extracted using Yahoo Finance. The source code of this study is available on GitHub (28).

Autoformer

We first adapted the necessary code from Rasul *et al*. for a functioning Autoformer (22). Once the model was functional, we formatted the daily historical open, high, low, adjusted close, and trade volume of Apple's stock into the past and future time features tensors. Next, we filled the past values tensor with the daily closing prices. Additionally, the Autoformer required a "past observed mask" tensor. This indicated which elements of a sequence to process. We then created and filled this tensor with ones using PyTorch's torch.ones() function. Afterward, we put all the tensors in a dictionary and inputted them into the pre-existing data loader for the Autoformer to make its predictions. We then modified the evaluation code to calculate the MAE and ran it. Afterward, we repeated the process with Amazon. With the initial testing complete, we tried tuning the model by changing the pre-set parameters but achieved the same performance.

Long short-term memory (LSTM)

Unlike transformers, which immediately look at the entire time sequence, LSTMs analyze sequences one part at a time (14, 15). To set it up, we used code found on Kaggle and gathered the necessary stock data (18). Then, we manually divided the data into an 80/20 training and testing split. The training data was fed into a sequence splitter and fitted with the model. Once fitted (trained), the model made its predictions with steps set to three, i.e., three days. We also tested the LSTM on 100 and 200 epochs to see how the results differ (**Figure 2**). Finally, we compared the predictions against the testing data and scored the model's performance using the MAE. We then fine-tuned the model by experimenting with various step sizes ranging from 3-15, with three steps resulting in performance seen in the results section.

Google's time series foundation model (TimesFM)

TimesFM is a type of transformer that was trained on 100 billion time-series data points and does not require dataset-specific training (19, 20). This allowed us to skip the training process, immediately input historical stock data, and begin forecasting (21, 29). To use it, we first downloaded the repository and installed TimesFM into the notebook. Then, we retrieved the model's code from GitHub and modified the model to predict values from January 2024 to the present day (June 2024), ensuring we could evaluate its entire prediction using the MAE (21, 29, 30). Finally, we set codelist to "AAPL" and dropped all the data except the closing prices. To repeat the process with Amazon, we changed codelist from "AAPL" to "AMZN." The initial model's parameters were context_len of 2048, horizon_len set to January 2024 to June 2024, per_core_batch_size of 32, num_layers of 50, use_positional_embedding set to False, and backend set to 'cpu.' After hyperparameter-tuning, TimesFM achieved its best score for Apple with context_len changed to 672. For Amazon, its best performance was achieved with context_len set to 608.

Received: July 15, 2024
Accepted: Novemeber 19, 2024
Published: August 12, 2025

REFERENCES

1. Collins, D.W. and S.P. Kothari. "An Analysis of Intertemporal and Cross-Sectional Determinants of Earnings Response Coefficients." *Journal of Accounting and Economics*, vol. 11, no. 2-3, July 1989, pp. 143–181, https://doi.org/10.1016/0165-4101(89)90004-9.
2. Fama, E.F. and K.R. French. "Common Risk Factors in the Returns on Stocks and Bonds." *Journal of Financial Economics*, vol. 33, no. 1, Feb. 1993, pp. 3–56, https://doi.org/10.1016/0304-405X(93)90023-5.
3. Bond, Frederic Drew. *Stock Prices*. 1911.
4. Zhang, Yu. *How Does Untimely Death of an Executive Affect Stock Prices and Company Performance?* 2015. Utah State University, MS dissertation. *DIGITALCOMMONS@USU* https://digitalcommons.usu.edu/gradreports/454.
5. Mian, G.M. and S. Sankaraguruswamy. "Investor Sentiment and Stock Market Response to Earnings News." *The Accounting Review*, vol. 87, no. 4, Mar. 2012, pp. 1357–1384, https://doi.org/10.2308/accr-50158.
6. Bary, Emily. "PayPal beat big on a profit metric, but here's why its stock is falling." *MarketWatch*, 29 Oct. 2024, www.marketwatch.com/story/paypal-boosts-profit-outlook-but-its-stock-falls-after-earnings-a9ab3170. Accessed 24 May 2025.
7. Bary, Emily. "Snowflake's Stock Plunges after Earnings, with CEO Switch a "Massive Surprise."" *MarketWatch*, 28 Feb. 2024, www.marketwatch.com/story/snowflakes-stock-plunges-as-outlook-misses-and-ceo-steps-down-a1e24917?utm_source=chatgpt.com. Accessed 15 Jan. 2025.
8. Appel, Gerald. *The Moving Average Convergence-Divergence Trading Method*. Traders Pr, 1985.
9. Hunter, J.S. "The Exponentially Weighted Moving Average." *Journal of Quality Technology*, vol. 18, no. 4, Oct. 1986, pp. 203–210, https://doi.org/10.1080/00224065.1986.11979014.
10. Kerl, A.G. "Target Price Accuracy." *Business Research*, vol. 4, no. 1, Mar. 2011, pp. 74–96, https://doi.org/10.1007/bf03342727.
11. Kim, T. and H.Y. Kim. "Forecasting Stock Prices with a Feature Fusion LSTM-CNN Model Using Different Representations of the Same Data." *PLOS ONE*, vol. 14, no. 2, 15 Feb. 2019, https://doi.org/10.1371/journal.pone.0212320.
12. Selvin, S. *et al*. "Stock Price Prediction Using LSTM, RNN, and CNN Sliding Window Model." *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI): Udupi, India, 13-16 September 2017*, IEEE, 2017, pp. 1643-1647. https://doi.org/10.1109/

https://doi.org/10.59720/24-228

ICACCI.2017.8126078.
13. Mehtab, S. and J. Sen. "Stock Price Prediction Using CNN and LSTM-Based Deep Learning Models." *2020 International Conference on Decision Aid Sciences and Application (DASA), Sakheer, Bahrain, 08-09 November 2020,* IEEE, 2020 pp. 447-453. https://doi.org/10.1109/DASA51403.2020.9317207.
14. Hochreiter, S. and J. Schmidhuber. "Long Short-Term Memory." *Neural Computation*, vol. 9, no. 8, Nov. 1997, pp. 1735–1780, https://doi.org/10.1162/neco.1997.9.8.1735.
15. Vaswani, A. *et al*. "Attention is All You Need." *NIPS'17: 31st International Conference on Neural Information Processing Systems: Long Beach, California, 04-09 December 2017,* Edited by Ulrike von Luxburg and Isabelle Guyon, Curran Associates Inc., 2017, pp. 6000-6010. *Association for Computing Machinery Digital Library*.
16. Muhammad, T., *et al*. "Transformer-Based Deep Learning Model for Stock Price Prediction: A Case Study on Bangladesh Stock Market." *International Journal of Computational Intelligence and Applications*, 6 Apr. 2023, https://doi.org/10.1142/s146902682350013x.
17. Wang, C., *et al*. "Stock Market Index Prediction Using Deep Transformer Model." *Expert Systems with Applications*, vol. 208, Dec. 2022, p. 118128, https://doi.org/10.1016/j.eswa.2022.118128.
18. Rukmana, P.U. *et al.* "Application of LSTM, RNN, and Transformer in Stock Price Prediction of Information Technology Companies: A Comparative Analysis." *2024 International Conference on Computer, Control, Informatics and Its Applications (IC3INA), Bandung, Indonesia, 09-10 October 2024*, IEEE, 2024, pp. 249-254. https://doi.org/10.1109/IC3INA64086.2024.10732754.
19. Wu, H. *et al*. "Autoformer: Decomposition Transformers with Auto-Correlation for Long-Term Series Forecasting." *NIPS'21: 35th International Conference on Neural Information Processing Systems, 06-14 December 2021,* Edited by M. Ranzato, A. Beygelzimer, and Y. Dauphin, Curran Associates Inc., 2021, pp. 22419 - 22430. *Association for Computing Machinery Digital Library.*
20. Das, A. *et al*. "A Decoder-only Foundation Model for Time-Series Forecasting." *41st International Conference on Machine Learning, Vienna, Austria, 21-27 July 2024,* Proceedings of Machine Learning Research, 2024, pp. 10148-10167.
21. Zhou, Y. *et al*. "Google-Research/Timesfm." *GitHub*, https://github.com/google-research/timesfm/tree/master. Accessed 18 January. 2025
22. "Yes, Transformers Are Effective for Time Series Forecasting (+ Autoformer)." *Huggingface*. https://huggingface.co/blog/autoformer. Accessed 4 May. 2024
23. Koa, K. *et al*. "Diffusion Variational Autoencoder for Tackling Stochasticity in Multi-Step Regression Stock Price Prediction." *32nd ACM International Conference on Information and Knowledge Management: Birmingham, United Kingdom, 21-25 October 2023*, Edited by Ingo Frommholz and Frank Hopfgartner, Association for Computing Machinery, 2023, pp. 1087-1096. *Association for Computing Machinery Digital Library*.
24. "LSTM Models for Time Series Forecasting." *Kaggle*. www.kaggle.com/code/pawan2905/lstm-models-for-time-series-forecasting#Univariate-LSTM-Models. Accessed 16 June. 2024.
25. "Autoformer-Transformers documentation." *Hugging Face*. https://huggingface.co/docs/transformers/en/model_doc/autoformer#transformers.AutoformerForPrediction. Accessed 29 June. 2024.
26. Gülmez, B. "Stock Price Prediction with Optimized Deep LSTM Network with Artificial Rabbits Optimization Algorithm." *Expert Systems with Applications*, vol. 227, no. 120346, May 2023, p. 120346, https://doi.org/10.1016/j.eswa.2023.120346.
27. Kambale, W.V. *et al.* "Transformers in Time Series Forecasting: A Brief Analysis of the Autoformer Transfer Learning Performance." *2023 30th IEEE International Conference on Electronics, Circuits and Systems (ICECS),* Istanbul, Turkiye, 2023, pp. 1-6, https://doi.org/10.1109/icecs58634.2023.10382781.
28. "Transformers For Stock Price Prediction." *GitHub*. https://github.com/GrLau/Transformers-For-Stock-Price-Prediction. Accessed 8 June. 2024
29. "Timesfm/README.md at Main · Lhw828/Timesfm." *GitHub*. https://github.com/lhw828/timesfm/blob/main/README.md. Accessed 1 July. 2024
30. "Google/Timesfm-2.0-500m-Pytorch" *Hugging Face.* https://huggingface.co/google/timesfm-2.0-500m-pytorch. Accessed 18 January 2025.