Received: 18 March 2025 | Revised: 17 June 2025 | Accepted: 11 July 2025 | Published online: 29 August 2025

RESEARCH ARTICLE

Journal of Computational and Cognitive Engineering
2025, Vol. 00(00) 1–15
DOI: 10.47852/bonviewJCCE52025731

Multi-sequential Neural Network Models for Stock Price Forecasting

Sheba Elizabeth Thomas[1], Rubell Marion Lincy George[1], Nevin Selby[1], Aditya Taparia[1] and Jobish Vallikavungal[2,*]

[1] *Indian Institute of Information Technology Kottayam, India*

[2] *School of Engineering and Sciences, Tecnologico de Monterrey, Mexico*

Abstract: Deep learning techniques are transforming stock market forecasting by significantly improving the accuracy of predicting price movements and market patterns. In this research work, we propose two novel hybrid architectures – MSLSTM (Multivariate Sequential Long Short-Term Memory) and MSLSTMA (Multivariate Sequential Long Short-Term Memory Autoencoder). Both models leverage the Long Short-Term Memory (LSTM) ability to capture complex temporal dependencies in sequential financial data. Our method performs well in a variety of industries and surpasses a few other LSTMs and their autoencoder-augmented variations, as well as conventional deep learning models like CNNLSTM. MSLSTMA achieves the highest sector-wise winning rates, notably 95.83% in the telecommunication sector, 87.5% in technology, and 83.33% in industrials. Model performance was rigorously evaluated using mean squared error, root mean squared error, mean absolute error, and mean absolute percentage error. Across all these metrics, MSLSTMA consistently delivered the lowest error rates, showcasing its superior accuracy and practical relevance for real-world financial forecasting. MSLSTM also demonstrated strong performance, with winning rates of 50% in consumer staples and 45.83% in the healthcare sector. This research introduces an effective and scalable tool for investors, with the potential to enhance investment decisions through precise stock price prediction.

Keywords: stock price forecasting, long short-term memory, financial market, investment decision-making, deep learning, time series

1. Introduction

Investment, a strategic financial practice, involves the allocation of capital to various assets to generate income or capital gains. This financial strategy not only allows individuals to combat the erosive impact of inflation but also serves as a mechanism for wealth accumulation over time, thereby contributing to financial security for both the investor and their family. Traditional models such as GARCH and ARIMA have difficulty capturing the complex nonlinear correlations that are present in stock prices. It is difficult for Recurrent Neural Networks (RNNs) to identify long-term dependencies in sequential data because of the vanishing gradient problem. The Long Short-Term Memory (LSTM) was introduced to overcome this limitation by allowing the network to selectively remember or forget information over extended sequences.

The proposed methodology utilizes the application of Multivariate Sequential Long Short-Term Memory (MSLSTM) and Multivariate Sequential Long Short-Term Memory Autoencoder (MSLSTMA), a novel approach in the realm of stock price prediction. By leveraging the inherent sequential nature of stock price data, MSLSTM networks aim to capture long-term dependencies and intricate relationships among multiple variables simultaneously. The sequential structure of MSLSTM makes it well-suited for forecasting tasks that involve temporal dependencies, aligning with the dynamic nature of market fluctuations. These models stand out for their capacity to simultaneously consider multiple variables, departing from the conventional practice of treating each variable in isolation, as often observed in traditional forecasting methods. This research endeavor aims to bridge the existing gap and lay the groundwork for future investigation, further exploration, and refinement of MSLSTM and MSLSTMA-based approaches in the domain of financial forecasting.

Further, we are conducting a comprehensive analysis of the models and algorithms employed in the field of deep learning for various tasks, including time series prediction and classification. Univariate Sequential LSTM (USLSTM), Univariate Sequential Long Short-Term Memory Autoencoder (USLSTMA), Gated Recurrent Unit (GRU), Generative Adversarial Networks (GANs), and Convolutional Neural Network Long Short-Term Memory (CNNLSTM) are the prominent ones in the field. The USLSTM is a type of RNN intended to identify persistent dependencies in sequential information. This model is specifically applied to univariate (single-variable) time series data, focusing on predicting future values based on the historical values of a single variable.

The USLSTMA model is a modified version of USLSTM that incorporates an autoencoder to improve the results. It consists of an encoder and a decoder, and its goal is to figure out how to represent the input data. This model is intended to represent the key elements and trends in a time series that consists of a single

***Corresponding author:** Jobish Vallikavungal, School of Engineering and Sciences, Tecnologico de Monterrey, Mexico. Email: jobish03@tec.mx

variable. GRU is a kind of RNN that is similar to LSTM and is intended to identify persistent dependencies in data that occur sequentially. This research contributes to the evolving landscape of stock price prediction by introducing and validating the effectiveness of two variations of LSTM networks.

1) Multi-sequential Long Short-Term Memory (MSLSTM) – The purpose of this model is to capture the temporal dynamics. The model can comprehend the complex interactions between the indicators better with additional LSTM layers. In this case, the network may learn temporal properties at different abstraction levels.
2) Multi-sequential Long Short-Term Memory Autoencoder (MSLSTMA) – In this framework, an autoencoder is included to incorporate the latent variables that affect the distribution of the data. This reduces noise and complexity by extracting only the most important features from the incoming data. The LSTM can focus on the most important information by receiving this compressed and denoised data, which could result in more accurate predictions.

Our experimental results demonstrate this technique's superiority. To ensure the accuracy and consistency of the financial data, the Refinitiv Eikon dataset is used. Together, these approaches produce market share forecasts that are more reliable, strong, and broadly applicable. These forecasts can be useful resources for investors and financial professionals attempting to comprehend the complexities of the financial system.

The rest of this research is structured as follows. In Section 2, we present the relevant literature, and then in Section 3, we present some predictive artificial intelligence (AI) models for stock price prediction and their architectures. It covers established machine learning techniques such as random forests (RFs) and advanced deep learning models including LSTM networks. In Section 4, we propose new predictive models that build upon existing methodologies. The experimental setups and data collection methods used to evaluate the proposed models are detailed in Section 5. Then, in Section 6, we present and analyze the results of our experiments. The analysis includes a comparison of prediction accuracies, computational efficiency, and robustness of the proposed models. Lastly, Section 7 summarizes the key findings of the research and discusses potential future directions. It highlights the contributions of the proposed models to the field of stock price prediction and suggests areas for further investigation. The discussion includes reflections on limitations and possible improvements to enhance model accuracy and applicability.

2. Literature Review on Predictive AI Models in Stock Market Price Prediction

Recent research has explored a range of AI and machine learning techniques for stock market prediction. Among these, LSTM networks have gained prominence for their superior accuracy in forecasting stock prices [1, 2]. Additionally, hybrid models that integrate multiple AI techniques are emerging, capitalizing on the strengths of each approach to enhance predictive performance. While a comprehensive analysis of all models is beyond the scope of this study, we will focus on the most significant AI models and their hybrid forms documented in the literature. This includes a detailed examination of recent advancements in deep learning techniques for price prediction, as outlined in Reference [3], which categorizes various aspects such as neural network architectures, databases, evaluation metrics, and implementation strategies. The study offers a thorough classification of standard models and their variants, hybrid models combining deep learning with traditional approaches, and those integrating different deep learning techniques. Shahi et al. [4] compared the performance of deep learning models based on LSTM and GRU models. To make predictions, the authors used multivariate inputs in both bidirectional and unidirectional stacked structures. Asgarian et al. [5] developed GAN-based models to predict stock prices by incorporating optimized price features and sentiment data from social media. The S&P 500 index's next-day closing price was predicted using single-layer and multi-layer LSTM models in Reference [6].

Wang et al. [7] created a complex model that improved the accuracy of stock price prediction across four significant Asian markets by combining secondary decomposition (SD), multi-factor analysis, and attention-based LSTM (ALSTM) networks. Similarly, Polamuri et al. [8] introduced the Stock-GAN framework, which integrates GANs with LSTM networks and Convolutional Neural Networks (CNNs) as the generator and discriminator, respectively. This framework is further refined by combining it with the Multi-Model-based Hybrid Prediction Algorithm and is optimized through reinforcement learning and Bayesian methods.

A notable contribution in Reference [9] involves a hybrid framework that integrates empirical wavelet transform for preprocessing, LSTM networks with dropout and particle swarm optimization for prediction, and outlier robust extreme learning machine (ORELM) for post-processing, demonstrating significant improvements over traditional forecasting methods. Additionally, Lin et al. [10] combined LSTM networks and Complete Ensemble Empirical Mode Decomposition with Adaptive Noise (CEEMDAN) to forecast realized volatility of financial indices, outperforming other models such as back propagation (BP) neural networks and support vector regression (SVR). Vidal and Kristjanpoller [11] developed a hybrid model that merges CNNs with LSTMs to enhance the prediction of gold volatility, showcasing superior performance compared to traditional GARCH and LSTM models. Forecasting accuracy for major stock indices was greatly improved in Reference [12] by a model that combined Adaptive Wavelet Transform with LSTM and several econometric models like ARIMAX-GARCH.

Other hybrid models also demonstrate promising advancements. Chong et al. [13] carried out a systematic evaluation of deep learning networks, including LSTM for the prediction of the stock market, highlighting the efficacy of LSTMs. To improve stock market predictions, Zhang et al. [14] created the CEEMD-PCA-LSTM model, which combines principal component analysis (PCA) and LSTM networks with Complementary Ensemble Empirical Mode Decomposition (CEEMD). Gao et al. [15] optimized a model that incorporated technical indicators and investor sentiment, comparing the performance of LSTM and GRU models and finding LASSO dimension reduction superior to PCA. Kim and Won [16] proposed a hybrid approach combining LSTM networks with various GARCH-type models, demonstrating improved forecasting accuracy. Furthermore, Bai et al. [17] enhanced stock price prediction by integrating neighborhood rough set theory and multivariate empirical mode decomposition (EMD) with LSTM networks, while Chen and Ge [18] improved prediction by incorporating an attention mechanism into LSTMs. These studies highlight the ongoing innovation and integration of LSTM networks in hybrid forecasting models, reflecting their potential for improving stock market predictions.

Many researchers have prominently featured the integration of LSTM networks and other RNNs. He and Kita [19] introduced a hybrid sequential GAN model that incorporates various types of RNNs, including LSTM and GRU, within both the generator and discriminator components of the GAN. This framework was applied to predict stock prices based on S&P 500 data and evaluated using diverse performance metrics. Similarly, Ding and Qin [20] developed a Deep Recurrent Neural Network (DRNN) model applying LSTM to simultaneously predict multiple stock prices, such as opening, lowest, and highest prices, and achieved an accuracy of over 95%, surpassing traditional LSTM and other DRNN models. Building on this, He and Kita [21] proposed a novel prediction model that integrates GANs with various types of RNNs, including LSTM and GRU. In their approach, LSTM was utilized as the generator within the GAN framework, and its performance was compared against different discriminators such as Multilayer Perceptron (MLP), RNN, LSTM, and GRU.

Many of the recent advancements in stock price prediction have prominently featured the integration of LSTM networks and CNNs, often within hybrid or adversarial frameworks. In Reference [22], a comparison was conducted between CNN and LSTM networks for stock price prediction. Building on this, Kumar et al. [23] developed a system incorporating GANs, where LSTM networks served as the generative model and CNNs as the discriminative model, aiming to enhance forecasting accuracy and reduce error. Similarly, Zhou et al. [24] proposed the GAN-FD framework, which combines LSTM and CNN within a GAN structure to forecast high-frequency stock market data, leveraging adversarial training to simulate trader behavior and improve prediction accuracy.

Expanding on these approaches, Wu et al. [25] introduced stock sequence array convolutional LSTM (SACLSTM), a novel framework that integrates CNNs and LSTMs to boost prediction accuracy. In this model, CNNs extract features, which are then processed by LSTMs for final predictions. Vidal and Kristjanpoller [11] also developed a hybrid system blending CNNs with LSTMs to improve gold volatility predictions, demonstrating superior performance over traditional models. Additionally, Jin et al. [26] created the empirical modal decomposition (EMD) and LSTM models, which integrates CNN-based sentiment analysis, EMD, and an enhanced LSTM featuring an attention mechanism to refine the accuracy of predictions and minimize delays.

Recent developments in stock market forecasting have prominently featured the integration of LSTM networks with various other models to enhance prediction accuracy. In Reference [27], researchers introduced NuNet, a deep learning framework that combines ConvLSTM for spatiotemporal feature extraction with LSTM for capturing temporal patterns. This approach is further enhanced by advanced sampling and data augmentation techniques, including trend sampling and column-wise random shuffling, to improve forecast accuracy. Similarly, Lin et al. [28] developed a forecasting model that integrates LSTM with CEEMDAN. They first decompose stock index data using CEEMDAN and then apply LSTM to forecast the decomposed components, reconstructing the final predictions from these components. Their CEEMDAN-LSTM model demonstrated superior performance in both emerging and developed equity markets.

Furthermore, Md et al. [29] proposed a Multi-Layer Sequential Long Short-Term Memory (MLS LSTM) model that was optimized using the Adam optimizer to address the vanishing gradient problem often encountered with simpler RNNs. Apart from that, Zhang et al. [30] developed a feature-enhanced LSTM network combined with residual-driven ν support vector regression (FEL-νSVR). This model integrates convolutional layers for feature extraction, LSTM for initial predictions, and νSVR to refine these predictions by modeling residuals, highlighting the potential of combining LSTM with other sophisticated methods to improve prediction capabilities.

Recent advancements in the forecasting of stock prices have showcased the effectiveness of various sophisticated models, each contributing unique strengths to enhance forecasting accuracy. Similarly, Ghosh et al. [31] examined the effectiveness of RF and CuDNNLSTM networks, demonstrating that incorporating multiple features surpasses single-feature models in forecasting price movements. In a different approach, Ito et al. [32] utilized LSTM networks to predict returns in foreign exchange markets by incorporating limited order book events. Their results showed that LSTM models applying limit order book events outperformed traditional benchmarks and transaction-only models in prediction accuracy, though the economic gains were not as significant when transaction costs were factored in. Additionally, Mirza et al. [33] enhanced online learning algorithms by integrating covariance information into LSTM and GRU networks, introducing covariance-based LSTM variants and weight matrix factorization to reduce computational complexity while improving performance.

Moreover, Patel et al. [34] developed a two-phase fusion approach for predicting stock market index values, using SVR to forecast technical indicators, which then served as inputs for Artificial Neural Networks (ANN), RF, and SVR in the second stage. This method outperformed single-stage models. In Reference [35], the researchers evaluated various time-varying volatility models, finding that models incorporating realized volatility, such as Realized EGARCH and Realized SV, generally provided better forecasts than traditional models like Exponential GARCH. Additionally, Wang and Chen [36] developed Factor-GAN, a framework that employs GANs for factor investing, integrating deep learning with a multi-factor pricing model. Factor-GAN demonstrated superior performance in stock return predictions and investment outcomes compared to traditional linear models. Lastly, Wu et al. [37] introduced the quantile autoregressive (QAR) model to forecast stock return volatility, showing that QAR outperforms traditional GARCH-type models, especially during financial turmoil, and offers valuable insights into the leverage effect on asset returns.

The recent reviews of stock market price prediction techniques have underscored the pivotal role of LSTM networks in advancing forecasting accuracy. In Reference [38], a review from 2015 onward highlighted the increasing use of LSTM among various deep learning methods, such as CNN, RNN, and advanced models like HAN and Wavenet. This review emphasized the benefits of deep learning and identified a gap in combining multiple deep learning approaches. Similarly, Hu et al. [39] systematically analyzed forecasting techniques from 2014 to 2018, focusing on how LSTM integrates with both traditional and nontraditional data for improved predictions. A comprehensive comparison of model types

Table 1
Comparison of model types and performance metrics from recent research studies

Article	Models	Performance measure
[5]	GAN, LSTM, CNN	Accuracy, F-Score
[17]	LSTM	MAE, MAPE, RMSE, R^2
[6]	SLSTM, MLSTM	RMSE, MAPE, R

(*Continued*)

Table 1
(*Continued*)

Article	Models	Performance measure
[18]	LSTM, AttLSTM	Wilcoxon signed-rank test, McNemar's test
[13]	DNN, PCA, AE, RBM	NMSE, RMSE, MAE, MI
[40]	HHT, XGBoost	Cumulative Rate of Return, Maximum Drawdown, SR
[20]	ADRNN, LSTM, DRNN,	MSE, MAE
[15]	LSTM, GRU	MSE, RMSE, MAE
[31]	CuDNNLSTM, RF	Daily Return, Sharpe Ratio, SD, Maximum Drawdown, VaR, CVaR
[19]	MLP, RNN, LSTM, GRU	RMSE, MAE
[21]	GAN, LSTM, MLP, RNN, GRU	RMSE, MAE
[32]	LSTM	Investment returns
[26]	CNN, LSTM	MAE, RMSE, MAPE, R^2
[41]	SVM	Diebold-Mariano and Giacomini-White tests
[22]	CNN, LSTM	Training and Testing Loss (Error), RMSE, MAPE
[16]	LSTM, DFN, GARCH	MAE, MSE, HMAE, HMSE
[23]	LSTM, CNN	RMSE, Forecast Error Loss
[27]	ConvLSTM, LSTM	MSE, MAE, MAPE
[28]	LSTM, SVM, BP	MSE, MAE, MAPE, RMSE, MCS
[10]	LSTM, BP, Elman, SVR, AR, HAR	MSE, MAE, HMSE, HMAE, MCS
[9]	LSTM, ORELM	MAE, MAPE, RMSE, SDE
[42]	CNN, LSTM, MLP, RNN, CNN-RNN, CNNLSTM	MAE, RMSE, R^2
[29]	MLS LSTM	MAPE, RMSE, R^2
[33]	LSTM, GRU	MSE, cross-entropy loss
[43]	ARIMA, SVM,	MSE
[34]	ANN, RF, SVR, SVR–ANN, SVR–RF, SVR–SVR.	MAPE, MAE, rRMSE, MSE
[8]	STM, CNN, GAN	MAE, MSE
[35]	EGARCH, SV, HAR, REGARCH, RSV	MSE, QLIKE
[11]	CNN, LSTM	MSE, MCS
[7]	VMD, ICEEMDAN, ALSTM	MAE, RMSE, MAPE, ROC, AUC
[36]	LSTM, GAN	RMSE, R^2
[25]	CNN, LSTM, SACLSTM	Accuracy
[37]	SSACNN	Accuracy
[30]	LSTM, SVR	R^2, RMSE, MAE, MAPE
[14]	EEMD, PCA, LSTM.	RMSE, MAE, NMSE, DS
[38]	QAR, GARCH, MS-GARCH	MSE, DM
[24]	LSTM, CNN	RMSRE, DPA
[12]	LSTM, XGBoost	RMSE, MAE, SMAPE, R^2

and performance metrics from recent research studies is presented in Table 1.

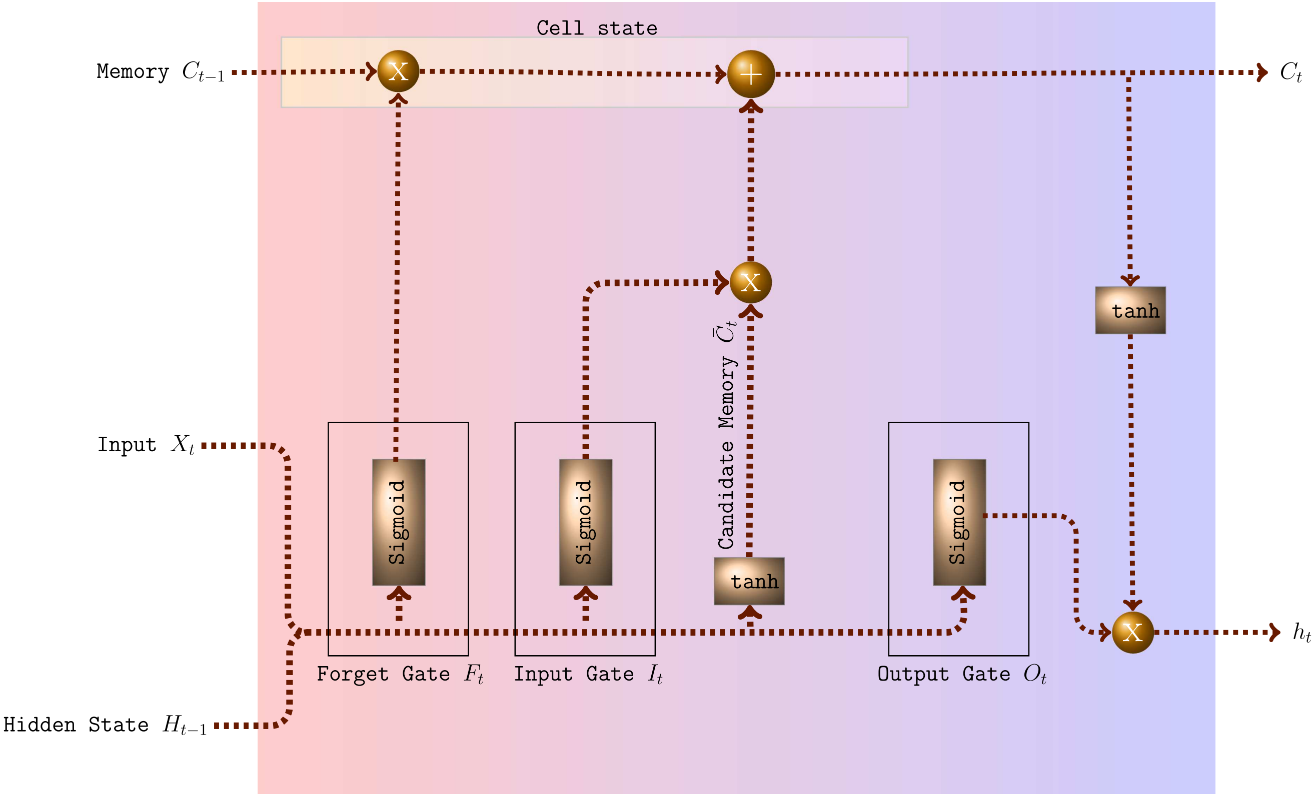
Additionally, Gandhmal and Kumar [44] reviewed various methodologies, including LSTM within a broader context of Bayesian models, Fuzzy classifiers, and ANN. Complementing these, Kehinde et al. [45] reviewed 220 articles from 2001 to 2021, revealing a trend toward advanced models and hybrid approaches, with LSTM networks increasingly being explored for their effectiveness in stock market forecasting. Mehtarizadeh et al. [46] used a hybrid modeling technique, initially using an ARIMA-GARCH framework to capture the statistical properties of the time series data before applying an LSTM network to improve predictive performance. Wang et al. [47] employs Large Language Model's (LLM) natural token prediction ability by treating stock prices as sequential tokens to enhance stock price prediction. The application of Gaussian process regression for price prediction in References [48–50] demonstrates the potential of advanced machine learning algorithms to capture complicated pricing behavior across domains.

The impressive performance metrics reported across various studies affirm the potential of these sophisticated techniques to advance the accuracy and effectiveness of stock market forecasting. To the best of our knowledge, none of the researchers explored LSTM in univariate and multivariate, sequential layers as we explored in this research.

3. AI Models for Stock Price Prediction

AI-based stock prediction models utilize techniques in machine learning and deep learning to study historical stock market

Figure 1
Long short-term memory architecture



information, identify patterns, and predict future stock prices. Deep learning algorithms, including LSTM networks, are particularly effective for stock prediction. LSTM networks, a type of RNN, excel at capturing long-term dependencies and patterns in stock prices. These models, along with others like Feedforward Neural Networks, autoencoders, and CNN, are adapted to handle the intricate patterns in stock market data, with hybrid architectures often enhancing their predictive accuracy. The Long Short-Term Memory architecture is presented in Figure 1.

The autoencoders are unsupervised learning models that aim to learn efficient representations of input data by encoding and decoding it and are used for feature extraction and dimensionality reduction in stock market data. Often associated with image processing, CNNs can be adapted to analyze sequential data and capture patterns using convolutional layers and are useful for extracting hierarchical features from stock market time series data. Deep learning algorithms, like the USLSTM model that blends LSTM and dense layers, are particularly good at learning from complicated, high-dimensional data in predictive modeling. This is further improved by the USLSTMA, which uses an autoencoder architecture with LSTM layers in both the encoding and decoding phases to enable precise sequence reconstructions.

3.1. The convolutional neural network model for stock price prediction (CNNLSTM)

The neural network architecture illustrated in Figure 2 integrates convolutional and LSTM layers to effectively model sequential data. The combination of convolutional and LSTM layers enables the model to make advantage of both spatial and temporal information for improved predictive performance.

The model is compiled using the Adam optimizer (with a learning rate of 0.00001) and mean squared error (MSE) as the loss function. The architecture is illustrated in Figure 2.

3.2. The Univariate Sequential Long Short-Term Memory model

The neural network architecture shown in Figure 3 illustrates USLSTM, a hybrid model combining LSTM layers with a dense layer, for time series prediction.

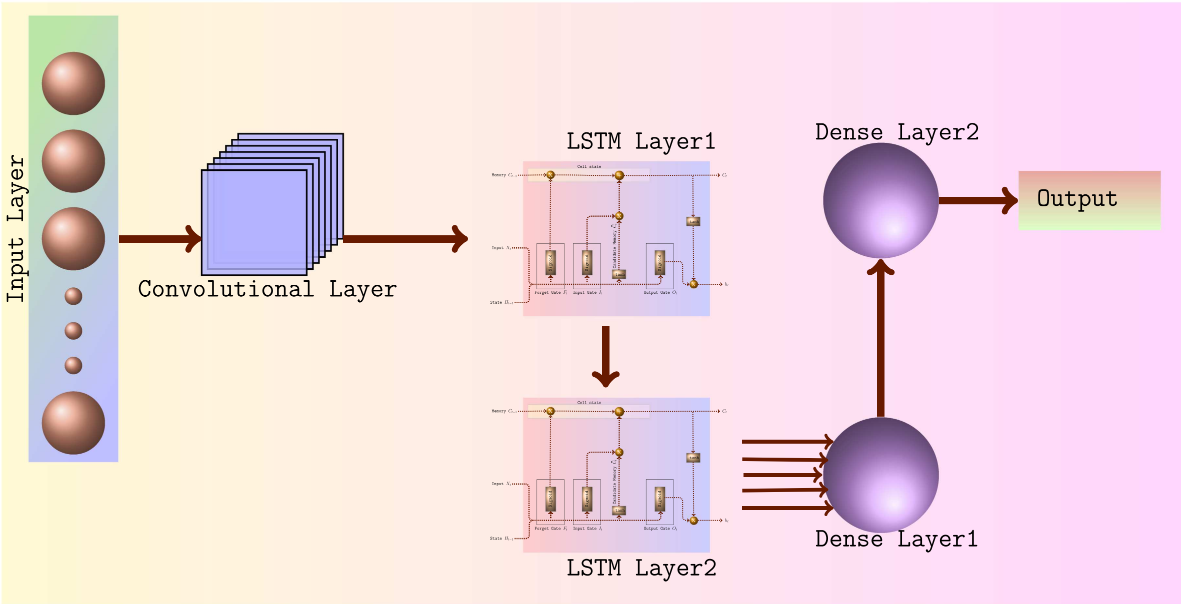
This architecture consists of two sequential LSTM layers followed by a dense output layer. The training procedure involves compiling the model with the Adam optimizer and MSE loss function. Early stopping and model checkpointing are implemented to avoid overfitting and save the best performing model. The model is trained on historical stock price data and validated on a separate validation set. Post training, the best model weights are loaded, and predictions are made on the test set.

3.3. The Univariate Sequential Long Short-Term Memory Autoencoder model

Figure 4 illustrates an autoencoder architecture, USLSTMA, designed for sequence data, employing LSTM layers in both the encoding and decoding phases. The architecture begins with an input layer that receives the sequence data, which is then processed by the first LSTM layer acting as an encoder. The encoder captures the temporal dependencies in the input sequence and produces a fixed size latent space representation.

The decoder, constituted by the second LSTM layer, reconstructs the sequence from the latent representation. The output of the

Figure 2
Architecture of CNNLSTM model



decoder is passed through a dense layer, which maps the decoder's output to the final reconstructed sequence.

The USLSTM model comprises two LSTM layers, while the USLSTMA model employs a single LSTM layer. Both models conclude with a dense layer with one unit, suitable for regression tasks such as stock price prediction.

4. Proposed Predictive AI Models for Stock Price Prediction

This section presents various LSTM-based models designed for stock price prediction, highlighting their architectures and functionalities. The MSLSTM model, featuring multiple LSTM layers followed by dense layers, and the MSLSTMA model, a sequence-to-sequence autoencoder, are introduced for robust sequence modeling, showcasing their capabilities in handling temporal dependencies and complex patterns in time series data. Even though LSTM-based architectures are widely used in stock price prediction, they frequently ignore the wealth of multivariate information available or rely on shallow architectures. Also, limited attention has been given to sector-wise comparisons of various deep learning architectures. This research work fills these gaps by employing MSLSTM and its autoencoder variation MSLSTMA to capture intricate temporal dynamics while also providing insights into the architecture that performs best in each sector.

Figure 3
Architecture of USLSTM model

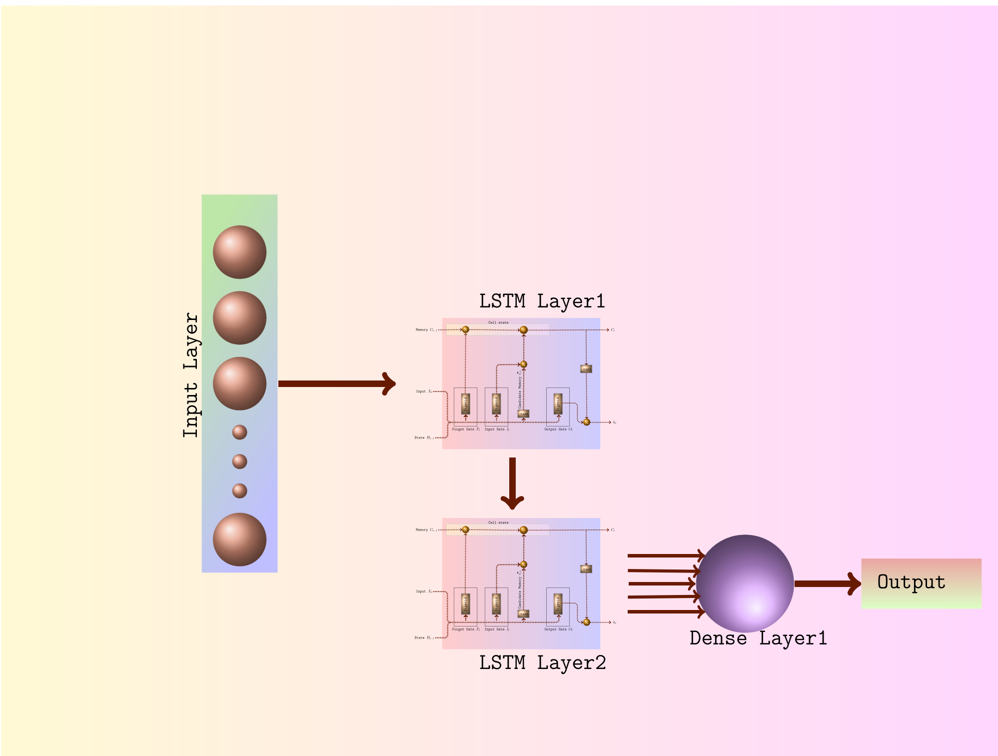


Figure 4
Architecture of USLSTMA model

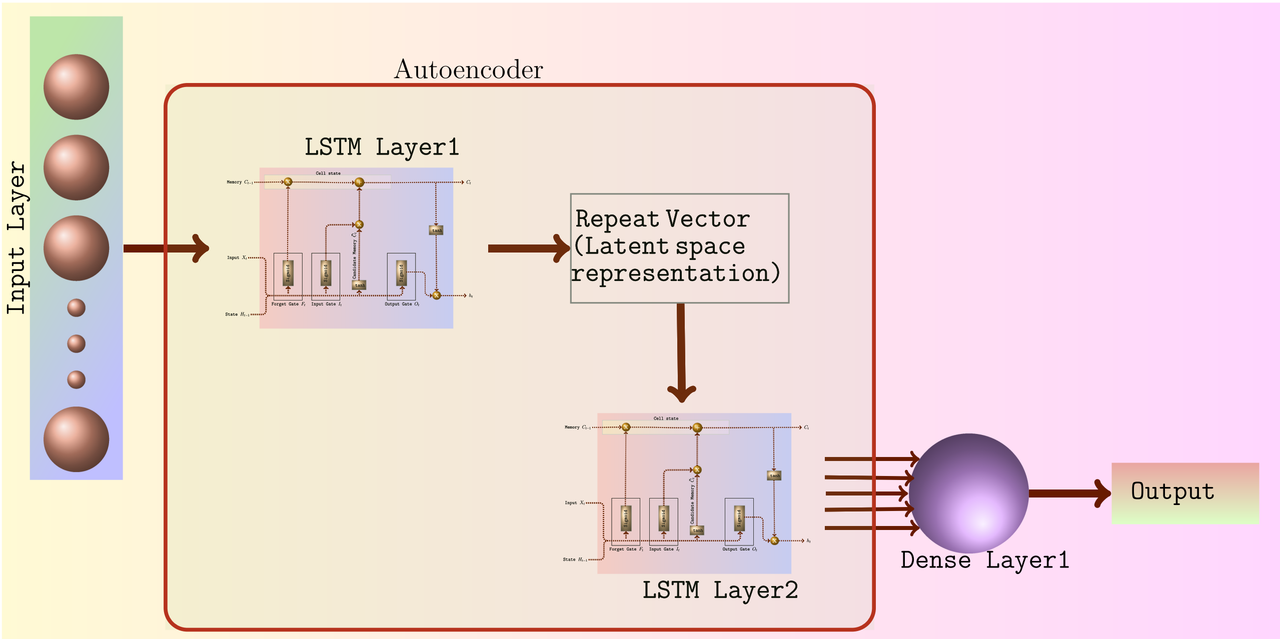
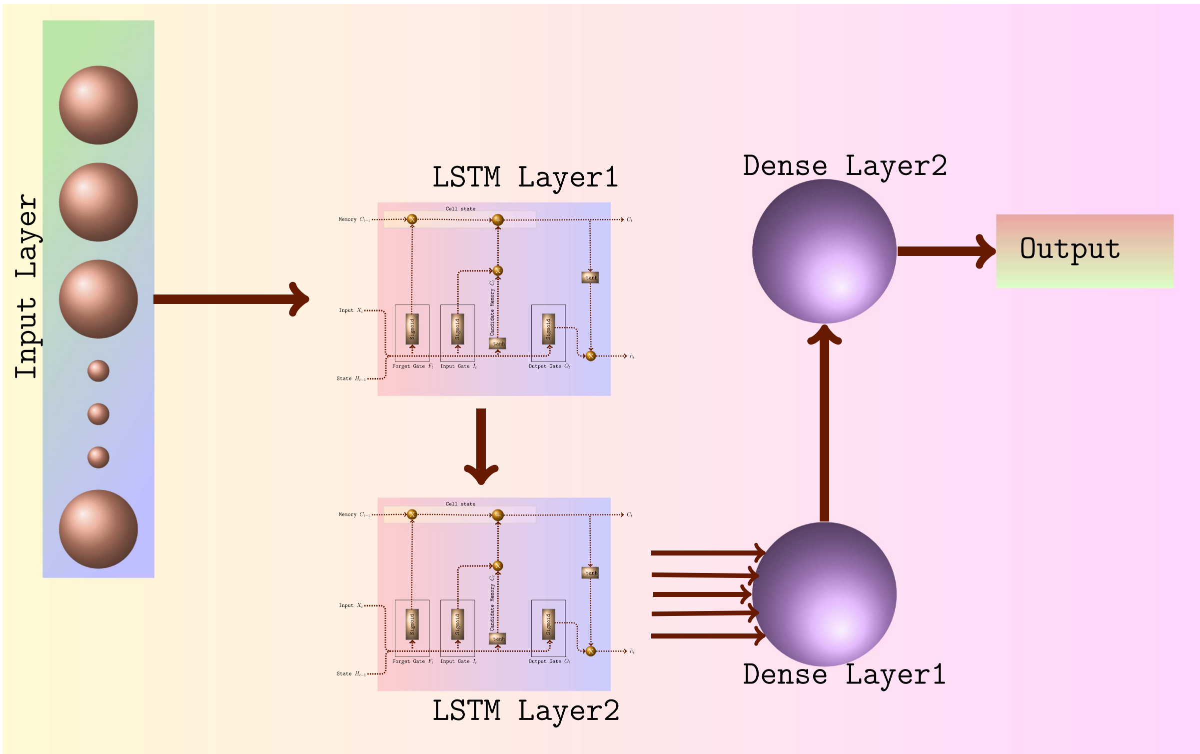


Figure 5
Architecture of MSLSTM model



4.1. The Multi-sequential Long Short-Term Memory model

A deep learning architecture, Multi-sequential Long Short-Term Memory, in short, MSLSTM architecture, is presented in Figure 5, which is designed for sequence modeling, featuring two LSTM layers followed by dense layers. The MSLSTM architecture leverages the LSTM units to identify the sequential correlations in the input data, and the multivariate nature enables it to consider interactions and relationships among multiple features. LSTMs can retain relevant information over extended sequences, allowing the model to effectively learn from historical market behavior. Furthermore, the internal gating mechanism allows selective memory updates, making them robust in dealing with noisy and

Table 2
Model summary: MSLSTM

Layer (Type)	Output shape	Param #
lstm (LSTM)	(None, 42, 210)	181,440
lstm_1 (LSTM)	(None, 210)	353,640
dense (Dense)	(None, 6)	1,266
dense_1 (Dense)	(None, 1)	7
Total params	536,353 (2.05 MB)	
Trainable params	536,353 (2.05 MB)	
Non-trainable params	0 (0.00 MB)	

non-stationary data such as stock prices. These characteristics make LSTM an appropriate choice for modeling the dynamic nature of stock market trends and for generating reliable predictions.

The process begins with the input layer, which receives the sequence data. This data is then fed into the first LSTM layer (LSTM Layer 1), where the temporal dependencies within the sequence are captured. The output from LSTM Layer 1 is subsequently processed by the second LSTM layer (LSTM Layer 2), which further refines the temporal representation.

The refined output from LSTM Layer 2 is then passed to Dense Layer 1. This layer performs a nonlinear transformation of the LSTM output, preparing it for the final prediction. The output from Dense Layer 1 is fed into Dense Layer 2, which acts as the final layer of the network, producing the ultimate output of the model. This architecture is well-suited for tasks requiring sequential data modeling, such as time series forecasting, natural language processing, and other applications where capturing temporal dependencies is crucial. The combination of multiple LSTM layers allows the model to learn complex patterns over time, while the dense layers provide the necessary transformations to map the LSTM outputs to the desired prediction space. The model summary is given in Table 2.

4.2. The Multi-sequential Long Short-Term Memory Autoencoder model

The MSLSTMA architecture is presented in Figure 6, which is designed for sequence-to-sequence tasks. The architecture begins with an input layer that feeds the sequence data into the LSTM encoder.

Following the encoding stage, the latent space representation is passed to the LSTM decoder, which reconstructs the sequence from this compressed form. The decoder processes the latent space representation, expanding it back into a sequence form. The output from the LSTM decoder is then fed into Dense Layer 1, which performs a nonlinear transformation to refine the reconstructed sequence. This refined output is further processed by Dense Layer 2, which produces the final output of the model.

This architecture is particularly effective for tasks requiring the understanding and generation of sequential data, such as time series prediction, anomaly detection, and natural language processing. The autoencoder can discover complex patterns and connections in the data by compressing the input sequence into a latent space and then rebuilding it. This enhances the model's capacity to produce

Figure 6
Architecture of MSLSTMA model

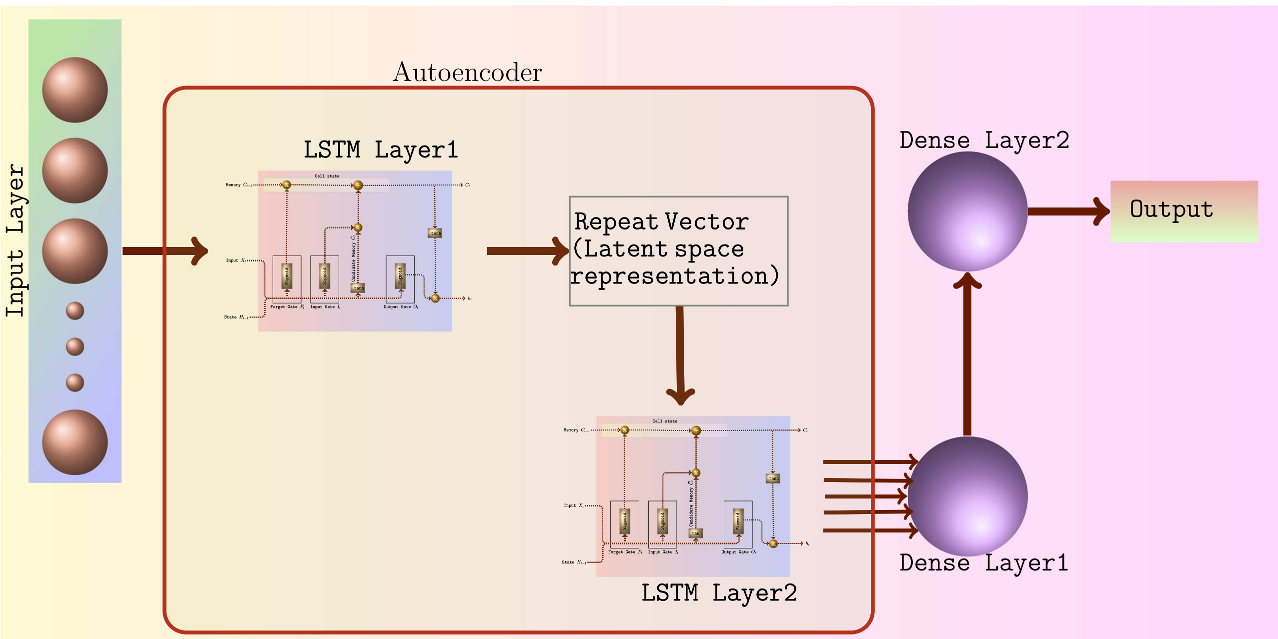


Table 3
Model summary: MSLSTMA

Layer (Type)	Output shape	Param #
lstm (LSTM)	(None, 210)	181,440
repeat_vector (RepeatVector)	(None, 210, 210)	0
lstm_1 (LSTM)	(None, 210)	353,640
dense (Dense)	(None, 6)	1,266
dense_1 (Dense)	(None, 1)	7
Total params	536,353 (2.05 MB)	
Trainable params	536,353 (2.05 MB)	
Non-trainable params	0 (0.00 MB)	

Table 4
Comparative summary of model architectures

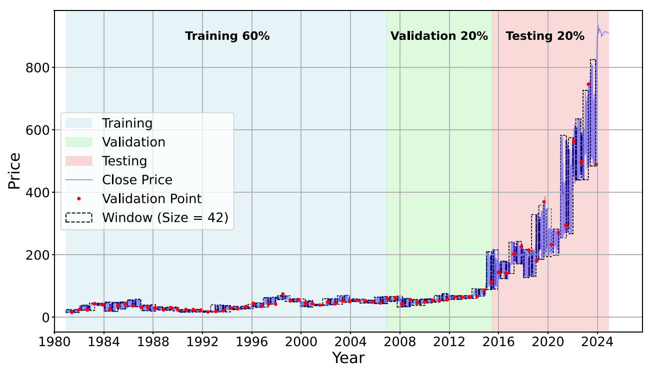
Model	Input type	LSTM layers	Autoencoder
CNNLSTM	Univariate	1	No
USLSTM	Univariate	1	No
USLSTMA	Univariate	1	Yes
MSLSTM	Multivariate	2	No
MSLSTMA	Multivariate	2	Yes

precise and significant outputs. The autoencoder component improves the stock price prediction by performing feature extraction, noise reduction, and dimensionality reduction. The MSLSTMA model summary is given in Table 3.

The primary differences between MSLSTM and MSLSTMA models lie in their architectures and intended functionalities. The architectural configuration of the model, including the number of units in the LSTM layers and neurons in the dense layers, was established through manual tuning. The objective was to achieve the right balance between model complexity and generalizability, ensuring enough capacity to represent temporal connections while avoiding overfitting.

A comparative overview of the main architectural differences between the models in this study is given in Table 4.

Figure 7
Validation process



5. Experimental Setups and Data Collection

Using Eikon Refinitiv as the data source ensures the accuracy and consistency of the financial data used in this research work. Eikon, a widely recognized platform used by financial institutions and researchers, delivers comprehensive and timely market information, which improves the validity of the analysis.

The models employ daily data from December 12, 1980, to February 05, 2024, to train the stock market data, encompassing key aspects like open, high, low, trading price, and volume. As some equities were listed after the initial date, their early records have missing values. These NaN entries are removed from the dataset. The remaining data was normalized using min-max scaling to ensure uniformity across input values during model training.

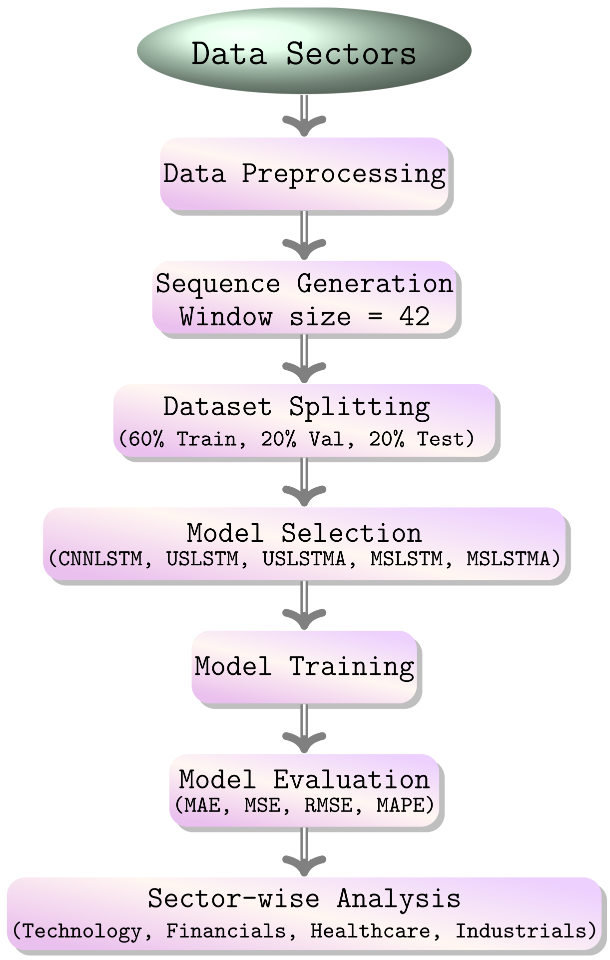
To guarantee the accuracy of time series forecasting, a sliding window approach was used with a fixed window size of 42 time steps. This means that each input to the model includes the previous 42 days of historical data. The dataset was chronologically divided into 60% training, 20% validation, and 20% testing without random shuffling. This sliding window technique was used to build input and output sequences, ensuring that the model is assessed on future values and only learns from historical values. This prevents data leakage and simulates a realistic forecast scenario. By incorporating early stopping and dropout layers, the model manages overfitting by avoiding excessive reliance on training data. Figure 7 illustrates the validation process with a representative temporal data split. The exact split may differ depending on the stock-specific data availability.

Although the models are trained using a single-step prediction mechanism, they are applied recursively for multistep forecasting over an extended horizon of approximately 5 years. The predictions are generated iteratively by predicting the subsequent time point using the past outputs as inputs. Figure 8 offers a detailed flow chart that demonstrates the methodological framework used in this study.

Here are some key points considered for choosing equities:

1) High liquidity (high trading volumes): to ensure that there is enough market activity and trading volume for accurate price discovery, reducing the impact of slippage and ensuring efficient execution of trades.

Figure 8
Methodological framework



2) Volatility: stocks with moderate to high volatility provide more opportunities for trading and testing the predictive capabilities of the models.
3) Industry representation: the selection across different industries to capture varying market dynamics. Testing the models on stocks from different sectors can help evaluate the robustness and generalizability of the models across different market conditions.
4) Market capitalization: stocks with different market capitalizations, including large-cap, mid-cap, and small-cap stocks. Each category has its own characteristics and risk-return profiles, which can impact the behavior of stock prices and the performance of the models.
5) Correlation analysis: to avoid overfitting the models to specific market conditions or stock-specific factors. We chose stocks with low correlation, which can help ensure the diversity of the testing dataset.
6) Availability of external data: incorporating external data sources, such as economic indicators, sentiment analysis, or news sentiment, to enhance the predictive power of the models and capture additional market insights.

Table 5 lists the stock names spanning diverse industries, carefully chosen for evaluating AI models aimed at stock price prediction. The *Sector* column delineates the industry sector to which each equity belongs, while the *Ticker* column provides the associated market ticker symbol. Liquidity levels are presented in the *Liquidity* column, offering insights into the ease of trading each equity. Additionally, the *P/E ratio* column showcases the respective price-to-earnings ratios, a crucial metric for assessing valuation, while the *MarCap* column reveals the market capitalization of each equity, underscoring their respective financial magnitude within the market landscape.

6. Experimental Results and Analysis

The AI models utilized in this study were implemented using Python and executed on a computing device with the following specifications: the device features a 12^{th} Gen Intel Core i7-12700 processor operating at a base frequency of 2.10 GHz. It is equipped with 16.0 GB RAM. The device runs on a 64-bit operating system, Windows 11 Pro edition. These specifications describe the computational infrastructure used to deploy and assess AI models for the research endeavor. Hyperparameter tuning was performed using random search to optimize the number of epochs and batch size for each stock. This approach enabled efficient exploration of the hyperparameter space while avoiding the computational cost of exhaustive search methods. Furthermore, we took model complexity into account and deliberately refrained from using excessively deep architectures to balance representational capacity and training efficiency. This constraint was imposed to maintain model generalizability, prevent overfitting, and guarantee that training could be completed within a reasonable time frame, given the multitude of tests conducted across multiple stocks.

The models are built with Tensorflow and Keras. In each of the models, we tested the commonly used metrics for evaluating the performance of each of the models, which are MSE, root mean squared error (RMSE), mean absolute error (MAE), and mean absolute percentage error (MAPE).

These metrics are used to assess the performance of predictive models and to compare the accuracy of different models in predicting stock prices. Lower values of MSE, RMSE, MAE, and MAPE indicate better predictive performance. Thus, a total of 480 experiments is carried out. The MSLSTMA model consistently outperforms others across most metrics and stocks, demonstrating its superior predictive accuracy. For a comprehensive view of the model performance, including detailed sector-wise comparisons and performance metrics, please refer to the Tables A1–A6.

Table 6 presents the performance results of various AI models in predicting stock prices across different industry sectors – shown in each row. Each cell in the table represents the winning rate of each model (expressed as a percentage) of a specific AI model within a particular sector. The winning rate refers to the proportion of times a particular model outperforms other models across various performance metrics within a sector. For each metric, the model yielding the lowest error is considered as the winner for that instance. The winning rate is then calculated as the percentage of times a model achieves the best performance across all stocks and metrics within that sector.

To instantiate, the performance rate of MSLSTMA is 95.83% in the telecommunication sector. In each sector, we choose six equities from a wide range of selection criteria.

The USLSTM model exhibited some predictive capability, particularly in the technology and financials sectors. However, its accuracy was relatively low compared to later models. The inclusion of moving averages in the USLSTMA model led to slight improvements in winning rates, especially in the financials and industrials sectors. The MSLSTM model exhibited considerable

Table 5
Equities used for AI model testing

Sector	Ticker	Liquidity	P/E ratio	MarCap
	INTC	Large	93.10	High
	NTES	Large	14.70	Low
	AEIS	Medium	27.90	High
	MTCH	Medium	14.11	Low
	VRNT	Small	100.48	High
Technology	VSAT	Small	2.07	Low
	CME	Large	24.02	High
	XP	Large	15.77	Low
	MKTX	Medium	29.78	High
	SEIC	Medium	19.36	Low
	PLMR	Small	25.39	High
Financials	EFSC	Small	7.55	Low
	ISRG	Large	68.06	High
	GILD	Large	14.90	Low
	RGEN	Medium	237.89	High
	ROIV	Medium	2.10	Low
	CPRX	Small	24.80	High
Healthcare	AMPH	Small	15.81	Low
	HON	Large	22.94	High
	CSX	Large	18.87	Low
	ESLT	Medium	41.72	High
	MIDD	Medium	19.24	Low
	USLM	Small	22.48	High
Industrials	ANDE	Small	19.81	Low

Table 6
Sector-wise winning rate of each model

	TEST				
SECTOR	CNNLSTM	USLSTM	USLSTMA	MSLSTM	MSLSTMA
TECHNOLOGY	0.00%	8.33%	0.00%	4.17%	87.50%
FINANCIALS	0.00%	12.50%	0.00%	37.50%	50.00%
HEALTHCARE	0.00%	0.00%	8.33%	45.83%	45.83%
CONSUMER STAPLES	0.00%	0.00%	8.33%	50.00%	41.67%
INDUSTRIALS	0.00%	0.00%	0.00%	16.67%	83.33%
TELECOMMUNICATION	0.00%	0.00%	0.00%	4.17%	95.83%

improvements over baseline models such as CNNLSTM, USLSTM, and USLSTMA across multiple sectors.

Finally, the MSLSTMA model emerged as the most effective across all sectors, demonstrating the highest prediction accuracy. This model achieved remarkable accuracy, especially in the technology sector, indicating its robust predictive capabilities when considering a combination of multivariate data and moving averages.

In summary, while early models struggled to provide accurate predictions, the more advanced models, particularly the MSLSTMA model, showcased promising predictive capabilities, offering potential utility in forecasting stock prices across diverse industry sectors.

7. Conclusion and Future Directions

In this work, we have proposed and evaluated two innovative AI models based on LSTM networks for stock price prediction. The first approach deepens the LSTM model by adding multiple layers, allowing the model to capture hierarchical patterns and capture temporal dependencies. The second approach is a hybrid model that integrates an autoencoder with the LSTM network to enhance feature extraction, noise reduction, and improvement of the generalization capabilities of the model.

Our findings demonstrate that both variations offer significant improvements to the LSTM model. Even while our models perform better, several limitations remain. Hyperparameter tuning was conducted manually, which, while effective in our case, may hinder scalability across diverse datasets or domains. Additionally, like most deep learning models, the proposed approach requires substantial data and computational resources. Future directions include incorporating automated hyperparameter optimization methods. A thorough knowledge of the strengths and trade-offs of our models could be obtained by extending comparisons to a wider range of machine learning models and alternate hybrid techniques. We believe this research can support investors in evaluating assets more effectively and making informed investment decisions.

Acknowledgments

We thank the anonymous reviewers for their constructive comments, which helped us to improve the manuscript.

Ethical Statement

This study does not contain any studies with human or animal subjects performed by any of the authors.

Conflicts of Interest

The authors declare that they have no conflicts of interest to this work.

Data Availability Statement

Data sharing is not applicable to this article as no new data were created or analyzed in this study.

Author Contribution Statement

Sheba Elizabeth Thomas: Methodology, Validation, Investigation, Data curation, Writing – review & editing, Visualization, Supervision. **Rubell Marion Lincy George:** Conceptualization, Validation, Resources, Writing – original draft, Project administration. **Nevin Selby:** Conceptualization, Software, Formal analysis, Writing – original draft, Project administration. **Aditya Taparia:** Software, Formal analysis, Supervision. **Jobish Vallikavungal:** Methodology, Validation, Investigation, Resources, Data curation, Writing – review & editing, Visualization, Supervision.

References

[1] Gupta, A., Akansha., Joshi, K., Patel, M., & Pratap, V. (2023). Stock market prediction using machine learning techniques: A systematic review. In *Proceedings of the 2023 International Conference on Power, Instrumentation, Control and Computing*, 1–6. https://doi.org/10.1109/PICC57976.2023.10142862

[2] Shah, J., Vaidya, D., & Shah, M. (2022). A comprehensive review on multiple hybrid deep learning approaches for stock prediction. *Intelligent Systems with Applications*, *16*, 200111. https://doi.org/10.1016/j.iswa.2022.200111

[3] Jiang, W. (2021). Applications of deep learning in stock market prediction: Recent progress. *Expert Systems with Applications*, *184*, 115537. https://doi.org/10.1016/j.eswa.2021.115537

[4] 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

[5] Asgarian, S., Ghasemi, R., & Momtazi, S. (2023). Generative adversarial network for sentiment-based stock prediction. *Concurrency and Computation: Practice and Experience*, *35*(2), e7467. https://doi.org/10.1002/cpe.7467

[6] Bhandari, H. N., Rimal, B., Pokhrel, N. R., Rimal, R., Dahal, K. R., & Khatri, R. K. C. (2022). Predicting stock market index using LSTM. *Machine Learning with Applications*, *9*, 100320. https://doi.org/10.1016/j.mlwa.2022.100320

[7] Wang, J., Cui, Q., Sun, X., & He, M. (2022). Asian stock markets closing index forecast based on secondary decomposition, multi-factor analysis and attention-based LSTM model. *Engineering Applications of Artificial Intelligence*, *113*, 104908. https://doi.org/10.1016/j.engappai.2022.104908

[8] Polamuri, S. R., Srinivas, K., & Mohan, A. K. (2022). Multi-model generative adversarial network hybrid prediction algorithm (MMGAN-HPA) for stock market prices prediction. *Journal of King Saud University—Computer and Information Sciences*, *34*(9), 7433–7444. https://doi.org/10.1016/j.jksuci.2021.07.001

[9] Liu, H., & Long, Z. (2020). An improved deep learning model for predicting stock market price time series. *Digital Signal Processing*, *102*, 102741. https://doi.org/10.1016/j.dsp.2020.102741

[10] Lin, Y., Lin, Z., Liao, Y., Li, Y., Xu, J., & Yan, Y. (2022). Forecasting the realized volatility of stock price index: A hybrid model integrating CEEMDAN and LSTM. *Expert Systems with Applications*, *206*, 117736. https://doi.org/10.1016/j.eswa.2022.117736

[11] Vidal, A., & Kristjanpoller, W. (2020). Gold volatility prediction using a CNN-LSTM approach. *Expert Systems with Applications*, *157*, 113481. https://doi.org/10.1016/j.eswa.2020.113481

[12] Zolfaghari, M., & Gholami, S. (2021). A hybrid approach of adaptive wavelet transform, long short-term memory and ARIMA-GARCH family models for the stock index prediction. *Expert Systems with Applications*, *182*, 115149. https://doi.org/10.1016/j.eswa.2021.115149

[13] Chong, E., Han, C., & Park, F. C. (2017). Deep learning networks for stock market analysis and prediction: Methodology, data representations, and case studies. *Expert Systems with Applications*, *83*, 187–205. https://doi.org/10.1016/j.eswa.2017.04.030

[14] Zhang, Y. A., Yan, B., & Aasma, M. (2020). A novel deep learning framework: Prediction and analysis of financial time series using CEEMD and LSTM. *Expert Systems with Applications*, *159*, 113609. https://doi.org/10.1016/j.eswa.2020.113609

[15] Gao, Y., Wang, R., & Zhou, E. (2021). Stock prediction based on optimized LSTM and GRU models. *Scientific Programming*, *2021*(1), 4055281. https://doi.org/10.1155/2021/4055281

[16] Kim, H. Y., & Won, C. H. (2018). Forecasting the volatility of stock price index: A hybrid model integrating LSTM with multiple GARCH-type models. *Expert Systems with Applications*, *103*, 25–37. https://doi.org/10.1016/j.eswa.2018.03.002

[17] Bai, J., Guo, J., Sun, B., Guo, Y., Bao, Q., & Xiao, X. (2023). Intelligent forecasting model of stock price using neighborhood rough set and multivariate empirical mode decomposition. *Engineering Applications of Artificial Intelligence*, *122*, 106106. https://doi.org/10.1016/j.engappai.2023.106106

[18] Chen, S., & Ge, L. (2019). Exploring the attention mechanism in LSTM-based Hong Kong stock price movement prediction. *Quantitative Finance*, *19*(9), 1507–1515. https://doi.org/10.1080/14697688.2019.1622287

[19] He, B., & Kita, E. (2020). Stock price prediction by using hybrid sequential generative adversarial networks. In *2020 International Conference on Data Mining Workshops*, 341–347. https://doi.org/10.1109/ICDMW51313.2020.00054

[20] Ding, G., & Qin, L. (2020). Study on the prediction of stock price based on the associated network model of LSTM.

International Journal of Machine Learning and Cybernetics, *11*(6), 1307–1317. https://doi.org/10.1007/s13042-019-01041-1

[21] He, B., & Kita, E. (2021). The application of sequential generative adversarial networks for stock price prediction. *The Review of Socionetwork Strategies*, *15*(2), 455–470. https://doi.org/10.1007/s12626-021-00097-2

[22] He, K., & Zhang, Z. (2022). Prediction stock price based on CNN and LSTM models. *Financial Engineering and Risk Management*, *5*(7), 14–21. http://dx.doi.org/10.23977/ferm.2022.050703

[23] Kumar, A., Alsadoon, A., Prasad, P. W. C., Abdullah, S., Rashid, T. A., Pham, D. T. H., & Nguyen, T. Q. V. (2022). Generative adversarial network (GAN) and enhanced root mean square error (ERMSE): Deep learning for stock price movement prediction. *Multimedia Tools and Applications*, *81*(3), 3995–4013. https://doi.org/10.1007/s11042-021-11670-w

[24] Zhou, X., Pan, Z., Hu, G., Tang, S., & Zhao, C. (2018). Stock market prediction on high-frequency data using generative adversarial nets. *Mathematical Problems in Engineering*, *2018*(1), 4907423. https://doi.org/10.1155/2018/4907423

[25] Wu, J. M.-T., Li, Z., Herencsar, N., Vo, B., & Lin, J. C.-W. (2023). A graph-based CNN-LSTM stock price prediction algorithm with leading indicators. *Multimedia Systems*, *29*(3), 1751–1770. https://doi.org/10.1007/s00530-021-00758-w

[26] Jin, Z., Yang, Y., & Liu, Y. (2020). Stock closing price prediction based on sentiment analysis and LSTM. *Neural Computing and Applications*, *32*(13), 9713–9729. https://doi.org/10.1007/s00521-019-04504-2

[27] Lee, S. W., & Kim, H. Y. (2020). Stock market forecasting with super-high dimensional time-series data using ConvLSTM, trend sampling, and specialized data augmentation. *Expert Systems with Applications*, *161*, 113704. https://doi.org/10.1016/j.eswa.2020.113704

[28] 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

[29] Md, A. Q., Kapoor, S., Chris Junni, A. V., Sivaraman, A. K., Tee, K. F., Sabireen, H., & Janakiraman, N. (2023). Novel optimization approach for stock price forecasting using multi-layered sequential LSTM. *Applied Soft Computing*, *134*, 109830. https://doi.org/10.1016/j.asoc.2022.109830

[30] Zhang, Y., Song, Y., & Wei, G. (2023). A feature-enhanced long short-term memory network combined with residual-driven ν support vector regression for financial market prediction. *Engineering Applications of Artificial Intelligence*, *118*, 105663. https://doi.org/10.1016/j.engappai.2022.105663

[31] 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

[32] Ito, K., Iima, H., & Kitamura, Y. (2022). LSTM forecasting foreign exchange rates using limit order book. *Finance Research Letters*, *47*, 102517. https://doi.org/10.1016/j.frl.2021.102517

[33] Mirza, A. H., Kerpicci, M., & Kozat, S. S. (2020). Efficient online learning with improved LSTM neural networks. *Digital Signal Processing*, *102*, 102742. https://doi.org/10.1016/j.dsp.2020.102742

[34] Patel, J., Shah, S., Thakkar, P., & Kotecha, K. (2015). Predicting stock market index using fusion of machine learning techniques. *Expert Systems with Applications*, *42*(4), 2162–2172. https://doi.org/10.1016/j.eswa.2014.10.031

[35] Takahashi, M., Watanabe, T., & Omori, Y. (2024). Forecasting daily volatility of stock price index using daily returns and realized volatility. *Econometrics and Statistics, 32, 34–56*. https://doi.org/10.1016/j.ecosta.2021.08.002

[36] Wang, J., & Chen, Z. (2024). Factor-GAN: Enhancing stock price prediction and factor investment with generative adversarial networks. *PLOS ONE*, *19*(6), e0306094. https://doi.org/10.1371/journal.pone.0306094

[37] Wu, J. M.-T., Li, Z., Srivastava, G., Tasi, M.-H., & Lin, J. C.-W. (2021). A graph-based convolutional neural network stock price prediction with leading indicators. *Software: Practice and Experience*, *51*(3), 628–644. https://doi.org/10.1002/spe.2915

[38] Zhao, Y., Upreti, V., & Cai, Y. (2021). Stock returns, quantile autocorrelation, and volatility forecasting. *International Review of Financial Analysis*, *73*, 101599. https://doi.org/10.1016/j.irfa.2020.101599

[39] Hu, Z., Zhao, Y., & Khushi, M. (2021). A survey of forex and stock price prediction using deep learning. *Applied System Innovation*, *4*(1), 9. https://doi.org/10.3390/asi4010009

[40] Dezhkam, A., & Manzuri, M. T. (2023). Forecasting stock market for an efficient portfolio by combining XGBoost and Hilbert–Huang transform. *Engineering Applications of Artificial Intelligence*, *118*, 105626. https://doi.org/10.1016/j.engappai.2022.105626

[41] Kang, H., Zong, X., Wang, J., & Chen, H. (2023). Binary gravity search algorithm and support vector machine for forecasting and trading stock indices. *International Review of Economics & Finance*, *84*, 507–526. https://doi.org/10.1016/j.iref.2022.11.009

[42] Lu, W., Li, J., Li, Y., Sun, A., & Wang, J. (2020). A CNN-LSTM-based model to forecast stock prices. *Complexity*, *2020*(1), 6622927. https://doi.org/10.1155/2020/6622927

[43] Pai, P.-F., & Lin, C.-S. (2005). A hybrid ARIMA and support vector machines model in stock price forecasting. *Omega*, *33*(6), 497–505. https://doi.org/10.1016/j.omega.2004.07.024

[44] Gandhmal, D. P., & Kumar, K. (2019). Systematic analysis and review of stock market prediction techniques. *Computer Science Review*, *34*, 100190. https://doi.org/10.1016/j.cosrev.2019.08.001

[45] Kehinde, T. O., Chan, F. T. S., & Chung, S. H. (2023). Scientometric review and analysis of recent approaches to stock market forecasting: Two decades survey. *Expert Systems with Applications*, *213*, 119299. https://doi.org/10.1016/j.eswa.2022.119299

[46] Mehtarizadeh, H., Mansouri, N., Zade, B. M. H., & Hosseini, M. M. (2025). Stock price prediction with SCA-LSTM network and statistical model ARIMA-GARCH. *The Journal of Supercomputing*, *81*(2), 366. https://doi.org/10.1007/s11227-024-06775-6

[47] Wang, S., Ji, T., Wang, L., Sun, Y., Liu, S.-C., Kumar, A., & Lu, C.-T. (2024). StockTime: A time series specialized large language model architecture for stock price prediction. *arXiv Preprint:2409.08281*. https://doi.org/10.48550/arXiv.2409.08281

[48] Jin, B., & Xu, X. (2024). Machine learning predictions of regional steel price indices for east China. *Ironmaking & Steelmaking: Processes, Products and Applications, 52*(3), *368–381*. https://doi.org/10.1177/03019233241254891

[49] Jin, B., & Xu, X. (2024). Palladium price predictions via machine learning. *Materials Circular Economy*, *6*(1), 32. https://doi.org/10.1007/s42824-024-00123-y

[50] Jin, B., & Xu, X. (2024). Gaussian process regression based silver price forecasts. *Journal of Uncertain Systems*, *17*(03), 2450013. https://doi.org/10.1142/S1752890924500132

How to Cite: Thomas, S. E., George, R. M. L., Selby, N., Taparia, A., & Vallikavungal, J. (2025). Multi-sequential Neural Network Models for Stock Price Forecasting. *Journal of Computational and Cognitive Engineering*. https://doi.org/10.47852/bonviewJCCE52025731

Appendix

Here, you will see the entire experiments.

Table A1
Performance comparison of predictive models for technology sector equities

		Model Name					
Ticker	Metric	USLSTM	USLSTMA	MSLSTM	CNNLSTM	MSLSTMA	Winner
	MSE	2.34	1.77	1.79	4.62	1.70	MSLSTMA
	RMSE	1.53	1.33	1.34	2.15	1.30	MSLSTMA
	MAE	1.06	0.91	0.92	1.54	0.88	MSLSTMA
INTC	MAPE	2.44	2.09	2.13	3.60	2.04	MSLSTMA
	MSE	745.04	825.04	489.30	873.97	351.98	MSLSTMA
	RMSE	27.30	28.72	22.12	29.56	18.76	MSLSTMA
	MAE	13.95	15.05	9.15	12.28	8.33	MSLSTMA
NTES	MAPE	6.41	6.62	5.96	8.07	5.77	MSLSTMA
	MSE	159.12	16.29	10.71	27.98	9.60	MSLSTMA
	RMSE	12.61	4.04	3.27	5.29	3.10	MSLSTMA
	MAE	4.77	2.99	2.44	4.01	2.29	MSLSTMA
AEIS	MAPE	2.94	3.68	3.12	4.94	3.00	USLSTM
	MSE	10.52	8.95	6.27	21.14	5.55	MSLSTMA
	RMSE	3.24	2.99	2.50	4.59	2.36	MSLSTMA
	MAE	2.41	2.33	1.89	3.67	1.73	MSLSTMA
MTCH	MAPE	3.16	5.17	4.09	8.42	3.70	USLSTM
	MSE	3.00	2.95	3.01	7.55	1.75	MSLSTMA
	RMSE	1.73	1.72	1.73	2.75	1.32	MSLSTMA
	MAE	1.03	1.07	1.01	1.77	0.90	MSLSTMA
VRNT	MAPE	2.42	2.51	2.40	4.21	2.96	MSLSTM
	MSE	4.62	4.72	4.59	12.28	4.42	MSLSTMA
	RMSE	2.15	2.17	2.14	3.50	2.10	MSLSTMA
	MAE	1.54	1.56	1.54	2.62	1.49	MSLSTMA
VSAT	MAPE	3.63	3.69	3.59	6.00	3.51	MSLSTMA

Table A2
Performance comparison of predictive models for financial sector equities

		Model Name					
Ticker	Metric	USLSTM	USLSTMA	MSLSTM	CNNLSTM	MSLSTMA	Winner
	MSE	22.20	28.16	21.75	58.19	21.98	MSLSTM
	RMSE	4.71	5.31	4.66	7.63	4.69	MSLSTM
	MAE	3.30	3.71	3.27	5.34	3.31	MSLSTM
CME	MAPE	1.69	1.91	1.68	2.80	1.71	MSLSTM
	MSE	2.49	1.25	1.54	2.69	1.01	MSLSTMA
	RMSE	1.58	1.12	1.24	1.64	1.01	MSLSTMA
	MAE	1.22	0.84	0.98	1.33	0.77	MSLSTMA
XP	MAPE	5.65	3.84	4.60	6.10	3.55	MSLSTMA
	MSE	159.12	423.76	179.78	517.43	189.49	USLSTM
	RMSE	12.61	20.58	13.41	22.75	13.77	USLSTM
	MAE	4.76	15.28	9.54	17.89	9.89	USLSTM
MKTX	MAPE	2.94	3.83	2.63	4.94	2.74	MSLSTM
	MSE	1.88	1.92	1.72	3.08	1.72	MSLSTMA
	RMSE	1.37	1.39	1.31	1.75	1.31	MSLSTMA
	MAE	0.98	0.99	0.94	1.29	0.94	MSLSTMA
SEIC	MAPE	1.78	1.79	1.71	2.34	1.70	MSLSTMA
	MSE	4.08	3.71	3.48	8.98	3.52	MSLSTM
	RMSE	2.02	1.93	1.87	2.99	1.88	MSLSTM
	MAE	1.53	1.47	1.37	2.49	1.42	MSLSTM
PLMR	MAPE	2.77	2.67	2.48	4.52	2.57	MSLSTM
	MSE	2.71	2.19	1.85	2.76	1.58	MSLSTMA
	RMSE	1.65	1.48	1.36	1.66	1.26	MSLSTMA
	MAE	1.23	1.22	0.99	1.24	0.92	MSLSTMA
EFSC	MAPE	3.12	2.77	2.49	3.11	2.32	MSLSTMA

Table A3
Performance comparison of predictive models for healthcare sector equities

			Model name				
Ticker	Metric	USLSTM	USLSTMA	MSLSTM	CNNLSTM	MSLSTMA	Winner
	MSE	967.65	1045.54	942.24	1380.27	940.79	MSLSTMA
	RMSE	31.11	32.33	30.70	37.15	30.67	MSLSTMA
	MAE	13.26	14.55	12.22	18.24	12.23	MSLSTM
ISRG	MAPE	2.99	3.03	2.78	4.00	2.77	MSLSTMA
	MSE	2.85	3.14	2.55	5.76	2.56	MSLSTM
	RMSE	1.69	1.77	1.60	2.40	1.60	MSLSTMA
	MAE	1.26	1.35	1.16	1.79	1.16	MSLSTMA
GILD	MAPE	1.78	1.92	1.63	2.51	1.64	MSLSTM
	MSE	171.13	213.07	46.64	191.38	52.11	MSLSTM
	RMSE	13.08	14.60	6.83	13.83	7.22	MSLSTM
	MAE	8.44	9.44	4.21	9.33	4.47	MSLSTM
RGEN	MAPE	5.29	5.93	3.11	6.86	3.27	MSLSTM
	MSE	0.54	0.40	0.46	0.86	0.43	USLSTMA
	RMSE	0.73	0.63	0.68	0.93	0.66	USLSTMA
	MAE	0.59	0.48	0.53	0.76	0.47	MSLSTMA
ROIV	MAPE	5.69	4.54	5.06	7.29	4.50	MSLSTMA
	MSE	2.25	1.69	0.41	1.16	0.64	MSLSTM
	RMSE	1.50	1.30	0.64	1.08	0.80	MSLSTM
	MAE	0.99	0.87	0.40	0.73	0.53	MSLSTM
CPRX	MAPE	7.76	6.98	3.80	6.37	4.66	MSLSTM
	MSE	8.54	9.62	6.53	12.68	4.49	MSLSTMA
	RMSE	2.92	3.10	2.56	3.56	2.12	MSLSTMA
	MAE	2.14	2.28	1.86	2.80	1.46	MSLSTMA
AMPH	MAPE	4.74	5.04	2.23	6.64	3.38	MSLSTMA

Table A4
Performance comparison of predictive models for consumer staples sector equities

			Model name				
Ticker	Metric	USLSTM	USLSTMA	MSLSTM	CNNLSTM	MSLSTMA	Winner
	MSE	15.84	14.98	15.76	36.18	14.84	MSLSTMA
	RMSE	3.98	3.87	3.97	6.01	3.85	MSLSTMA
	MAE	1.48	1.38	1.44	2.75	1.46	USLSTMA
MNST	MAPE	2.29	2.09	2.22	4.43	2.22	USLSTMA
	MSE	3.27	5.41	1.42	4.47	1.84	MSLSTM
	RMSE	1.81	2.32	1.19	2.12	1.36	MSLSTM
	MAE	1.50	1.91	0.90	1.67	1.07	MSLSTM
MDLZ	MAPE	2.39	2.96	1.49	2.71	1.73	MSLSTM
	MSE	0.51	0.51	0.51	0.74	0.49	MSLSTMA
	RMSE	0.72	0.71	0.71	0.86	0.70	MSLSTMA
	MAE	0.50	0.50	0.49	0.62	0.49	MSLSTMA
PPC	MAPE	1.96	1.96	1.95	2.42	1.92	MSLSTMA
	MSE	1101.37	1032.12	318.07	1510.83	348.02	MSLSTM
	RMSE	33.19	32.13	17.83	38.87	18.66	MSLSTM
	MAE	20.93	20.27	11.25	28.88	11.69	MSLSTM
COKE	MAPE	4.51	4.45	2.78	7.25	3.07	MSLSTM
	MSE	2.56	1.96	2.43	3.46	1.95	MSLSTMA
	RMSE	1.60	1.40	1.56	1.86	1.39	MSLSTMA
	MAE	1.17	1.02	1.18	1.51	1.00	MSLSTMA
COCO	MAPE	4.50	3.98	4.59	5.87	3.87	MSLSTMA
	MSE	26.29	39.15	9.48	22.41	11.09	MSLSTM
	RMSE	5.13	6.26	3.08	4.73	3.33	MSLSTM
	MAE	3.88	4.62	2.23	3.64	2.42	MSLSTM
MGPI	MAPE	4.83	5.54	3.07	5.31	3.25	MSLSTM

Table A5
Performance comparison of predictive models for industrial sector equities

		Model name				
Ticker	Metric	USLSTM	USLSTMA	MSLSTM	CNNLSTM	MSLSTMA	Winner
Ticker	Model Name	USLSTM	USLSTMA	MSLSTM	CNNLSTM	MSLSTMA	Winner
	MSE	28.83	86.56	23.09	67.97	44.29	MSLSTM
	RMSE	5.37	9.30	4.81	8.24	6.66	MSLSTM
	MAE	4.10	7.71	3.76	6.27	5.49	MSLSTM
HON	MAPE	2.32	4.06	2.06	3.57	2.94	MSLSTM
	MSE	5.73	5.44	5.64	6.96	5.39	MSLSTMA
	RMSE	2.39	2.33	2.37	2.64	2.32	MSLSTMA
	MAE	1.09	0.98	1.04	1.37	0.96	MSLSTMA
CSX	MAPE	2.22	2.05	2.26	3.21	2.03	MSLSTMA
	MSE	80.81	106.88	61.12	56.23	33.90	MSLSTMA
	RMSE	8.99	10.34	7.82	7.50	5.82	MSLSTMA
	MAE	6.78	8.07	5.81	5.44	4.30	MSLSTMA
ESLT	MAPE	3.87	4.57	3.38	3.34	2.50	MSLSTMA
	MSE	18.11	19.35	16.77	55.33	16.23	MSLSTMA
	RMSE	4.26	4.40	4.10	7.44	4.03	MSLSTMA
	MAE	3.11	3.20	2.94	5.57	2.89	MSLSTMA
MIDD	MAPE	2.56	2.63	2.42	4.54	2.37	MSLSTMA
	MSE	39.30	87.58	30.20	45.32	21.75	MSLSTMA
	RMSE	6.27	9.36	5.50	6.73	4.66	MSLSTMA
	MAE	3.97	5.75	3.53	4.57	2.92	MSLSTMA
USLM	MAPE	3.00	4.12	2.76	3.88	2.37	MSLSTMA
	MSE	2.15	2.21	2.03	2.83	1.76	MSLSTMA
	RMSE	1.47	1.49	1.42	1.68	1.33	MSLSTMA
	MAE	1.02	1.03	0.99	1.19	0.91	MSLSTMA
ANDE	MAPE	3.35	3.43	3.31	4.05	2.99	MSLSTMA

Table A6
Performance comparison of predictive models for telecommunication sector equities

			Model name			
Ticker	Metric	USLSTM	USLSTMA	MSLSTM	CNNLSTM	MSLSTMA	Winner
	MSE	12.20	16.62	9.25	31.48	7.83	MSLSTMA
	RMSE	3.49	4.08	3.04	5.61	2.80	MSLSTMA
	MAE	2.85	3.39	2.34	4.40	2.16	MSLSTMA
TMUS	MAPE	2.08	2.46	1.78	3.36	1.62	MSLSTMA
	MSE	1.12	1.19	1.15	2.34	1.12	MSLSTMA
	RMSE	1.06	1.09	1.07	1.53	1.06	MSLSTMA
	MAE	0.75	0.79	0.75	1.12	0.74	MSLSTMA
CSCO	MAPE	1.60	1.71	1.62	2.42	1.60	MSLSTMA
	MSE	1.25	1.07	1.15	1.86	1.05	MSLSTMA
	RMSE	1.12	1.04	1.07	1.36	1.03	MSLSTMA
	MAE	0.91	0.83	0.87	1.12	0.81	MSLSTMA
FYBR	MAPE	4.96	4.42	4.76	6.10	4.34	MSLSTMA
	MSE	14.50	15.24	13.45	29.21	13.21	MSLSTMA
	RMSE	3.81	3.90	3.67	5.40	3.63	MSLSTMA
	MAE	2.78	2.81	2.64	3.99	2.61	MSLSTMA
LBRDK	MAPE	3.09	3.13	2.96	4.44	2.91	MSLSTMA
	MSE	7.22	21.00	5.91	11.38	4.34	MSLSTMA
	RMSE	2.69	4.58	2.43	3.37	2.08	MSLSTMA
	MAE	1.72	3.01	1.72	2.28	1.26	MSLSTMA
NSSC	MAPE	6.75	10.08	6.59	8.11	5.06	MSLSTMA
	MSE	0.50	0.66	0.38	1.44	0.38	MSLSTMA
	RMSE	0.68	0.81	0.61	1.20	0.62	MSLSTM
	MAE	0.45	0.58	0.40	0.88	0.40	MSLSTMA
	MAPE	4.26	5.52	3.89	8.80	3.84	MSLSTMA