Transactions on Engineering and Computing Sciences - Vol. 13, No. 03
Publication Date: May 25, 2025
DOI:10.14738/tmlai.1303.18843.
Sendi, A., Hoque, F., & Hoque, M. T. (2025). Decoding Stock Trends: A Comparative Study of GRU, LSTM, and Transformer Models in Tech Sector Prediction. *Transactions on Engineering and Computing Sciences*, 13(03). 46-60.

Decoding Stock Trends: A Comparative Study of GRU, LSTM, and Transformer Models in Tech Sector Prediction

Asaad Sendi
0000-0003-3656-365X
Islamic University of Madinah

Foyzul Hoque
ORCID: 0009-0009-5132-6477
Independent University Bangladesh

Md Tamjidul Hoque*
ORCID: 0000-0002-0110-2194
University of New Orleans, USA

ABSTRACT

This study investigates the predictive capabilities of three state-of-the-art neural network architectures—Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), and Transformer models—in forecasting stock prices for five leading technology companies: Apple Inc. (AAPL), Cisco Systems, Inc. (CSCO), Meta Platforms, Inc. (META), Microsoft Corporation (MSFT), and Tesla, Inc. (TSLA). The dataset spans from July 2019 to July 2023, with models evaluated using key performance metrics such as Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R-squared (R^2) values. The results show that GRU models consistently achieve the lowest MAE, indicating superior precision in stock price prediction. In contrast, LSTM models, while showing slightly higher error rates, are particularly effective in capturing long-term trends and the variance in stock prices, as evidenced by higher R^2 values. Transformer models, which utilize self-attention mechanisms, showed promise in handling complex relationships but struggled with the volatility of certain stocks, such as TSLA, leading to higher errors and lower R^2. These findings provide valuable insights for financial analysts and investment professionals, offering guidance on selecting the most suitable deep learning models based on specific market conditions and forecasting needs. This study also lays the groundwork for further exploration into model optimization and hybrid approaches for more robust financial forecasting.

Keywords: GRU, LSTM, Transformer, stock prediction, machine learning, financial forecasting, AI in finance.

INTRODUCTION

Stock price prediction remains one of the most challenging problems in financial analysis due to the inherently complex and dynamic nature of the market. The nonlinearity, volatility, and non-stationarity of financial time series make it difficult for traditional statistical models to

* Corresponding Author, thoque@uno.edu

achieve high accuracy (Abu-Mostafa & Atiya, 1996; Henrique et al., 2019; Lin et al., 2021). As a result, researchers have continually sought more robust methods to enhance prediction accuracy (Hu, 2021; Lu et al., 2020; Mian, 2023; Qiao et al., 2022).

While conventional methods like autoregressive models and generalized autoregressive conditional heteroskedasticity (GARCH) have been widely used, they often fall short of addressing the complexities of stock market data, especially in the presence of noise and unpredictable shifts. This limitation has spurred the adoption of machine learning techniques, particularly deep learning, which has proven to be a powerful tool for capturing intricate patterns in time-series data (Beniwal et al., 2024; Shahi et al., 2020).

Deep learning models, such as neural networks, are well-suited to stock price prediction tasks because they can process large amounts of noisy and nonlinear data without requiring extensive manual feature engineering. These models are capable of learning from raw data and automatically identifying patterns that traditional models might overlook (Ali et al., 2023; Y. Yu & Kim, 2019). As deep learning continues to evolve, its applications in finance, particularly in predicting stock prices, have gained significant attention. Recent advancements, including Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), and Transformer models, have contributed to improving the accuracy and efficiency of stock market forecasts (Tao et al., 2024; Wu et al., 2022; X. Yu et al., 2023; Zhou et al., 2023).

This study focuses on evaluating and comparing the performance of three advanced deep learning models—GRU, LSTM, and Transformer—in predicting stock prices for five prominent companies: AAPL, CSCO, META, MSFT, and TSLA. Through an analysis of key metrics such as Mean Absolute Error (MAE) and R-squared (R^2), the study demonstrates that GRU models consistently yield the lowest MAE, reflecting superior prediction precision. LSTM models, while exhibiting slightly higher MAE, outperform Transformers in capturing broader stock trends and variance, as evidenced by higher R^2 values. In contrast, Transformer models generally show lower accuracy and less effective fitting to historical price data, particularly for volatile stocks like TSLA.

The remainder of this paper is organized as follows: Section 2 reviews related literature, Section 3 outlines the data and methodology used in this study, and Sections 4 and 5 present the results and conclusions, respectively.

RELATED WORK

These models were designed to handle linear time-series data and have been useful in capturing short-term patterns and volatility clustering in financial markets. ARIMA models are based on the assumption that future values in a time series are linearly dependent on past observations, while GARCH models are designed to model and predict time-varying volatility, an important aspect of financial forecasting. Although these methods have demonstrated effectiveness in various contexts, they often struggle to capture the nonlinear and nonstationary nature of financial markets, where sudden shifts, complex interactions, and unanticipated events frequently influence stock prices(Beniwal et al., 2024; Shahi et al., 2020).

In recent years, there has been a shift toward using machine learning techniques, particularly neural network-based models, which are better equipped to handle the complexities and intricacies of financial data. Among these, Long Short-Term Memory (LSTM) networks have gained significant popularity due to their ability to model temporal dependencies in sequential data. LSTM networks, as demonstrated in studies by Lin *et al.*, (2021) and Ghosh *et al.*, (2022), excel at capturing long-range dependencies by mitigating the vanishing gradient problem commonly seen in traditional recurrent neural networks (RNNs). This makes them particularly effective in predicting stock prices, where past events can have long-term impacts on market behavior. LSTM's ability to capture trends, fluctuations, and seasonality in stock prices has contributed to its widespread adoption in finance.

Similarly, Gated Recurrent Unit (GRU) networks, introduced by Chung *et al.*, (2014), offer similar capabilities to LSTMs but with a more streamlined architecture. GRUs require fewer parameters, which reduces the computational cost and training time, making them a more practical solution for real-time applications. Their efficiency in learning sequential data while preserving long-term dependencies makes them a viable alternative to LSTM models, especially when quick model updates or fast decision-making are critical in financial markets.

Initially designed for natural language processing (NLP) tasks, transformer models have recently been applied to time-series forecasting, including stock market prediction. Unlike RNN-based architectures, transformers leverage a self-attention mechanism to capture dependencies between all elements in the input sequence, irrespective of their position. This approach allows transformers to model local and global dependencies in data, making them well-suited for capturing complex patterns in financial data. Although still relatively new in financial forecasting, studies by Tao *et al.*, (2024) and Li & Qian, (2022) have shown that transformer models can successfully predict stock prices by capturing intricate temporal relationships and market dynamics through attention mechanisms.

Despite these advancements, there remains a lack of comprehensive studies comparing the performance of these three models—LSTM, GRU, and Transformer—across diverse datasets, particularly in the context of the technology sector. While individual studies have demonstrated the effectiveness of each model in stock prediction, a direct comparison, especially in terms of their ability to handle the specific challenges of technology sector stocks, remains scarce. This study aims to fill this gap by benchmarking LSTM, GRU, and Transformer models against each other, offering a more holistic view of their relative strengths and weaknesses. Doing so provides valuable insights into their applicability and effectiveness in financial forecasting, particularly for stocks with varying levels of volatility, like those in the technology sector.

DATA AND METHODOLOGY

A detailed dataset description, data processing, and methodologies will be discussed next.

Dataset Description

The dataset utilized in this study consists of daily closing stock prices for five prominent technology companies: Apple Inc. (AAPL), Cisco Systems, Inc. (CSCO), Meta Platforms, Inc. (META), Microsoft Corporation (MSFT), and Tesla, Inc. (TSLA). These companies were chosen due to their significance in the technology sector and their varied market behaviors, which provide a comprehensive basis for evaluating the models' performance in different market conditions. The dataset spans four years, from July 2019 to July 2023, ensuring a sufficiently long timeline to capture both market trends and volatility across different economic phases, including periods of growth and contraction.

The raw stock price data was sourced from Yahoo Finance, a widely used and reliable platform for historical financial data. Yahoo Finance provides free access to a wealth of stock market data, which is crucial for conducting time-series analysis and training machine learning models. The data includes daily closing prices, which were selected as the primary feature for stock price prediction. This time-series data captures the daily fluctuations in stock value, which are influenced by various market forces, economic news, earnings reports, and investor sentiment.

Prior to feeding the data into the machine learning models, several preprocessing steps were applied to ensure its suitability for training the deep learning algorithms:

- *Normalization*: Given the varying scales of stock prices between companies (e.g., TSLA often being more volatile than MSFT), the data was normalized to a range of [0, 1]. This step helps improve the neural networks' convergence rate by ensuring that all features are on a similar scale. The Min-Max normalization technique was applied, which rescales each feature in the dataset to fall within the specified range.
- *Handling Missing Data*: Any missing values in the dataset were addressed using linear interpolation, ensuring a continuous time series without gaps. This is particularly important in stock market data, where missing values distort trends and predictions.
- *Smoothing*: To reduce the impact of noise and short-term fluctuations, the dataset was optionally smoothed using a moving average technique, which helps highlight underlying market trends.

For model evaluation and to avoid overfitting, the data was split into three distinct subsets:

- *Training Set* (80%): Most of the data was used to train the models, allowing the algorithms to learn the underlying patterns and relationships in the stock prices.
- *Validation Set* (10%): This subset was used during training to fine-tune hyperparameters and make decisions about model architecture, ensuring that the model generalizes well to unseen data without overfitting.
- *Test Set* (10%): After the models were trained, the test set was used to evaluate their final performance. This set provided an unbiased assessment of how the models would perform on future, unseen data, simulating real-world prediction scenarios.

The careful preprocessing and splitting of the data ensure that this study's results are robust, reliable, and reflective of real-world conditions, where stock price predictions need to be accurate and generalizable.

Model Architectures

The GRU, LSTM, and Transformer models were implemented using TensorFlow and Keras. Each architecture was fine-tuned to optimize performance on the given datasets. The GRU and LSTM models featured two recurrent layers with 64 units each, followed by dense layers. The

Transformer model included three encoder blocks, each with multi-head attention and feed-forward layers. Hyperparameters such as learning rate, batch size, and number of epochs were optimized through grid search. The insight of each of the models is given next.

GRU:

Introduced by Chung *et al.*, (2014) the gated recurrent unit (GRU) is a relatively newer advancement that tackles the problem of long-term dependencies. This challenge can result in weak gradients in larger conventional RNNs. The GRU streamlines the LSTM design by incorporating a gating mechanism to regulate the internal information flow, but it does this without using distinct memory cells. The GRU architecture will be designed with an input layer that accepts time-series data, one or more GRU layers, and a dense output layer to predict the stock price.

$$r_t = \sigma\left(W_{xz}\, Tx_{(t)} + W_{hz}\, Th_{t-1} + b_r\right) \tag{1}$$

$$z_t = \sigma(W_{xr}\, Tx_{(t)} + W_{hr}\, Th_{t-1} + b_z) \tag{2}$$

$$g_t = tanh(W_{xg}\, Tx_{(t)} + W_{hg}(r_t * h_{t-1}) + b_g) \tag{3}$$

$$h_t = z_t + h_{t-1} + (1 - z_t) * g_t \tag{4}$$

where σ is the sigmoid activation shown in Eq.5 , r_t is the rest gate, z_t is the update gate, $h_t - 1$ is the previous hidden state, x_t is the input at time t, W_r, W_z, W are the weights, b_r, b_z, b are the biases, and h_t is the current hidden state. The tanh activation function is shown in Eq. 6.

As depicted in Figure 1, the model's input layer contains several neurons, with the count determined by the feature space's dimensionality. In the GRU architecture, the forget and input gates are unified into one entity known as the "update gate," complemented by a "reset gate" that manages information flow within the unit, eliminating the need for separate memory cells. GRUs have shown an enhanced capacity to grasp long-term relationships within sequences, leading to their increasingly widespread adoption (Géron, 2019). As shown in Figure 1, the operations are specifically described using from Eq. 1 to Eq 4.

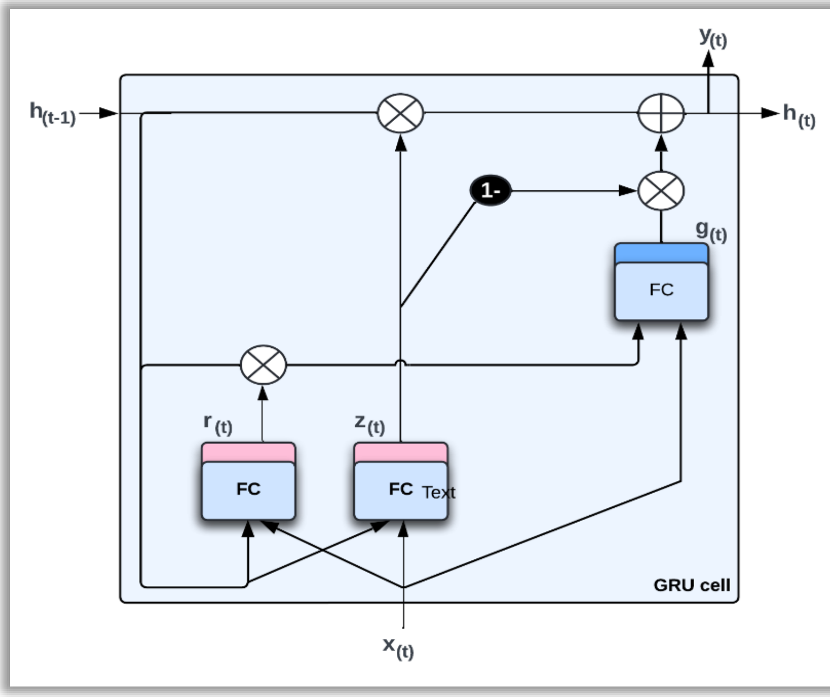

Figure 1: The Gated Recurrent Unit (GRU), introduced by Chung *et al*., (2014), addresses long-term dependency challenges in RNNs by incorporating a streamlined gating mechanism without distinct memory cells. Its architecture includes input and output layers for time-series data prediction.

LSTM:

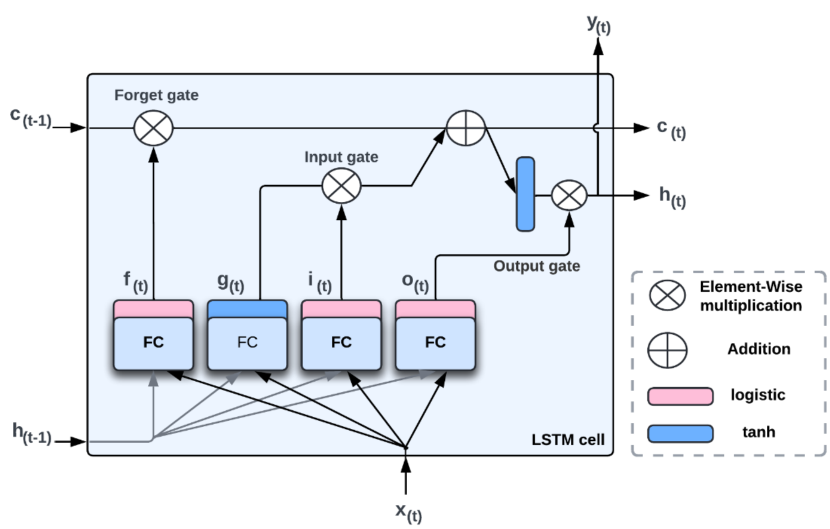

Figure 2: Long-short-term memory (LSTM) networks, a specialized form of recurrent neural networks (RNNs), utilize gated architectures to selectively retain and discard information, enhancing their ability to handle sequential data.

As depicted in Figure 2, a long short-term memory network (LSTM) is a specialized variant of a recurrent neural network (RNN), which broadly refers to neural networks designed to handle sequential data. An LSTM has a distinctive architecture includes three types of "gates." These gates control the flow of information into the network by filtering the incoming data according to preset rules. Information that aligns with these rules is retained, while information deemed irrelevant is discarded via the "forget gate." The gating mechanism functions selectively, with the standard activation function within an LSTM being the sigmoid function, as illustrated in Equation 1. This function outputs a value between 0 and 1, representing the degree to which information can pass through. The LSTM can add or remove information from its neurons using these gates. Furthermore, the LSTM contains a layer with the tanh activation function, as shown in Eq. 2, to refresh the neurons' state.

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{5}$$

$$\tanh(x) = \frac{e^{x} - e^{x}}{e^{x} + e^{x}} \tag{6}$$

The LSTM model will have a similar structure with input and output layers but will include LSTM cells with a more complex internal structure to manage long-term dependencies.

$$i_t = \sigma(W_{xi}\, Tx_{(t)} + W_{hi}\, Th_{(t-1)} + b_i) \tag{7}$$
$$f_t = \sigma\left(W_{xf}\, Tx_{(t)} \,.\, W_{hf}\, Th_{(t-1)} + b_f\right) \tag{8}$$
$$o_t = \sigma(W_{x0}\, Tx_{(t)} \,.\, W_{h0}\, Th_{(t-1)} + b_o) \tag{9}$$
$$g_t = tanh\left(W_{xg} Tx_{(t)} + W_{hg}\, Th_{(t-1)} + b_g\right) \tag{10}$$
$$c_t = f_t c_{t-1} + i_t * tanh\left(W_{xg} Tx_{(t)} + W_{hg}\, Th_{(t-1)} + b_g\right) \tag{11}$$
$$y_t = o_t * tahn(c_t) \tag{12}$$

where f_t is the forget gate, i_t is the input gate, o_t is the output gate, c_t is the state, h_t is the hidden state (Géron, 2019). As shown in Figure 2, the operations are specifically described using from Eq. 7 to Eq 12.

Transformer:

The Transformer model differs by using self-attention mechanisms to process sequences, lacking recurrence entirely.

$$Attention(Q, K, V) = SoftMax\left(\frac{Qk^{t}}{\sqrt{d_k}}\right)V \tag{13}$$

where Q, K, and V are the query, key, and value matrices derived from the input, d_k is the dimension for the keys, and SoftMax operation is applied across the keys for each query.

As depicted in Figure 3, the Transformer model encodes input sequences into vectors and integrates positional information to reflect the order of words, unlike RNNs. Its multi-head attention mechanism (see Eq 13) allows simultaneous focus on various sequence segments, enhancing global dependency recognition. This, along with residual connections and normalization, stabilizes deep network training. After transforming these vectors through a feed-forward network, the model's decoder mimics this process, adding positional encoding and using masked multi-head attention to predict outputs without future data insight. Attention between the encoder and decoder aligns focus on pertinent input aspects, aiding the sequence's contextual understanding. The Transformer's final output predictions are shaped into probabilities after a linear transformation. Adaptable beyond NLP, Transformers apply their intricate attention and pattern recognition capabilities to stock price forecasting, evaluating historical data to predict future market behaviors, potentially outperforming traditional models in precision (Jia, 2019).

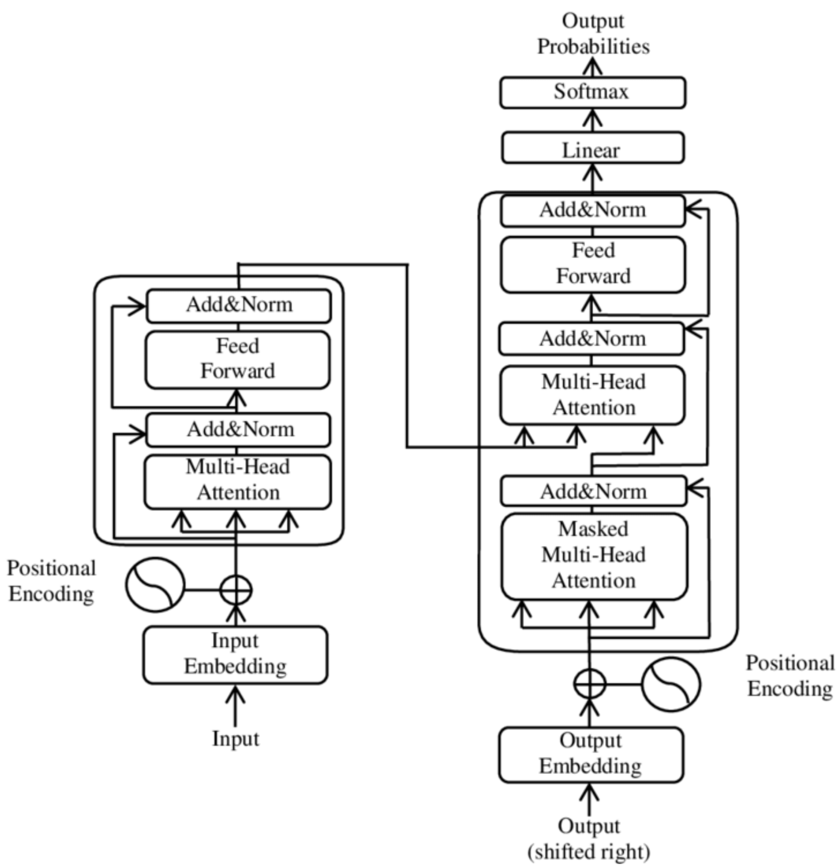

Figure 3: The Transformer model employs multi-head attention and positional encoding to process input sequences efficiently, facilitating global dependency recognition and stable training. Its adaptable architecture extends beyond NLP, showing promise in enhancing precision for stock price forecasting (Jia, 2019).

Metrics

Evaluation metrics included Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R-squared (R^2). These metrics provided a comprehensive view of each model's accuracy and ability to capture underlying trends, which are defined below in Eq. 14 to Eq 16:

$$Mean\ Absolute\ Error = \frac{1}{n}\sum_{i=1}^{n}|y_i - \widehat{y}_i| \tag{14}$$

$$Root\ Mean\ Absolute\ Error = \sqrt{\frac{1}{n}\sum_{i=1}^{n}|y_i - \widehat{y_i}|} \tag{15}$$

$$R^2 = \frac{\frac{1}{n}\sum_{i=1}^{n}|y_i - \widehat{y_i}|^2}{\sum_{i=1}^{n} y_i^2} \tag{16}$$

Training Hypermeters

Each model is trained using the respective training dataset with hyperparameters tuned to optimize performance. The models' training hypermeters are given in Table 1 and Table 2.

Table 1: Comparison of hyperparameters used for GRU and LSTM models in an experiment, highlighting their similarities in optimization, activation, and learning configurations.

Item	Hyper-Parameter of GRU	Hyper-Parameter of LSTM
Optimization	Adam	Adam
Activation Function	ReLU	ReLU
Initial Learning Rate	0.001	0.001
Number of GRU units	64	64
Batch size	32	32
Epoch Number	100	100

Table 2: The hyperparameters chosen for a Transformer model include optimization with Adam and specific architectural configurations.

Item	Hyper-Parameter for Transformer
Optimization	Adam
Initial Learning	0.001
Rate Epoch Number	100
Dimensionality of the embedding space	32
Number of attention heads	2
Dimensionality of the feed-forward layer	32
Number of transformer blocks	3
List of units in the Multi-Layer Perceptron	32
Dropout rate	0.1
Dropout rate in the MLP layers	0
epsilon	1.00E-06

RESULTS

The dataset analysis reveals distinct trends and behaviors across AAPL, CSCO, META, MSFT, and TSLA stocks.

Descriptive Statistics

From the experimental results, as illustrated in Figure 4, the decade-long closing stock prices for five companies: AAPL (blue), CSCO (orange), META (green), MSFT (red), and TSLA (purple), from 2014 to early 2024. All show general growth, but TSLA exhibits significant volatility post-2020, unlike the steadier gains of MSFT and AAPL. CSCO's relatively flat trend indicates underperformance, while META shows a sharp rise and fall around 2020, suggesting market corrections. MSFT and AAPL followed similar trends pre-2020, with TSLA later soaring past them before fluctuating. The chart highlights market volatility, TSLA's rapid growth and instability, and CSCO's modest performance.

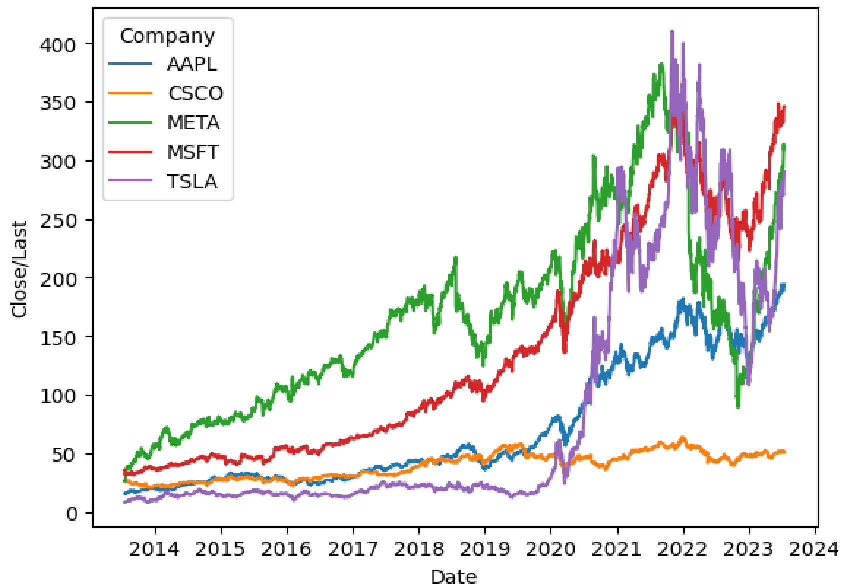

Figure 4: A decade-long comparison of closing stock prices for AAPL, CSCO, META, MSFT, and TSLA reveals overall upward trends, with TSLA exhibiting pronounced volatility post-2020 and contrasting with the steadier growth of MSFT and AAPL. CSCO's flat trend suggests underperformance, while META experienced sharp rises and falls around 2020, reflecting market dynamics and individual company events.

Figure 5 depicts a decade of investor activity for five tech giants, using trading volume as a proxy for market sentiment. Volume spikes, marked by vertical surges, align with events like product launches, earnings reports, or market shifts, reflecting heightened investor reactions. TSLA (purple) dominates with towering peaks, highlighting strong attention and volatility tied to its dynamic news cycle. AAPL (blue) shows consistently high volumes, indicating sustained investor interest. In contrast, CSCO (orange) and MSFT (red) have steadier, lower volumes, reflecting less speculative engagement. META (green) experiences episodic surges during impactful events. The chart showcases the interplay of market events and investor behavior.

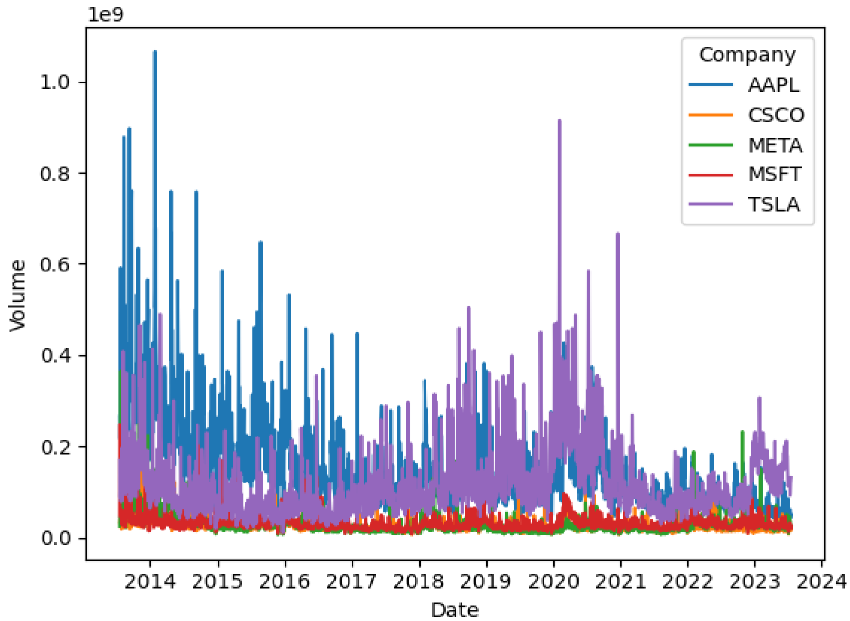

Figure 5: Trading volume dynamics for five tech giants over a decade reveal surges coinciding with pivotal events, showcasing correlations between high-impact news and investor response. TSLA's towering peaks indicate robust attention and volatility, contrasting with AAPL's consistent volume. CSCO and MSFT exhibit steadier profiles, and META experiences episodic crescendos reflecting market reactions to corporate developments.

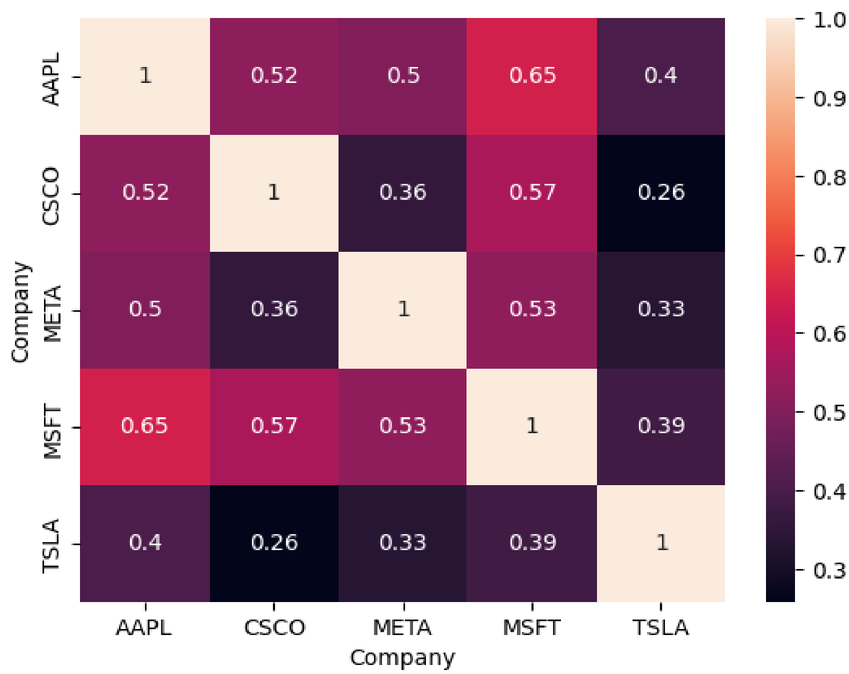

Figure 6: A heatmap visualizing correlations between daily stock price changes for AAPL, CSCO, META, MSFT, and TSLA depicts varied levels of correlation, with AAPL and MSFT showing the strongest positive correlation (0.65) and CSCO and TSLA exhibiting the lowest correlation (0.26), aiding portfolio diversification and strategic decision-making.

The heatmap of **Figure 6** shows the correlations between daily percentage changes in the stock prices of five companies. A correlation of 1 indicates perfect alignment, 0 shows no relationship, and -1 denotes a perfect inverse relationship. AAPL and MSFT exhibit the strongest correlation (0.65), likely due to shared market drivers in the tech sector, while CSCO and TSLA have the weakest correlation (0.26), reflecting their differing industry influences. Such heatmaps help investors manage risk through diversification and analyze market trends to inform strategic decisions.

Comparative Model Performance

The comparative evaluation of GRU, LSTM, and Transformer models demonstrates significant differences.

The GRU model achieved the lowest MAE across all datasets, with outstanding performance for CSCO (MAE: 0.56), as shown in **Table 3**. Its streamlined architecture enabled precise and efficient predictions.

While the LSTM model's MAE was slightly higher than GRU's, LSTM excelled at capturing broader price trends, evidenced by higher R^2 values for AAPL (0.92) and MSFT (0.90).

Although the Transformer leveraged self-attention mechanisms, the Transformer struggled with highly volatile stocks like TSLA, showing higher MAE (e.g., 9.20 for TSLA) and lower R-squared values (e.g., 0.88 for TSLA).

Table 3: The MAE, RMSE, and R^2 of three models (GRU, LSTM, & TRANSFOERMER)

Model	GRU			LSTM			TRANSFORMER		
	MAE	RMSE	R^2	MAE	RMSE	R^2	MAE	RMSE	R^2
AAPL	2.52911	1.59033	0.9177	2.43845	1.56147	0.92119	3.09639	1.75921	0.88342
CSCO	0.55887	0.74758	0.86132	0.61354	0.78347	0.87096	0.77970	0.88324	0.78418
META	8.07122	2.84099	0.62097	5.2274	2.28640	0.85876	6.93982	2.6347	0.75338
MSFT	4.3006	2.0732	0.8880	4.7772	2.1854	0.8697	5.6889	2.3854	0.82845
TSLA	7.4418	2.7275	0.9087	7.9124	2.8126	0.8965	9.2030	3.0338	0.88116

Table 4 compares the GRU model's MAE performance to LSTM and Transformer across five companies. GRU outperforms both models for AAPL and MSFT, with notable improvements such as 32.28% over Transformer for MSFT. For CSCO, GRU outperforms Transformer by 39.47% but is 9.83% behind LSTM. META shows LSTM significantly outperforming GRU by 35.26%, though GRU still surpasses Transformer by 14.04%. In TSLA, GRU beats Transformer by 23.69% but trails LSTM by 6.32%. GRU performs best except for CSCO and TSLA, where LSTM slightly leads, and META, where LSTM dominates.

Table 4: Minimum and Maximum RMSE Values for the Three Models

Model	GRU MAE	LSTM MAE	Transformer MAE	$\frac{\text{GRU} - \text{LSTM}}{GRU} * 100$	$\frac{\text{GRU} - \text{Transformer}}{GRU} * 100$
AAPL	2.52911	2.43820	3.09639	**3.60%**	**-22.43%**
CSCO	0.55887	0.61383	0.77970	**-9.83%**	**-39.47%**
META	8.07122	5.22763	6.93982	**35.26%**	**14.04%**
MSFT	4.30064	4.77796	5.68892	**-11.08%**	**-32.28%**
TSLA	7.44184	7.91210	9.20304	**-6.32%**	**-23.69%**

Table 5: The MAE, RMAE, and R2 of Three Different Models (GRU, LSTM, & TRANSFOERMER)

Model	GRU RMSE		LSTM RMSE		TRANSFORMER RMSE	
	MIN	MAX	MIN	MAX	MIN	MAX
AAPL	2.97128	104.707	2.95009	150.436	3.67839	15.4198
CSCO	0.68980	20.6755	0.67028	39.6035	0.86884	30.1045
META	11.4971	120.165	5.67821	27.6134	10.1143	151.109
MSFT	5.67906	22.2298	12.4912	145.810	9.94773	150.690
TSLA	8.76743	93.7775	10.4482	249.601	8.96747	175.762

Table 5 presents RMSE data for GRU, LSTM, and Transformer models across multiple firms, where lower values indicate better predictions. LSTM performs well for AAPL, with a slightly lower minimum RMSE but a higher maximum than GRU. For CSCO and META, LSTM is the most accurate and consistent, with the lowest RMSE values. GRU shows better average performance for MSFT with the lowest minimum RMSE, while LSTM's lower maximum indicates better outlier handling. Similarly, GRU achieves the lowest minimum RMSE for TSLA, but LSTM's lower maximum suggests better handling extreme cases. Overall, LSTM consistently delivers lower RMSE values, indicating more reliable performance in stock price prediction.

CONCLUSIONS

The study's results highlight critical insights into the suitability of different deep learning models for stock price prediction.

The GRU model consistently outperformed the other architectures in precision, making it ideal for applications requiring high accuracy. Its ability to handle long-term dependencies while maintaining a simpler architecture than LSTM contributed to its efficiency.

The LSTM model demonstrated strengths in capturing temporal patterns, particularly for stocks with stable historical trends. Its slightly higher error margins than GRU suggest potential challenges in real-time applications but make it valuable for strategic trend analysis.

Despite its innovative self-attention mechanism, the Transformer's performance was hindered by the volatility and noisiness of stock market data. Additional tuning or hybridization with recurrent layers might improve its utility in this domain.

These findings align with studies like Lin *et al.*, (2021) and Tao *et al.*, (2024), emphasizing the strengths of recurrent models in time-series analysis. This study's contribution lies in benchmarking these models within the technology sector and highlighting their specific strengths and weaknesses.

These findings provide valuable insights for practitioners and researchers, emphasizing the importance of aligning model selection with specific forecasting objectives. Future research could explore the development of hybrid models that combine the strengths of GRU and LSTM, leveraging GRU's precision for less volatile data and LSTM's ability to capture broader trends. Incorporating sentiment analysis and macroeconomic indicators, such as public opinion on social media or GDP growth rates, could enrich the predictive models by accounting for market sentiment and broader economic conditions.

Expanding these approaches to multi-sectoral datasets would further generalize their utility. For example, models could be applied to healthcare companies like Pfizer, whose stock prices may respond to drug approvals, or energy firms like ExxonMobil, influenced by oil price fluctuations. Similarly, integrating consumer goods data from companies like Amazon, shaped by consumer spending trends, or financial institutions like JPMorgan, affected by interest rate changes, could broaden the models' applicability. Such cross-sector analysis would enhance their versatility and impact in financial forecasting, enabling deeper insights across diverse industries.

Supplementary Material

The data, code, and additional material can be found here:
https://github.com/aman5196/Decoding-Stock-Trends-Data-code-

Reference

Lin et al., (2021) and Abu-Mostafa, Y. S., & Atiya, A. F. (1996). Introduction to financial forecasting. Applied Intelligence, 6(3), 205–213. https://doi.org/10.1007/BF00126626

Ali, M., Khan, D. M., Alshanbari, H. M., & El-Bagoury, A. A.-A. H. (2023). Prediction of Complex Stock Market Data Using an Improved Hybrid EMD-LSTM Model. Applied Sciences, 13(3), 1429. https://doi.org/10.3390/app13031429

Beniwal, M., Singh, A., & Kumar, N. (2024). Forecasting multistep daily stock prices for long-term investment decisions: A study of deep learning models on global indices. Engineering Applications of Artificial Intelligence, 129, 107617. https://doi.org/10.1016/j.engappai.2023.107617

Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling (Version 1). arXiv. https://doi.org/10.48550/ARXIV.1412.3555

Ghosh, P., Neufeld, A., & Sahoo, J. K. (2022). Forecasting directional movements of stock prices for intraday trading using LSTM and random forests. Finance Research Letters, 46, 102280. https://doi.org/10.1016/j.frl.2021.102280

Henrique, B. M., Sobreiro, V. A., & Kimura, H. (2019). Literature review: Machine learning techniques applied to financial market prediction. Expert Systems with Applications, 124, 226–251. https://doi.org/10.1016/j.eswa.2019.01.012

Hu, J. (2021). Local-constraint transformer network for stock movement prediction. International Journal of Computational Science and Engineering, 24(4), 429. https://doi.org/10.1504/IJCSE.2021.117030

Jia, Y. (2019). Attention Mechanism in Machine Translation. Journal of Physics: Conference Series, 1314(1), 012186. https://doi.org/10.1088/1742-6596/1314/1/012186

Li, C., & Qian, G. (2022). Stock Price Prediction Using a Frequency Decomposition Based GRU Transformer Neural Network. Applied Sciences, 13(1), 222. https://doi.org/10.3390/app13010222

Lin, Y., Yan, Y., Xu, J., Liao, Y., & Ma, F. (2021). Forecasting stock index price using the CEEMDAN-LSTM model. The North American Journal of Economics and Finance, 57, 101421. https://doi.org/10.1016/j.najef.2021.101421

Lu, W., Li, J., Li, Y., Sun, A., & Wang, J. (2020). A CNN-LSTM-Based Model to Forecast Stock Prices. Complexity, 2020, 1–10. https://doi.org/10.1155/2020/6622927

Mian, T. S. (2023). Evaluation of Stock Closing Prices using Transformer Learning. Engineering, Technology & Applied Science Research, 13(5), 11635–11642. https://doi.org/10.48084/etasr.6017

Qiao, R., Chen, W., & Qiao, Y. (2022). Prediction of stock return by LSTM neural network. Applied Artificial Intelligence, 36(1), 2151159. https://doi.org/10.1080/08839514.2022.2151159

Shahi, T. B., Shrestha, A., Neupane, A., & Guo, W. (2020). Stock Price Forecasting with Deep Learning: A Comparative Study. Mathematics, 8(9), 1441. https://doi.org/10.3390/math8091441

Tao, Z., Wu, W., & Wang, J. (2024). Series decomposition Transformer with period-correlation for stock market index prediction. Expert Systems with Applications, 237, 121424. https://doi.org/10.1016/j.eswa.2023.121424

Wu, D., Wang, X., & Wu, S. (2022). Jointly modeling transfer learning of industrial chain information and deep learning for stock prediction. Expert Systems with Applications, 191, 116257. https://doi.org/10.1016/j.eswa.2021.116257

Yu, X., Wu, W., Liao, X., & Han, Y. (2023). Dynamic stock-decision ensemble strategy based on deep reinforcement learning. Applied Intelligence, 53(2), 2452–2470. https://doi.org/10.1007/s10489-022-03606-0

Yu, Y., & Kim, Y.-J. (2019). Two-Dimensional Attention-Based LSTM Model for Stock Index Prediction. Journal of Information Processing Systems, 15(5), 1231–1242. https://doi.org/10.3745/JIPS.02.0121

Zhou, F., Zhang, Q., Zhu, Y., & Li, T. (2023). T2V_TF: An adaptive timing encoding mechanism based Transformer with multi-source heterogeneous information fusion for portfolio management: A case of the Chinese A50 stocks. Expert Systems with Applications, 213, 119020. https://doi.org/10.1016/j.eswa.2022.119020