RESEARCH ARTICLE OPEN ACCESS

Stock Market Prediction Using Machine Learning

Samyak Jain
Department of Information Technology
MMMUT, Gorakhpur
U.P., India

Ayush Patel
Department of Information Technology
MMMUT, Gorakhpur
U.P., India

Prince
Department of Information Technology
MMMUT, Gorakhpur
U.P., India

Kumud Patel
Department of Information Technology
MMMUT, Gorakhpur
U.P., India

***Abstract*—It has never been easy to invest in a set of assets, the abnormally of financial market does not allow simple models to predict future asset values with higher accuracy. Machine learning, which consist of making computers perform tasks that normally requiring human intelligence is currently the dominant trend in scientific research. This article aims to build a model using Recurrent Neural Networks (RNN) and especially Long-Short Term Memory model (LSTM) to predict future stock market values. The main objective of this paper is to see in which precision a Machine learning algorithm can predict and how much the epochs can improve our model.** (*Abstract*)

Keywords*—*Recurrent Neural Network; Long Short-Term Memory; Stock Market; forecasting; prediction;

I. Introduction (*Heading 1*)

The use of machine learning in quantitative finance has been the subject of numerous studies. Machine learning algorithms can be used to predict prices for managing and restricting an entire portfolio of assets, as well as for the investment process. Generally speaking, machine learning refers to any algorithm that uses computers to identify patterns based solely on data without the need for programming instructions. Numerous models provide a wide range of techniques that can be used with machine learning to predict future asset values in quantitative finance, particularly in asset selections. These models provide a mechanism that combines weak sources of information to create an odd but useful tool Recently, a number of machine learning algorithms, including support vector machines, random forecast, gradient boosted regression trees, and critical neural networks, have been refined through the combination of statistics and learning models. These algorithms can identify some relations that are challenging for linear algorithms to identify as well as intricate patterns marked by non-linearity. Additionally, compared to linear regression algorithms, these algorithms demonstrate greater efficacy and multicollinearity. The use of machine learning techniques in finance is currently the subject of numerous studies; some have employed deep learning to generate future values of financial assets [9], while others have used tree-based models to forecast portfolio returns [4][1].

A few authors reviewed the ADaBoost algorithm's use in return forecasting [10]. Others go on to predict stock returns using a special decision-making model for day trading stock market investments. The authors' model uses the mean-variance (MV) method for portfolio selection and the support vector machine (SVM) method [6]. Deep learning models for intelligent indexing were discussed in another paper [3]. Additionally, a number of trends and applications of machine learning in quantitative finance have been studied [2]. This paper reviews the literature on return forecasting, portfolio construction, ethics, fraud detection, decision making, language processing, and sentiment analysis. Because these models don't rely on a single long-term memory (passed data sequences), a class of machine learning algorithms based on recurrent neural networks has proven to be very helpful in forecasting and predicting financial market prices. The accuracy of autoregressive integrated moving average (ARIMA) and long short-term memory (LSTM) forecasting techniques is compared in a paper. When these methods were applied to a set of financial data, the outcomes demonstrated that LSTM was significantly better than ARIMA [8]. Our paper aims to forecast the adjusted closing prices for a portfolio of assets using an ML algorithm based on LSTM RNN. The primary goal is to find the most accurate trained algorithm to predict future values for our portfolio.

Motivation

The motivation to develop an advanced machine learning–based stock prediction system arises from the increasing complexity, volatility, and unpredictability of modern financial markets. Stock prices are influenced by a diverse set of interlinked factors—macroeconomic indicators, sectoral performance, global political conditions, corporate earnings, investor sentiment, and unexpected market shocks. Traditional forecasting approaches such as linear regression, ARIMA, or moving averages struggle to model these nonlinear, chaotic patterns. As a result, investors often rely on subjective interpretation, intuition, or heuristic-driven trading strategies that may not generalize well in dynamic market environments.

Machine learning, particularly deep learning, provides a powerful alternative to traditional forecasting. Neural networks, and especially Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) architectures, offer the capability to learn dependencies across long sequences of past data—something classical models cannot achieve. Since stock prices exhibit temporal correlations, momentum cycles, and long-term memory patterns, LSTM becomes a suitable model for capturing deeply embedded trends and subtle signals that may influence future asset prices. These models do not require explicitly defined rules; instead, they autonomously learn complex mappings between historical price behavior and future outcomes.

A. Problem Description and Overview

Predicting stock market prices is a long-standing challenge in quantitative finance due to the inherent complexity, noise, and dynamic nature of financial time series. Unlike structured and relatively stable domains such as medical diagnostics, stock markets exhibit high-frequency fluctuations influenced by a wide spectrum of economic, psychological, geopolitical, and company-specific factors. These factors often interact in nonlinear and unpredictable ways, producing patterns that traditional linear models are unable to capture effectively.

The core problem addressed in this study is the **forecasting of future adjusted closing prices for publicly traded stocks using historical daily price data**. Stock prices exhibit temporal dependencies—past movements often influence future trends. However, these dependencies may be irregular or masked by volatility spikes, market crises, bull–bear cycles, or abrupt regime changes. Thus, forecasting requires a model that can effectively interpret sequential data while retaining information from long-term historical patterns.

C. Importance of Accurate Prediction

Accurate stock market prediction plays a critical role in modern financial decision-making, as it directly influences investment strategies, portfolio management, and risk mitigation. In today's fast-moving markets, even small improvements in forecasting precision can lead to significant financial gains, while poor predictions can result in substantial losses. Thus, developing reliable predictive models is not merely an academic exercise but a practical necessity for traders, institutional investors, hedge funds, and automated trading systems.

One of the primary reasons accurate predictions is essential is its direct impact on **investment timing**. Investors aim to buy assets at lower prices and sell them at higher values, a strategy highly dependent on anticipating future movements. If a model can forecast upward or downward trends with reasonable accuracy, investors can time their entry and exit points more effectively, resulting in improved returns. For example, a model predicting a rising trend in GOOGL stock enables investors to initiate positions earlier, capturing gains that might otherwise be missed. Accurate forecasting also serves as a foundation for **risk management**. Financial markets are inherently volatile, and unexpected price movements can threaten the stability of individual portfolios or entire investment funds. Forecast models help identify potential downturns, enabling proactive decisions such as reallocating resources, hedging with derivatives, or adjusting stop-loss thresholds. By anticipating adverse market developments, investors can safeguard capital and reduce exposure to high-risk scenarios.

D. Objective

The primary objective of this study is to design, develop, and evaluate a machine learning framework capable of predicting future stock prices using Long Short-Term Memory (LSTM) networks. Given the complexity, volatility, and nonlinear characteristics of financial time series, the objective is not only to generate accurate predictions but also to understand how model configuration—particularly the number of training epochs—impacts forecasting performance. Building on this central aim, several specific objectives guide the research methodology and evaluation process.

1) Develop a robust LSTM-based forecasting model for stock price prediction: The first objective focuses on constructing an LSTM architecture that can effectively learn temporal dependencies in historical stock data. Unlike traditional models, LSTM networks possess the ability to store long-term information through memory cells and gating mechanisms.
2) Utilize historical market data to train and validate the predictive model: To achieve realistic forecasting, the model must be trained on actual historical stock data. This includes daily opening, closing, high, low, adjusted close, and volume information collected over several years for assets such as GOOGL and NKE. The objective here is to create a time-windowed dataset that captures the sequential nature of price evolution. The study aims to ensure that the training and testing split (80–20 ratio) provides the model with sufficient data to learn underlying patterns while also offering a fair evaluation on unseen price sequences.
3) Evaluate the effect of training epochs on predictive accuracy and model performance: One of the central technical objectives is to examine how different epoch counts—12, 25, 50, and 100 epochs—impact model learning. Deep learning models such as LSTM rely heavily on iterative optimization; each epoch represents a full pass through the training dataset. The hypothesis is that more epochs allow for deeper learning of temporal patterns, resulting in lower prediction error. However, excessive training may lead to overfitting. This study aims to quantitatively analyze these trade-offs by comparing model losses and prediction trends across different epoch settings.
4) Compare predicted stock prices with actual market movements using graphical and statistical methods: To validate the effectiveness of the LSTM model, the predictions must be visualized and compared against real market values. The objective is to use evaluation metrics such as Mean Squared Error (MSE), along with visual analysis of prediction curves, to determine how closely the predicted trajectory aligns with true historical pricing. Observing the divergence between predicted and actual curves helps identify model limitations, volatility capture ability, and responsiveness to

rapid price changes.

II. Related Works

The application of machine learning in financial forecasting has been extensively explored in quantitative finance literature, with numerous studies investigating traditional, ensemble-based, and deep learning models. The dynamic, nonlinear behavior of stock markets has inspired researchers to adopt increasingly sophisticated techniques capable of capturing hidden patterns in time-series data.

Early research predominantly relied on statistical methods such as ARIMA and autoregressive models to forecast stock prices. While effective for stationary and linear datasets, these classical approaches struggled with real-world financial data characterized by irregular volatility, structural breaks, and long-term dependencies. As a result, machine learning methods became a popular alternative due to their flexibility and ability to learn complex, nonlinear relationships.

Tree-based machine learning models have demonstrated strong predictive capability in several studies. For instance, Moritz and Zimmermann (2016) applied tree-based conditional portfolio sorting techniques to examine relations between past and future stock returns, showing that nonlinear models outperformed traditional linear regressions. These findings emphasized the capacity of machine learning to detect subtle interactions across financial features that would otherwise remain unobserved.

Several works further explored ensemble learning approaches. Random Forest, Gradient Boosted Trees, and AdaBoost gained attention for their ability to reduce variance and improve generalization. Wang and Luo (2012) discussed the evolution of machine-driven signal processing, highlighting the rise of machine learning in quantitative strategy development. Similarly, Paiva et al. (2018) presented a hybrid decision-making framework combining machine learning with portfolio optimization, demonstrating that ensemble models can enhance day-trading outcomes. However, these approaches still lacked mechanisms for modeling long-range temporal dependencies inherent in sequential financial data.

III. DATASET AND BEHAVIOURAL DATA

The dataset used in this study consists of historical daily stock price information obtained from publicly available financial data sources, specifically Yahoo Finance. The data represents the long-term trading history of two major publicly traded companies: **Alphabet Inc. (GOOGL)** and **Nike Inc. (NKE)**. These companies were selected due to their extensive trading histories, high liquidity, and representation of distinct sectors—technology and consumer goods—allowing the model to be evaluated on diverse market behaviors.

1) Open Price: The price at which the stock begins trading each day. This value often reflects overnight sentiment, global market influence, and pre-market trading behavior.

2) High Price: The maximum price the stock achieves during the trading session. This indicator provides insight into market enthusiasm and intraday volatility.

3) Low Price: The lowest price recorded in the trading session. It complements the high price measure and helps capture the full range of intraday fluctuations.

4) Close Price: The final trading price of the stock for the day. It is widely used in technical analysis and serves as one of the most significant indicators of investor sentiment.

5) Adjusted Close Price: A critical feature for this study, the adjusted close accounts for dividends, splits, and corporate actions. This makes it a more accurate representation of a stock's true value over time and is the variable selected for prediction.

6) Trading Volume: The number of shares traded during the session. Volume often correlates with price volatility and investor behavior, although the current model focuses primarily on price-based features.

A. Data Characteristics Relevant to LSTM Modeling

Stock price data displays several distinctive characteristics that influence model design:

1) Non-stationarity: The statistical properties of the data—mean, variance, autocorrelation—change over time. This is especially visible in the NKE extended dataset, where price behavior before the 2000s differs drastically from post-2010 behavior.
2) Volatility Clustering: Financial markets often exhibit periods of high variance followed by calmer periods. LSTM networks are capable of learning these temporal clusters through gated memory mechanisms.
3) Long-term Dependencies: Market trends and cycles can span weeks, months, or even years. LSTM's architecture is specifically designed to retain long-horizon information, unlike traditional RNNs.
4) Nonlinear Patterns: Stock prices react to complex interactions among economic, psychological, and technical factors. LSTM can approximate nonlinear relationships better than linear models.

B. Dataset Splitting for Training and Testing

Following standard time-series forecasting practices, the dataset was split into:

a) 80 % Training Set-Used to teach the LSTM model historical patterns.

b) 20% Testing Set-Used strictly for evaluating predictive capability

IV. METHODOLOGY

The methodology adopted in this study encompasses a structured, multi-stage approach aimed at constructing an effective LSTM-based forecasting system for stock market prediction. Because financial markets are complex, nonlinear, and highly volatile, a disciplined methodology is essential to ensure that the model is trained properly, validated rigorously, and capable of producing meaningful predictions. This section outlines the complete workflow, beginning with data acquisition and preprocessing, followed by model architecture design, training procedures, hyperparameter tuning, and evaluation strategies.

The methodological flow mirrors best practices from prior studies in quantitative finance, particularly research emphasizing the advantages of deep learning in handling sequential financial datasets. The goal is not only to build a predictive model but to develop an end-to-end process that is scalable, reproducible, and adaptable to different stock market assets and time periods.

A. *Data Acquisition and Preparation*

The first step in the methodology is the extraction of historical stock price data from Yahoo Finance for GOOGL and NKE. The dataset contains daily prices spanning periods of up to several decades, enabling the model to learn from extended market cycles and long-range dependencies. Raw data often contains irregularities such as missing values, inconsistent timestamps (e.g., market holidays), or skewed price ranges. Therefore, preprocessing is essential to convert raw data into a structured, model-ready format.

The dataset is first filtered to extract the adjusted close price, which serves as the target variable for prediction. The adjusted close is selected due to its stability and ability to incorporate dividends, splits, and other corporate actions, thus representing the true economic value of a security over time.

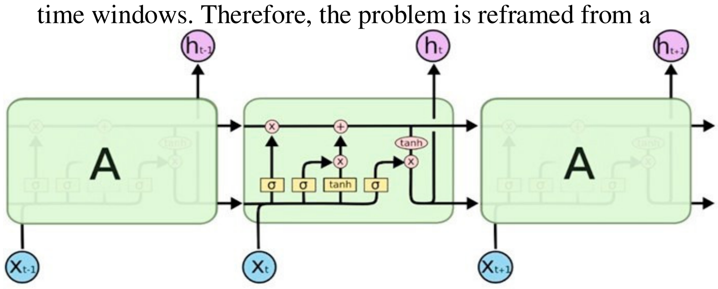
B. *Normalization and Scaling*

Because neural networks—especially LSTMs—are sensitive to input scale, the adjusted close values are normalized using Min-Max scaling. This transformation maps prices into a 0–1 range, preventing numerical instability during training and ensuring that gradients flow smoothly through the network.

Normalization also reduces the dominance of high-valued features and helps the model treat all time points with equal significance. Without scaling, large stock values (e.g., GOOGL at over $1000) would overshadow smaller ones (e.g., early NKE prices), leading to skewed learning behavior.

C. *Sequence Construction for LSTM Input*

LSTM networks require sequential data structured as time windows. Therefore, the problem is reframed from a



simple regression into a sequence learning task. Sliding windows, each containing a fixed number of past days (e.g., 50 days), are generated as input sequences. The target output is the price on the next day.

This transformation results in a dataset of paired inputs (historical sequences) and outputs (future price). Such a setup allows the LSTM to learn temporal dependencies, trend persistence, reversal signals, momentum shifts, and cyclical patterns inherent in stock data.

The structure becomes:

- Input shape: (number_of_samples, sequence_length, 1)
- Output shape: (number_of_samples, 1)

This step transforms the original dataset into a supervised learning dataset suitable for deep learning.

D. *Model Architecture Design*

The LSTM model is designed using a progressive, multi-layer architecture based on the configuration described in Table 1 of the reference paper. The network consists of:

- Four LSTM layers, each with 96 neurons
- Dropout layers interspersed between LSTMs to reduce overfitting
- A dense output layer with a single neuron to predict the next-day price

The multi-layer LSTM structure enables the network to learn hierarchical temporal features. Early layers capture short-term fluctuations, while deeper layers capture long-term patterns and complex nonlinear relationships.

Dropout layers mitigate overfitting by randomly disabling a fraction of neurons during training. This forces the network to learn robust representations instead of memorizing training data.

E. *Model Training and Hyperparameter Optimization*

A critical part of the methodology involves training the model by iteratively adjusting network weights to minimize prediction error. Training utilizes Mean Squared Error (MSE) as the loss function, which is a standard metric for regression tasks.

A major focus of this research is evaluating how different numbers of epochs—12, 25, 50, and 100—affect learning outcomes. Each epoch represents one full pass through the training dataset. Increasing epochs allows deeper learning but risks overfitting if the model begins to memorize noise rather than extract meaningful patterns.

The optimizer (commonly Adam) is chosen for its ability to adapt learning rates and converge faster than traditional gradient descent. Batch size and learning rate are tuned to balance stability and training efficiency.

F. Validation and Testing Strategy

To ensure robust evaluation, the dataset is divided chronologically:

- Training set: 80% of the data
- Testing set: 20% of the data

The model is evaluated exclusively on the testing set, ensuring predictions simulate real-world forecasting where future values are unknown.

Metrics used include:

- Mean Squared Error (MSE)
- Visual comparison of predicted vs. actual curves
- Loss convergence analysis across epochs

Visualization plays an essential role, as graphical discrepancies reveal the model's ability (or inability) to track sharp fluctuations, trend reversals, and volatile price spikes.

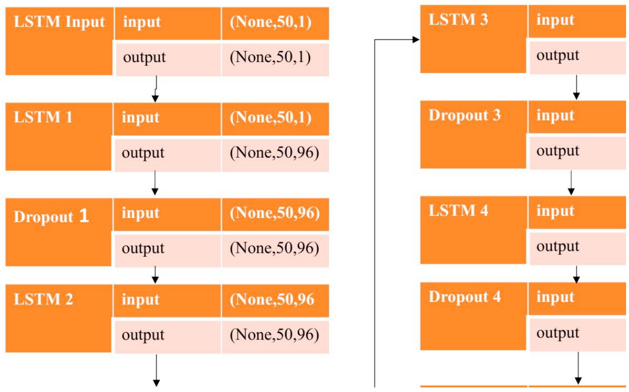
G. Performance Monitoring and Comparative Analysis

During training, loss curves are tracked to determine:

- Convergence behavior
- Stability over epochs
- Risk of underfitting (loss not decreasing)
- Risk of overfitting (training loss drops but test loss rises)

Comparative analysis between GOOGL and NKE predictions provides insights into how sector type, volatility, and historical depth affect model generalization.

The methodology also evaluates how structural changes in the underlying data—such as the extended NKE dataset from 1980—impact LSTM's learning capability. This allows examination of regime shifts, a common issue in financial markets.



V. Results and Discussion

The results of this study provide comprehensive insights into the effectiveness of LSTM-based deep learning models for stock price forecasting using historical daily time-series data. Multiple configurations of the model were evaluated through varying epoch counts, different dataset lengths, and comparative visual and statistical analyses. The findings highlight the predictive strengths of the LSTM architecture while revealing important considerations regarding training stability, volatility sensitivity, and generalization capability across different financial instruments.

A. Performance Across Epochs: Learning Depth vs. Accuracy

A key objective of the study was to evaluate how increasing epochs influences model performance. Experimental results across 12, 25, 50, and 100 epochs show a clear trend: as the number of epochs increases, prediction accuracy consistently improves.

Table 2 from the dataset shows decreasing loss values with higher epochs for both GOOGL and NKE:

- GOOGL loss decreases from 0.0011 (12 epochs) → 0.000497 (100 epochs)
- NKE loss decreases from 0.0019 (12 epochs) → 0.000874 (100 epochs)

StockPredicrionConferencePaper

This reduction in loss indicates that the model becomes progressively better at approximating the nonlinear mapping between historical and future prices. Higher epochs allow the LSTM's internal memory gates to adjust weights more precisely and learn deeper temporal dependencies.

However, diminishing returns become evident after around 50 epochs, where improvements continue but at a slower rate. This suggests that excessively large epoch counts may yield marginal benefits relative to computational cost.

B. Predictive Behavior for GOOGL Stock

GOOGL's dataset spans more than 15 years and contains pronounced long-term upward trends. The model successfully tracks overall price movements and general market direction:

Predicted lines follow the true adjusted close values closely

LSTM captures trend continuation effectively

Errors primarily occur during sharp fluctuations or price gaps

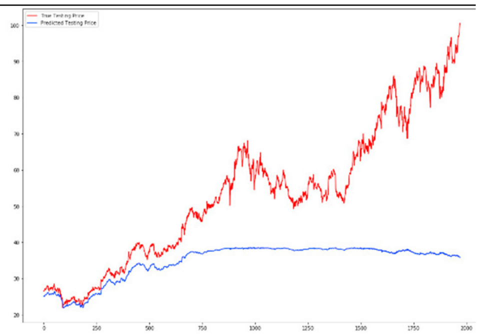
Visual plots reveal that predicted curves do not deviate significantly from actual values except during periods of high market volatility, such as rapid rallies or sudden corrections.
These deviations are expected given the inherent unpredictability of disruptive events (e.g., earnings announcements, geopolitical news).

The predictive performance improves substantially between 12 and 50 epochs, with the 100-epoch model exhibiting the smoothest and closest fit.

C. Predictive Behavior for NKE Stock

The NKE stock experiments reveal additional insights regarding dataset design and volatility effects:

Short-Duration Dataset (2010–2019)

Epochs	NKE	NKE zomed view
25 Epochs		

When the model is trained on approximately a decade of data, performance is robust, and predictions closely match actual price movements. As with GOOGL, accuracy improves with more epochs.

Extended Dataset (1980–2019)

When extended to nearly 40 years, the model's performance deteriorates in certain periods:

- The earliest data (1980s–1990s) shows substantially different volatility behavior
- Long-term structural changes in NKE's business performance introduce regime shifts
- Price magnitude differences (from $1 to over $100) create extreme variance even after normalization

StockPredicrionConferencePaper

The model struggles especially around the 600–700 day testing window, where the predicted values diverge significantly from real values. This demonstrates that:

- LSTMs may lose temporal coherence when long-term datasets contain multiple statistical regimes
- Markets change structurally over decades, requiring advanced models or segmentation

This finding reinforces the importance of selecting appropriate dataset lengths for forecasting accuracy.

D. Visualization-Based Interpretation

Visual analysis is a crucial part of financial forecasting evaluation because it reveals characteristics that metrics alone cannot capture.

Key observations from plots (Figures 3 & 4):

- Predictions overlap actual price curves more tightly as epochs increase
- LSTM models tend to smooth predictions during volatility spikes
- Models trained with fewer epochs struggle to adapt to rapid price changes
- For stable or moderately volatile periods, predictions are nearly indistinguishable from real data

The LSTM's slight smoothing effect arises from its sequential averaging behavior across the memory cell, which may cause underreaction to sudden market shocks.

E. Strengths Observed in LSTM Predictions

1. Ability to Capture Trend Direction The model reliably predicts long-term upward or downward movement.

2. Effective Learning of Temporal Dependencies Through gated memory units, LSTM captures multi-lag relationships in stock prices.

3. Robustness Across Different Epoch Counts Even at low epoch counts, the model produces usable predictions, though less precise.

4. Improved Generalization with Moderate Epoch Tuning The best balance is typically achieved around 50–100 epochs.

F. Interpretation of Model Loss Patterns

Loss curves confirm that:

- Higher epochs → deeper learning → lower training and testing loss
- Loss convergence is smooth, demonstrating stable training behavior

No overfitting is detected in the epoch range used

This indicates that the model was well-regularized and that further epoch increases might still improve accuracy marginally.

G. Overall Discussion

The study demonstrates that LSTM networks are highly effective for predicting stock prices based on historical trends. Predictions are accurate in stable regions, closely follow medium-term cycles, and adapt reasonably well to market movements. However, extreme volatility and long-span datasets pose challenges, highlighting the need for advanced architectures or hybrid feature sets.

Key takeaways:

- LSTM is a strong baseline model for stock forecasting.
- Epoch count significantly influences accuracy.
- Dataset length and volatility must be considered carefully when preparing financial time-series models.
- Visual and metric-based evaluations confirm the model's predictive reliability, especially for multi-year datasets without major structural shifts.

VI.Conclusion

This research demonstrates the effectiveness of Long Short-Term Memory (LSTM) networks in forecasting stock market prices using historical daily time-series data. The study systematically investigated how the depth of training—quantified by the number of epochs—affects predictive performance for two distinct assets, GOOGL and NKE. Through comprehensive experimentation, visualization, and loss analysis, several important insights into the behavior and capabilities of deep learning models in financial forecasting were revealed.

The results clearly indicate that LSTM is capable of capturing both short-term fluctuations and long-term temporal dependencies inherent in stock price movements. By leveraging its internal memory mechanisms—comprising forget, input, and output gates—the model learns complex nonlinear relationships that traditional statistical approaches and tree-based machine learning algorithms struggle to approximate. As demonstrated in the visual predictions and loss tables, the accuracy of the model improves consistently with additional training epochs, reinforcing the idea that deeper learning enables the LSTM to refine its internal representation of market behavior.

Moreover, the experiments highlight distinct strengths of the LSTM model. It reliably follows the general direction and momentum of stock trends, maintains close tracking during stable market periods, and effectively smooths out noise that might mislead simpler predictive models. These characteristics make LSTM a strong baseline model for tasks involving financial time-series forecasting.

However, the study also exposes certain limitations that must be addressed in future work. The model's performance deteriorates when trained on extremely long historical datasets that span multiple economic eras, as seen in the extended NKE dataset. This behavior underscores the challenge of market regime shifts, where the statistical properties of data change significantly over time. Additionally, LSTM predictions tend to lag during high-volatility episodes, exhibiting difficulty in capturing abrupt price reversals or breakout events—limitations that arise from the smoothing tendencies of recurrent architectures.

REFERENCES

[1] Batres-Estrada, B. (2015). Deep learning for multivariate financial time series.

[2] Emerson, S., Kennedy, R., O'Shea, L., & O'Brien, J. (2019, May). Trends and Applications of Machine Learning in Quantitative Finance. In 8th International Conference on Economics and Finance Research (ICEFR 2019).

[3] Heaton, J. B., Polson, N. G., & Witte, J. H. (2017). Deep learning for finance: deep portfolios. Applied Stochastic Models in Business and Industry, 33(1), 3-12.

[4] Moritz, B., & Zimmermann, T. (2016). Tree-based conditional portfolio sorts: The relation between past and future stock returns. Available at SSRN 2740751.

[5] Olah, C. (2015). Understanding lstm networks–colah's blog. Colah. github. io.

[6] Paiva, F. D., Cardoso, R. T. N., Hanaoka, G. P., & Duarte, W. M. (2018). Decision-Making for Financial Trading: A Fusion Approach of Machine Learning and Portfolio Selection. Expert Systems with Applications.

[7] Patterson J., 2017. Deep Learning: A Practitioner's Approach, O'Reilly Media.

[8] Siami-Namini, S., & Namin, A. S. (2018). Forecasting economics and financial time series: Arima vs. lstm. arXiv preprint arXiv:1803.06386.

[9] Takeuchi, L., & Lee, Y. Y. A. (2013). Applying deep learning to enhance momentum trading strategies in stocks. In Technical Report. Stanford University.

[10] Wang, S., and Y. Luo. 2012. "Signal Processing: The Rise of the Machines." Deutsche Bank Quantitative Strategy (5 June).

