Proceedings of CONF-MLA 2026 Symposium: Explainable Computing, Modeling & Data Science in Complex Systems
DOI: 10.54254/2755-2721/2026.GF35824

Deep Learning for VIX Volatility Forecasting: A Comparative Study of LSTM, BiLSTM, and CNN-LSTM Architectures

Siyi Zheng

The University of Hong Kong, Pokfulam, Hong Kong, China
siyi.zheng.712@gmail.com

Abstract. This study evaluates the predictive performance of three deep learning architectures: Long Short-Term Memory (LSTM), Bidirectional LSTM (BiLSTM), and a hybrid Convolutional Neural Network-LSTM (CNN-LSTM), for forecasting the CBOE Volatility Index (VIX). Using daily VIX data from January 2000 to June 2024, we benchmark these models using mean absolute error (MAE), mean squared error (MSE), and the coefficient of determination (R2). The CNN-LSTM model consistently outperforms both unidirectional and bidirectional LSTMs, achieving the lowest MAE (1.758), MSE (9.033), and highest R2 (0.872). Contrary to expectations, the BiLSTM performs worst among the three, with an R2 of 0.670, and this suggests that bidirectional information flow may introduce noise rather than improve accuracy for volatility forecasting. The results also indicate that deep learning models maintain predictive stability during periods of market turbulence, with CNN-LSTM demonstrating the strongest resilience. These findings have practical implications for real-time risk monitoring systems in volatile financial markets.

Keywords: Volatility forecasting, VIX, LSTM, BiLSTM, CNN-LSTM, deep learning, risk management

1. Introduction

The CBOE Volatility Index (VIX) measures the market's expectation of short-term volatility derived from S&P 500 index options and serves as a forward-looking indicator of market uncertainty. Unlike price-based metrics, the VIX embeds investor expectations and downside risk perceptions, making it central to asset pricing, derivative valuation, and systemic risk monitoring. Empirically, the VIX exhibits three challenging characteristics: pronounced volatility clustering, abrupt spikes during crisis episodes, and asymmetric responses to negative market shocks.

Traditional econometric models such as GARCH-family models capture volatility clustering effectively under stable conditions but rely on assumptions of linearity and error normality that fail during periods of market stress. These limitations motivate people to explore data-driven approaches that learn nonlinear patterns directly from historical data. Recent progresses in deep learning offer promising alternatives. Long Short-Term Memory (LSTM) networks address the vanishing gradient problem and capture long-term temporal dependencies through the gated architecture. Bidirectional LSTM (BiLSTM) extends this by processing sequences in both forward and backward directions, theoretically capturing richer contextual information. Hybrid models combining convolutional neural networks with LSTM (CNN-LSTM) extract local patterns through convolutional layers before modeling sequential dependencies.

Despite of these advances, several important research gaps still remain. First, although LSTM-based architectures have been widely applied to financial forecasting, existing studies predominantly evaluate individual models in isolation. Systematic comparisons of LSTM, BiLSTM, and CNN–LSTM under identical datasets, preprocessing procedures, and evaluation criteria are still limited, making it difficult to identify their relative strengths and weaknesses. Second, the forecasting performance of these architectures has rarely been examined in the context of the VIX, whose forward-looking nature, pronounced volatility clustering, heavy-tailed distribution, and abrupt regime shifts present substantially greater modeling challenges than conventional financial time series. Consequently, it remains unclear whether the advantages reported for equity prices or returns can be generalized to implied volatility forecasting. Third, while increasingly sophisticated deep learning architectures have been proposed, there is little empirical evidence on whether greater architectural complexity consistently leads to superior predictive performance, particularly during periods of heightened market uncertainty. This raises an important practical question for financial institutions, where forecasting accuracy must be balanced against computational efficiency, model robustness, and real-time applicability.

This study addresses these gaps by conducting a comprehensive comparative analysis of LSTM, BiLSTM, and CNN–LSTM models using daily VIX data from 2 January 2000 to 24 June 2024, which encompasses multiple episodes of severe financial stress, including the Global Financial Crisis and the COVID-19 pandemic. Model performance is evaluated by multiple complementary metrics, including MAE, MSE, and the coefficient of determination (R^2), while additional analysis examines each model's ability to capture volatility dynamics during periods of elevated market uncertainty.

The findings contribute to the existing literature in several respects. First, the study provides a consistent empirical comparison of three representative LSTM-based architectures under a unified experimental framework, offering a clearer understanding of their relative forecasting capabilities for implied volatility. Second, the results demonstrate that the hybrid CNN–LSTM architecture delivers the most accurate and robust forecasts, highlighting the benefits of combining local feature extraction with temporal dependency learning for modeling complex volatility dynamics. Third, the analysis shows that increased model complexity does not necessarily guarantee better predictive performance, as the BiLSTM model appears more susceptible to overfitting in highly non-stationary volatility environments. These findings provide useful guidance for selecting deep learning architectures in practical applications such as volatility forecasting, portfolio risk management, and real-time market surveillance.

The structure of this paper is organized as follows: Section 2 reviews the relevant literature on volatility forecasting and deep learning methods. Section 3 presents the methodological framework and model architectures. Section 4 introduces the dataset, experimental design, and empirical results, followed by a discussion of the comparative findings. Finally, Section 5 concludes the paper, outlines its practical implications, and proposes future research directions.

2. Literature review

Volatility forecasting has traditionally relied on the GARCH family of models [1, 2]. These models capture volatility clustering, the tendency for large price movements to be followed by large movements, through conditional variance equations. Extensions such as EGARCH and GJR-GARCH incorporate asymmetric responses to positive and negative shocks. However, GARCH

DOI: 10.54254/2755-2721/2026.GF35824

models assume a parametric form for the volatility process and normally distributed innovations, both of which fail empirically for financial returns characterized by heavy tails and nonlinear dynamics [3]. Machine learning methods partially address these limitations. Random forests and gradient boosting models capture nonlinear relationships without strong parametric assumptions [4]. Support vector machines have shown effectiveness in classifying volatility regimes [5]. However, these methods typically treat observations as independent and identically distributed, neglecting the temporal dependencies that define volatility clustering. This limitation has motivated the adoption of recurrent neural networks specifically designed for sequential data.

Recent advances in deep learning have substantially improved financial volatility forecasting by enabling models to learn complex nonlinear relationships directly from large-scale financial data without relying on restrictive parametric assumptions. In particular, deep learning models trained on high-frequency financial data have demonstrated superior predictive performance over conventional statistical approaches in volatile market environments [5, 6]. Recurrent neural networks (RNNs) maintain hidden states that propagate information across time steps, making them suitable for time-series forecasting. However, standard RNNs suffer from vanishing gradients, limiting their ability to capture long-range dependencies. LSTM networks overcome this limitation through a gated architecture consisting of memory cells, input gates, forget gates, and output gates [7]. This design allows LSTMs to retain information over extended periods while adaptively forgetting irrelevant past observations. Empirical studies demonstrate LSTM's advantages for volatility forecasting. Fischer and Krauss showed that LSTM improves directional accuracy compared to traditional models [8]. Pan found that LSTM provides earlier warning signals for financial crises than random forests or gradient boosting [9]. Kabir et al. documented LSTM's superior ability to model volatility persistence and nonlinear tail risks [10].

Bidirectional LSTM processes the sequence in both the forward and backward directions, and concatenates the hidden states obtained from the two passes. This design theoretically captures richer contextual information, which may benefit forecasting tasks where future observations inform present predictions [11]. However, bidirectional processing carries risks for volatility forecasting. Because VIX data during crises exhibit sharp, non-stationary spikes, backward information may inadvertently incorporate future noise, leading to overfitting rather than improved generalization. Hybrid CNN-LSTM models combine local feature extraction with sequential dependency modeling. Convolutional layers apply learnable filters across time windows to extract local patterns such as short-term volatility bursts. These extracted features are then passed to LSTM layers to model longer-term dependencies [12, 13].

This architecture offers two potential advantages: convolutional layers reduce noise through local averaging while retaining sharp features, and the hierarchical design enables the model to learn patterns at multiple temporal scales simultaneously.

Despite these theoretical advantages, systematic comparisons of LSTM, BiLSTM, and CNN-LSTM on VIX data remain scarce. The present study fills this gap by evaluating all three architectures under identical experimental conditions.

3. Methodology

3.1. LSTM network

Long Short-Term Memory (LSTM) unit contains a cell state C_t that acts as memory, and three gates: forget gate f_t, input gate i_t, and output gate o_t, which regulate information flow. These gates enable

DOI: 10.54254/2755-2721/2026.GF35824

the LSTM to selectively store, forget, or output information at each time step, making it particularly suitable for modeling financial time series [7, 14]. The structure of LSTM is given in Figure 1.

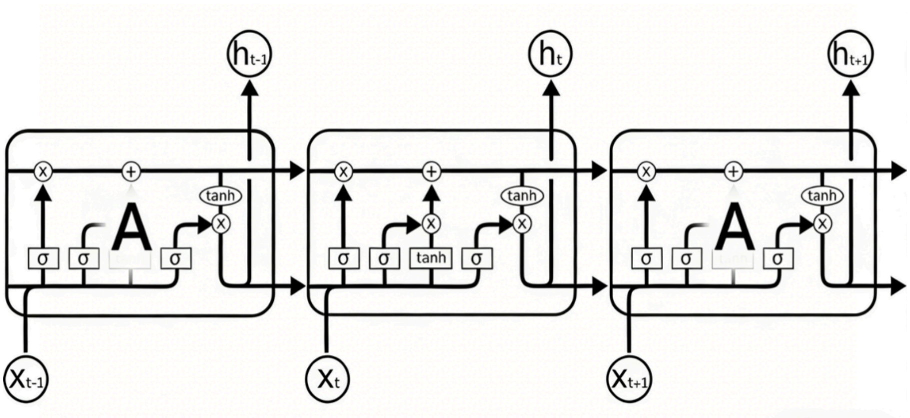

Figure 1. The structure diagram of LSTM. Source: olah [14]

The forget gate illustrates how much information from the previous cell state C_{t-1} should be retained or discarded. This allows the LSTM to adaptively forget irrelevant information and focus on salient patterns over time. The forget gate is computed as:

$$f_t = \sigma\left(W_f \bullet \ [h_{t-1},\ x_t] + \ b_f\right. \tag{1}$$

Where σ denotes the sigmoid function, which outputs values in the range [0,1], h_{t-1} is the hidden state from the previous time step, x_t is the current input vector, W_f and b_f are the weight matrix and bias vector for the forget gate, respectively.

The input gate controls how much new information from the current input x_t should be added to the cell state. It is computed as:

$$i_t = \sigma(W_i \bullet \ [h_{t-1},\ x_t] + \ b_i) \tag{2}$$

A candidate cell state $\widetilde{C_t}$ is generated to represent potential new information:

$$\widetilde{C_t} = tanh(W_c \bullet [h_{t-1}, x_t] + b_c) \tag{3}$$

Where W_i, W_c and b_i, b_c, are the corresponding weights and biases, and is tangent function with values in [-1,1] that ensures nonlinearity in the output.

The output of f_t and i_t is then element-wise multiplied with C_{t-1} and $\widetilde{C_t}$ to determine which information to retain:

$$C_t = f_t \odot C_{t-1} + i_t \odot \widetilde{C_t} \tag{4}$$

Where $\odot$ denotes element-wise multiplication.

The output gate determines which parts of C_t should be exposed as the hidden state h_t for the next time step or as the model output, calculated as:

$$o_t = \sigma(W_o \bullet [h_{t-1}, x_t] + b_o)$$

DOI: 10.54254/2755-2721/2026.GF35824

$$h_t = o_t \odot tanh(C_t) \tag{5}$$

3.2. Bi-LSTM network

Bidirectional LSTM (Bi-LSTM) network captures both past and future contextual information. Different from LSTM, Bi-LSTM consists of 2 layers, that are forward and reverse directions. For each input x_t, the forward LSTM computes hidden state $\vec{h_t} = LSTM(x_t, \vec{h_{t-1}})$ and the backward hidden states $\overleftarrow{h_t} = LSTM(x_t, \overleftarrow{h_{t-1}})$. The final feature vector h_t at time t is obtained by the forward and backward hidden states, given by:

$$h_t = LSTM\left[\overrightarrow{h_t},\ \overleftarrow{h_t}\right] \tag{6}$$

This concatenation allows the Bi-LSTM to extract long-range dependencies and context-aware features by considering both prior and future information simultaneously. The structure diagram of BiLSTM is shown in Figure 2.

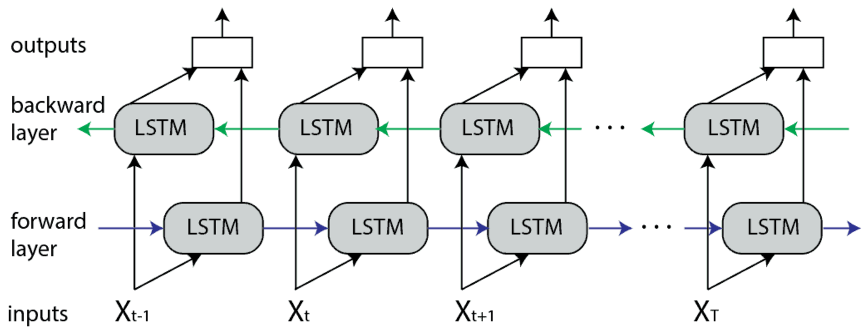

Figure 2. The structure diagram of BiLSTM

3.3. CNN-LSTM

The CNN-LSTM hybrid first applies convolutional layers for local pattern extraction. A 1D convolution with learnable filters k slides across the input sequence:

$$f(t) = (x * k)(t) = \sum_{a=-\infty}^{\infty} x(a) \bullet k(t-a) \tag{7}$$

This operation captures local patterns such as short-term volatility fluctuations. Max pooling reduces dimensionality and provides translation invariance.

After convolution and pooling, the feature maps are flattened and passed to LSTM layers that model sequential dependencies. The Rectified Linear Unit (ReLU) activation introduces nonlinearity:

$$f(x) = max(0,\ x) \tag{8}$$

3.4. Measurement

To evaluate the predictive performance of the proposed volatility forecasting models, this study employs a set of widely adopted statistical and machine-learning performance metrics. These

DOI: 10.54254/2755-2721/2026.GF35824

metrics quantify the discrepancy between observed and predicted values and assess the classification ability of models when volatility is categorized into different levels. The key evaluation measures include Mean Squared Error (MSE), Mean Absolute Error (MAE) and R-squared.

The Mean Squared Error (MSE) measures the average squared difference between the actual value and the predicted value. It is among the most commonly used error metrics in regression-based forecasting tasks because it penalizes large deviations more heavily than small ones, thus providing a sensitive measure of prediction accuracy. The MSE is mathematically defined as:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2 \tag{9}$$

where y_i denotes the observed value, $\hat{y}_i$ is the predicted value, and n is the total number of observations. Lower MSE indicates better model performance, with an ideal value of 0 representing a perfect prediction. When MSE approaches 0, it means the model's predictions are very close to the actual values. Conversely, a larger MSE indicates greater overall error in the predictions.

The Mean Absolute Error (MAE) measures the average absolute difference between actual and predicted values:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|y_i - \hat{y}_i\right|^2 \tag{10}$$

Compared with MSE, MAE is less sensitive to outliers and provides a more intuitive measure of average error magnitude. Lower MAE indicates better model performance, with an ideal value of 0 representing a perfect prediction.

The R-squared (R^2) measures the proportion of variance in the observed values that is explained by the model predictions and is defined as:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2}{\sum_{i=1}^{n}\left(y_i - \overline{Y}\right)^2} \tag{11}$$

where y_i denotes the actual value, $\hat{y}_i$ is the predicted value, $\overline{Y}$ represents the sample mean of the observed series, and n is the total number of observations. R^2 provides an intuitive measure of the model's explanatory power by quantifying how well the predicted values replicate the variability of the observed data. A higher R^2 indicates better model performance, with a value of 1 corresponding to perfect explanatory accuracy, while values closer to 0 suggest limited explanatory capability. Unlike MAE and MSE, which focus on prediction errors in absolute or squared terms, R^2 emphasizes the overall goodness-of-fit and is particularly useful for comparing the relative performance of competing models.

4. Empirical analysis

4.1. Data description

Daily VIX data from January 2, 2000, to June 24, 2024, were obtained from CBOE. The sample includes 8,685 observations spanning major crisis periods: the dot-com bubble (2000-2002), the Global Financial Crisis (2008-2009), the European debt crisis (2011), and the COVID-19 pandemic

DOI: 10.54254/2755-2721/2026.GF35824

(2020). Among open, high, low, and close prices, we use the closing price as the primary series because it best represents daily market settlement conditions.

Figure 3 shows the full time series. The VIX exhibits three notable spikes exceeding 40: the 2008 financial crisis (peak 89), the 2011 debt ceiling episode (peak 48), the 2020 pandemic (peak 85). Smaller spikes during 2015-2016 reflect the oil price crash and China slowdown concerns.

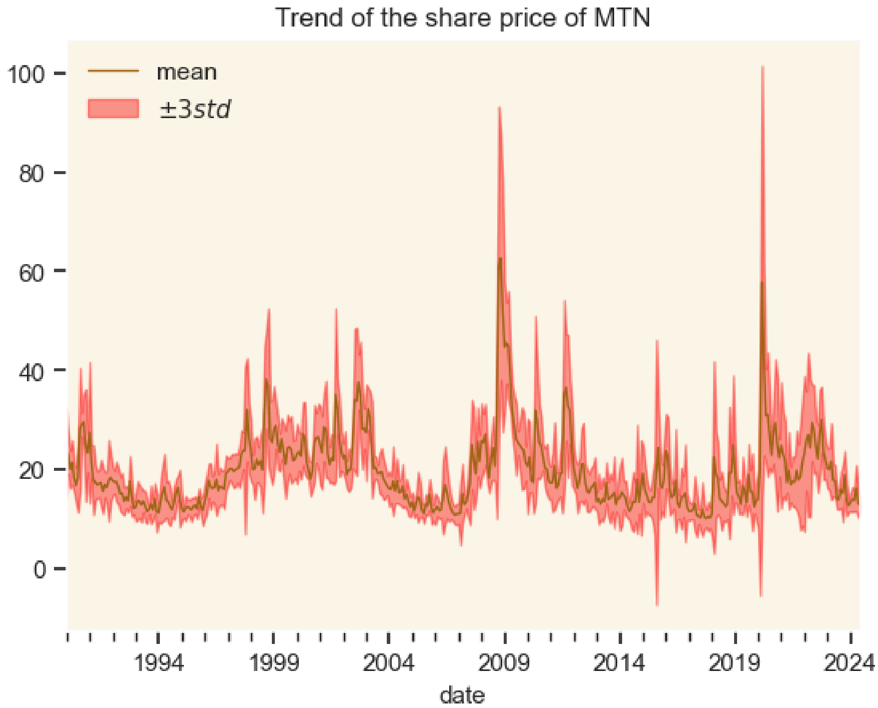

Figure 3. Daily VIX closing price, January 2000 - June 2024

The distribution in Figure 4 shows positive skewness (2.48) and excess kurtosis (9.47), confirming heavy-tailed behavior that violates normality assumptions. The Jarque-Bera test strongly rejects normality ($p < 0.001$). These characteristics justify using nonlinear models while also posing challenges for bidirectional architectures that may overfit to extreme observations.

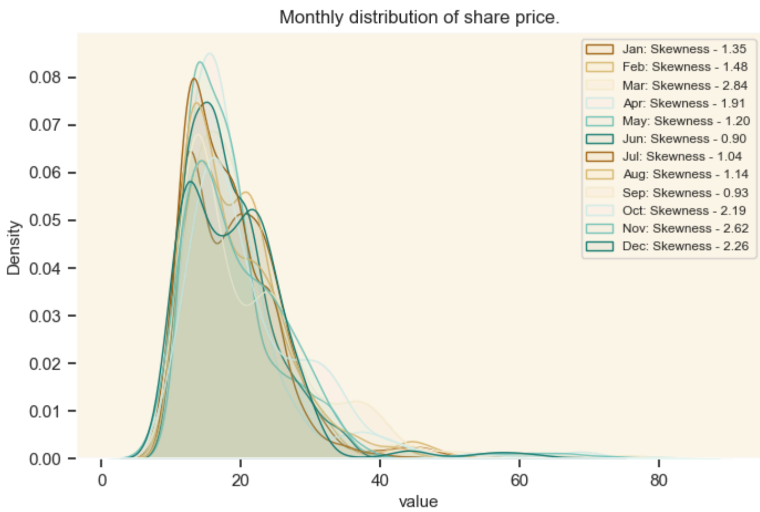

Figure 4. Histogram and kernel density of VIX closing prices

DOI: 10.54254/2755-2721/2026.GF35824

4.2. Experimental setup

Data were split chronologically: 80% training (6,948 observations) and 20% testing (1,737 observations). A sliding window of 62 days (approximately three trading months) was used to create input sequences. The Adam optimizer with learning rate 0.001 was used for all models. Dropout rate of 0.2 was applied to prevent overfitting. Early stopping with patience of 20 epochs monitored validation loss. Models were trained for up to 10 epochs with batch size 32.

4.3. Results

Table 1 shows the performance metrics for all three models. The CNN-LSTM model achieves the best performance across all metrics. Its MAE of 1.758 is 23.7% lower than LSTM and 40.1% lower than BiLSTM. Its R^2 of 0.872 indicates that the model explains 87.2% of the variance in VIX movements, compared to 78.0% for LSTM and 67.0% for BiLSTM. The poor performance of BiLSTM is notable. Despite its theoretical advantage of incorporating future context, BiLSTM performs worst, particularly on MSE which penalizes large errors. We attribute this to two factors. First, the highly non-stationary nature of VIX, especially during crisis periods, means that "future information" often contains large, unpredictable spikes that degrade rather than improve predictions. Second, bidirectional processing may overfit to noise, as the backward pass effectively uses future data to fit past observations, reducing generalization.

Table 1. Reports the performance metrics for all three models

	MAE	MSE	R^2
LSTM	2.302535	15.540974	0.779978
Bi-LSTM	2.934829	23.326354	0.669756
CNN-LSTM	1.758256	9.033082	0.872114

Figure 5 compares LSTM predictions against actual values. The model captures directional trends reasonably well but systematically underestimates peak magnitudes. During the 2008 and 2020 crises, LSTM predictions lag actual spikes by 2-3 days and underestimate heights by 30-40%.

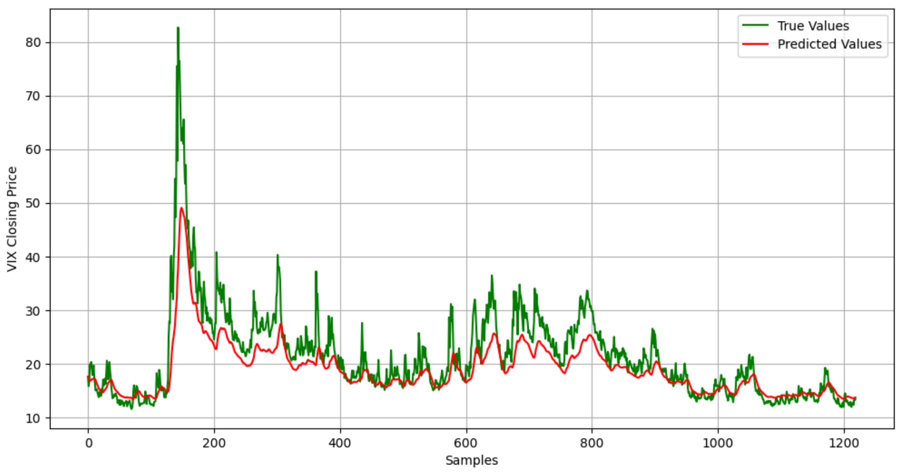

Figure 5. True values vs predicated values by LSTM

DOI: 10.54254/2755-2721/2026.GF35824

Figure 6 shows BiLSTM predictions. The model exhibits more volatile predictions with larger errors during stable periods, suggesting that bidirectional processing introduces noise rather than signal. The MAE of 2.935 confirms this visual impression.

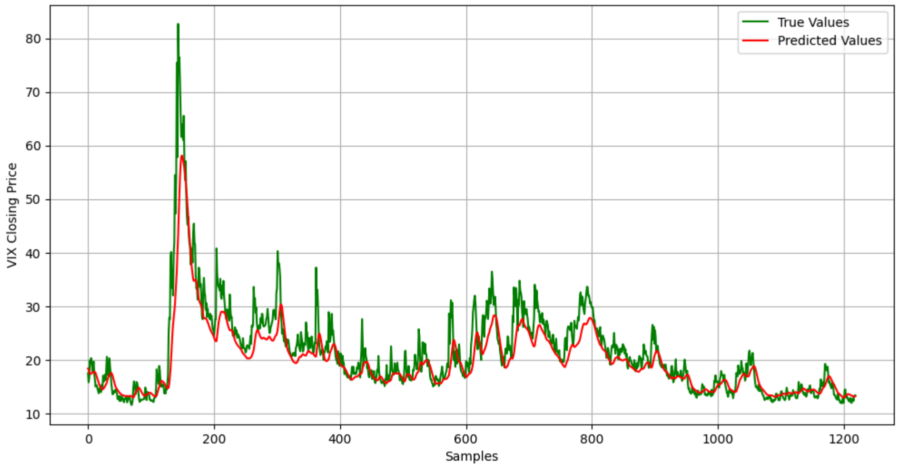

Figure 6. True values vs predicated values by Bi-LSTM

Figure 7 demonstrates CNN-LSTM predictions. The hybrid model tracks actual VIX more closely across both stable and volatile periods. Peak magnitudes are better captured, and phase lags are reduced. The convolutional layers' ability to extract local patterns, particularly the sharp rise-and-fall patterns characteristic of VIX spikes, combined with LSTM's sequential modeling appears to produce this improvement.

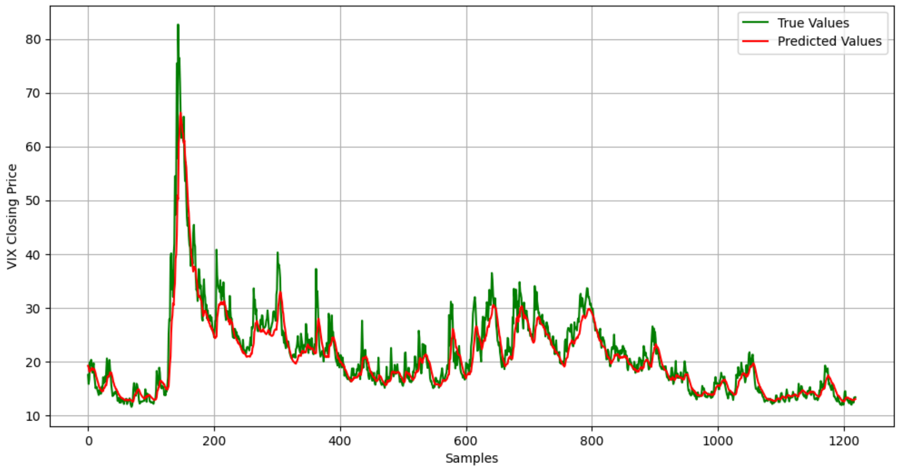

Figure 7. True values vs predicated values by CNN-LSTM

5. Conclusion

This study compared LSTM, BiLSTM, and CNN-LSTM architectures for VIX volatility forecasting using daily data from January 2000 to June 2024. The CNN-LSTM hybrid consistently outperformed both unidirectional and bidirectional LSTMs across all evaluation metrics, achieving R^2 of 0.872 compared to 0.780 for LSTM and 0.670 for BiLSTM. Three findings warrant emphasis. First, the superior performance of CNN-LSTM suggests that local feature extraction through convolutional layers provides substantial benefits for volatility forecasting, likely because VIX spikes follow characteristic short-term patterns that convolutions are designed to detect. Second, BiLSTM performed worst despite its architectural complexity, indicating that bidirectional processing may be counterproductive for highly non-stationary financial time series where "future" information during crisis periods is noise rather than signal. Third, all deep learning models maintained reasonable predictive stability during extreme events, though CNN-LSTM showed the smallest performance degradation during the 2008 and 2020 crisis periods. These findings have practical implications for risk management. The CNN-LSTM modelachieves sufficient accuracy for real-time volatility monitoring, with R^2 of 0.872 suggesting that most predictable variation is captured. The poor performance of BiLSTM suggests that practitioners should prioritize architectural choices based on empirical validation rather than theoretical complexity.

Future research should explore three directions. First, attention mechanisms could be integrated with CNN-LSTM to identify which historical patterns most influence current predictions. Second, transfer learning could be examined to determine whether models trained on VIX generalize to other volatility indices such as VXN (NASDAQ volatility) or VXEEM (emerging markets volatility). Third, the inclusion of macroeconomic variables, particularly interest rates and credit spreads, may improve predictions during crisis periods when volatility exhibits stronger linkages to broader financial conditions.

References

[1] Engle, R. F. (1982). Autoregressive conditional heteroscedasticity with estimates of the variance of United Kingdom inflation. *Econometrica, 50*(4), 987–1007.
[2] Bollerslev, T. (1986). Generalized autoregressive conditional heteroskedasticity. *Journal of Econometrics, 31*(3), 307–327.
[3] Zhang, C., et al. (2024). Volatility forecasting with machine learning and intraday features. *Journal of Financial Econometrics, 22*(2), 492–515.
[4] Feng, Y., Zhang, C., & Li, Z. (2023). Machine learning approaches for volatility forecasting and risk management: Evidence from global equity markets. *Expert Systems with Applications, 223*, 119992.
[5] Nguyen, H. T. (2024). Deep learning enhanced volatility modeling with covariates: The RECH-X framework. *Journal of Forecasting, 43*(5), 984–1003.
[6] Moreno-Pino, F., et al. (2024). DeepVol: Volatility forecasting from high-frequency data using deep learning. *Scientific Reports, 14*, 2291.
[7] Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation, 9*(8), 1735–1780.
[8] Fischer, T., & Krauss, C. (2018). Deep learning with long short-term memory networks for financial market predictions. *European Journal of Operational Research, 270*(2), 654–669.
[9] Pan, J. (2024). Predicting financial crises using LSTM-based network spillover forecasting. *Asia-Pacific Financial Studies, 51*(2), 67–84.
[10] Kabir, M. R., Bhadra, D., Ridoy, M., & Milanova, M. (2025). LSTM–Transformer-based robust hybrid deep learning model for financial time series forecasting. *Sci, 7*(1), 7. https://doi.org/10.3390/sci7010007
[11] Cheng, Y., Xu, Z., Chen, Y., Wang, Y., Lin, Z., & Liu, J. (2025). A deep learning framework integrating CNN and BiLSTM for financial systemic risk analysis and prediction. *arXiv preprint*. arXiv:2502.06847.
[12] Gürüler, H. (2023). A hybrid CNN-LSTM model for stock market volatility prediction. *Forecasting, 5*(2), 301–318.
[13] Wei, J., Yang, S., & Cui, Z. (2025). Integrated GARCH-GRU in financial volatility forecasting. *arXiv preprint*. arXiv:2504.09380.
[14] Olah, C. (2015). Understanding LSTM networks. *Colah's Blog*. https://colah.github.io/posts/2015-08-Understanding-LSTMs/