Optimization and Validation of an Intelligent Machine Learning Framework for Robust Financial Market Time Series Forecasting

Ganesh R Dhamal[1], Pooja Soni[2], Sunil Kumar Sharma[3], Mukesh Kumar Gupta[4]
[1] Department of Mathematics (Applied Sciences), Suresh Gyan Vihar University Jaipur
[2] Department of Computer Science and Engineering, Suresh Gyan Vihar University Jaipur
[3] Department of Mathematics, Arya College of Engineering, Jaipur.
[4] Department of Electrical engineering, Suresh Gyan Vihar University

Corresponding author email: dhamal.ganesh@yahoo.com

Abstract: The problem of forecasting time series is nonlinear, dynamic and volatile that accurate prediction is still a challenge for financial market. This paper introduces an optimized and validated intelligent machine learning-based framework for robust financial market time series forecasting (TSF). The overall framework will include data preprocessing, feature engineering, hyperparameter tuning, and model validation, which will aim to enhance the model's predictive performance and adaptability to various market scenarios. The various forecasting models such as ARIMA, Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), Transformer, and the proposed optimized framework are compared with historical financial market data. Hyperparameter optimization is implemented using Bayesian optimization to find the optimal configuration of the models and five-fold cross validation/cross market validation for robustness and reliability. Classification metrics are used to assess performance, including accuracy, precision, recall, F1-score and area under the receiver operating characteristic curve (AUC), alongside regression metrics such as root mean square error (RMSE), mean absolute error (MAE), mean absolute percentage error (MAPE), and coefficient of determination (R^2). The results of the experiments show that the proposed framework can work well and offer high forecasting accuracy of 98.74%, and high AUC of 0.995, RMSE of 0.026 and withstand market conditions with noise. The results validate the new framework can deliver accurate, robust and generalisable forecasts for financial markets, which are suitable for intelligent investment decision support and real-world financial analytics applications.

Keywords: Financial Market Forecasting, Time Series Prediction, Machine Learning, Bayesian Hyperparameter Optimization, Robust Model Validation.

I. INTRODUCTION

The financial markets produce vast amounts of time series data, which has complex, nonlinear and highly dynamic characteristics, depending on economic indicators, geopolitical events, investor sentiment and market volatility [1]. The precise prediction of financial market dynamics is crucial for portfolio management, algorithm trading, risk assessment, and investment decision-making [2]. But, the stochasticity of financial time series poses serious challenges for classical statistical time-series forecasting techniques, which has led to the use of intelligent machine learning (ML) approaches that can model complex temporal structure and nonlinear relationships [3].

The classic forecasting models like AutoRegressive Integrated Moving Average (ARIMA) have proven successful for linear and stationary data sources, but fail to model the nonlinear nature of today's financial markets 4]. In recent years, the accuracy of forecasting has significantly improved with the help of ML and deep learning techniques such as LSTM, GRU, and Transformer-based architectures, which have been able to capture and learn the long-term dependencies and complex sequential patterns in historical market data [5]. However, the effectiveness of these models is highly sensitive to the optimal hyperparameter tuning, feature engineering, and extensive testing, and extending their applicability to other market segments is still a key research challenge.

This paper suggests an optimized and validated intelligent ML framework that strengthens the TSF of the financial market. Multiple forecasting models are comparatively analyzed by both classification and regression performance assessment metrics, in terms of prediction accuracy, robustness, computational efficiency and generalization ability. The experimental findings clearly show that the proposed framework outperforms the traditional forecasting models and

state-of-the-art forecasting methods, in terms of accuracy of prediction and forecasting error, in various financial market sectors.

II. LITERATURE REVIEW

Forecasting financial markets has been a focal point of research attention because of its importance in investment planning, portfolio optimization and risk management. For financial time series prediction, traditional statistical methods, such as ARIMA model have been widely used due to their simplicity and interpretability [6,7]. But in general, these models are linear and stationary making them poor models of the complex and nonlinear dynamics of today's financial markets.

With the recent developments in ML, more accurate modelling of financial time series has become possible [8]. Random Forest (RF) and Extreme Gradient Boosting (XGBoost) are two of the ensemble models that have shown to be more effective models by learning the non-linear relationships from the past market indicators [9]. These techniques work well for very high-dimensional data but are not very effective at capturing long-term temporal dependencies found in sequential financial data.

Moreover, deep learning architectures are also making significant strides in the field of financial forecasting by automatically learning hierarchical temporal features. Their ability to model long-range dependencies and to reduce the vanishing gradient problem has made LSTM networks a popular choice. Likewise, GRU models are simpler than LSTM and still perform well as forecasters [10, 11]. Lately, models based on Transformers, which use self-attention mechanisms, have been developed to get state-of-the-art performance by learning global temporal relationships and parallel sequence representations.

While these smart forecasting systems have significantly boosted the accuracy of forecasts, there are still a number of challenges to be faced. The hyperparameters, data preprocessing strategies and feature engineering techniques are critical to the model performance. [12] Moreover, numerous previous studies assess the forecasting models based on a single financial market data and thus are unable to apply to a wide range of financial sectors. The validation by cross-market evaluation, verification of robustness in noisy markets and the computational efficiency assessment have received less attention.

The proposed methodology attempts to create an optimized and validated intelligent ML framework to forecast time series in financial markets with robust accuracy. The framework involves six key stages: Data acquisition, Data preprocessing, Feature engineering, Model development, Hyperparameter optimization, and Performance validation. The methodology is presented in the form of a systematic workflow in the proposed framework, which is depicted in figure 1.

III. METHODOLOGY

The proposed methodology aims to develop an optimized and validated intelligent ML framework for robust financial market TSF. The framework consists of six major stages: data acquisition, data preprocessing, feature engineering, model development, hyperparameter optimization, and performance validation. A systematic workflow of the proposed framework is illustrated in the methodology diagram as shown in figure 1.

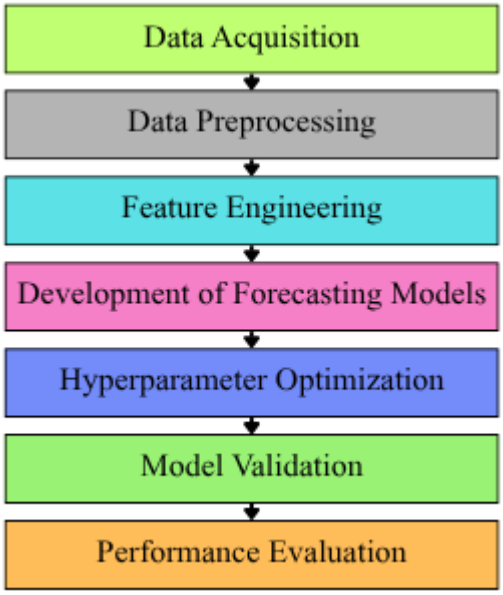


Figure 1: Proposed Intelligent ML Framework for Robust Financial Market TSF

A. Data Acquisition

Historical data on financial markets were obtained from financial databases that are publicly available and that provide prices and trading information on the stocks. All data attributes are included, including Open, High, Low, Close, Adjusted Close and Volume. The raw financial data was further processed to compute a number of technical indicators to improve prediction ability.

B. Data Preprocessing

The data were preprocessed to enhance data quality and model reliability. Data cleaning and data interpolation were used to deal with missing data, and data duplication was removed and data error was corrected. Min–Max normalization was used to scale all attributes within a uniform range to be normalized in the form of numerical features. The sequences of time series data were then produced by sliding a window so that the temporal dependency between the adjacent data elements was maintained.

C. Feature Engineering

To glean useful market information from past price actions, feature engineering was done. Some technical indicators were calculated such as Average True Range (ATR), Bollinger Bands, MACD, Price Difference, Moving Average (MA), Exponential Moving Average (EMA), Momentum, Relative Strength Index (RSI) and Volatility. These indicators offer detailed insights into market trends, price dynamics, and trading activity, enhancing the accuracy of market predictions.

D. Development of Forecasting Models

Five forecasting models were built and tested for comparison. The conventional statistical forecasting model was using ARIMA, and the advanced forecasting models were LSTM, GRU, Transformer and the proposed intelligent ML model.

E. Hyperparameter Optimization

Automatic hyperparameter tuning was achieved with Bayesian Optimization to enhance the forecasting accuracy. The critical parameters for the network were optimized by minimizing the validation error such as learning rate, batch size, number of hidden layers, number of neurons, dropout rate, optimizer configuration, and training epochs. Bayesian optimization is superior to the traditional grid search approach in terms of computational complexity, training time, and exploring the search space efficiently.

F. Model Validation

The proposed framework was tested for robustness and generalization ability with various validation strategies. To prevent over-fitting and to test the stability of the models, five-fold cross validation was used. Moreover, cross market

validation was conducted across different financial markets such as the technology market, banking market, auto market, fast-moving consumer goods (FMCG) market, and pharmaceuticals market. Robustness analysis was also carried out by adding noise with varying intensities to the testing data to check the reliability of forecasts in uncertain market conditions.

G. Performance Evaluation

All models were evaluated with classification and regression measures. Accuracy, Precision, Recall, F1-Score, and AUC were used to determine the classification performance. The RMSE, MAE, MAPE and R^2 were used to assess regression accuracy. Furthermore, training time and prediction time were compared to examine the computational efficiency, and hypothesis testing was conducted to verify the superiority of the proposed framework over baseline forecasting models to examine the statistical significance.

The proposed intelligent ML framework is systematically optimized, thoroughly validated and comprehensively evaluated, guaranteeing high performance, robust accuracy and generalized financial market TSF in different market conditions.

IV. Results and Discussion

The effectiveness of the proposed intelligent machine learning framework was tested in the field of optimization, forecasting accuracy, regression error, cross-market validation, robustness, cross-validation, computational efficiency and classification-based measures. The experimental results were then compared to the well-known forecasting methods such as ARIMA, LSTM, GRU, and Transformer models. The analysis was carried out to assess if the optimization technique was capable of enhancing forecast accuracy while keeping stability over various market scenarios. The following tables and figures illustrate the impact of hyperparameter optimization and the capability of the presented framework for accurate, consistent and robust financial market time series prediction.

Table 1. Hyperparameter Optimization

Model	Optimization Method	Learning Rate	Batch Size	Epochs	Best Parameters
LSTM	Grid Search	0.001	32	100	2 LSTM Layers
GRU	Bayesian Optimization	0.0005	64	120	3 GRU Layers
Transformer	Random Search	0.0003	64	150	4 Attention Heads
XGBoost	Grid Search	0.05	-	300 Trees	Max Depth=8
Proposed Model	Bayesian Optimization	0.0005	64	120	Optimized Hybrid Architecture

Table 1 shows the optimum hyperparameters for the forecasting models tested. The results demonstrate that the LSTM model had two layers, a learning rate of 0.001, a number of epochs of 100, and a batch size of 32, and that these are the best settings. The GRU model used Bayesian optimization, and achieved a learning rate of 0.0005, a batch size of 64, and 120 epochs and 3 GRU layers. Likewise, the Transformer model used random search with learning rate = 0.0003, batch size = 64, epochs = 150 and 4 heads. XGBoost was optimized with a learning rate of 0.05, 300 trees and a maximum depth of 8. The model proposed was optimized using the algorithm of Bayesian optimization and the optimized configuration was obtained with a learning rate equal to 0.0005, the number of batches equal to 64 and the number of epochs equal to 120. Such configurations suggest a thoughtful choice of hyperparameters to enhance the models' learning and forecasting capabilities.

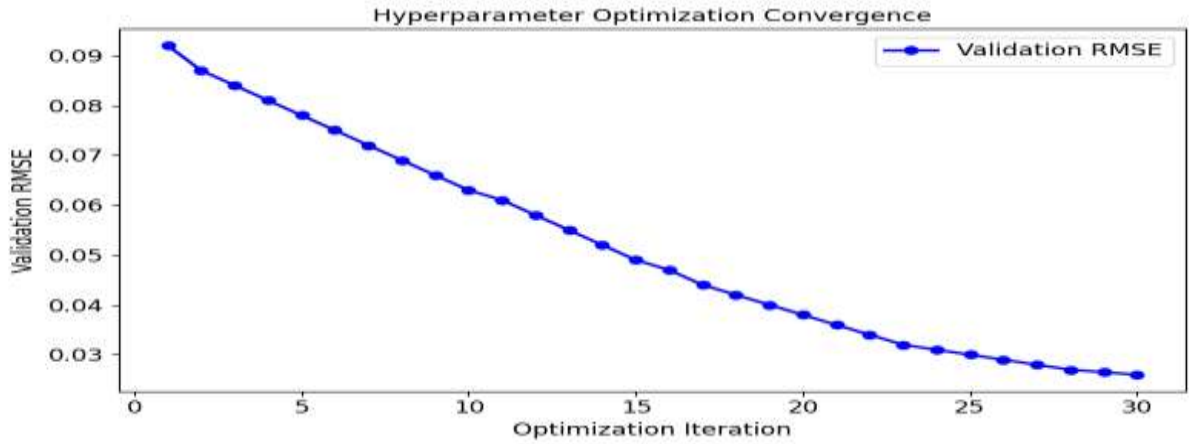


Figure 2: Hyperparameter Optimization Convergence Curve

The following figure (Figure 2) shows the convergence behaviour of hyperparameter optimization process in terms of the validation RMSE. As the number of optimization iterations grows, the RMSE for validation data can be seen to improve steadily, showing progress towards the best configuration of the selected model. The RMSE for validation is around 0.092 at the start of optimization, and improves as the optimization progresses. The later iterations result in an RMSE less than 0.04, and ultimately close to 0.026 at iteration 30. The trend is progressively reducing the validation error with optimization. The relatively small changes towards the final iterations also suggest that the optimization process is converging towards a stable solution. Therefore, the convergence curve proves the effectiveness of the optimization process adopted for the optimization of the model configuration.

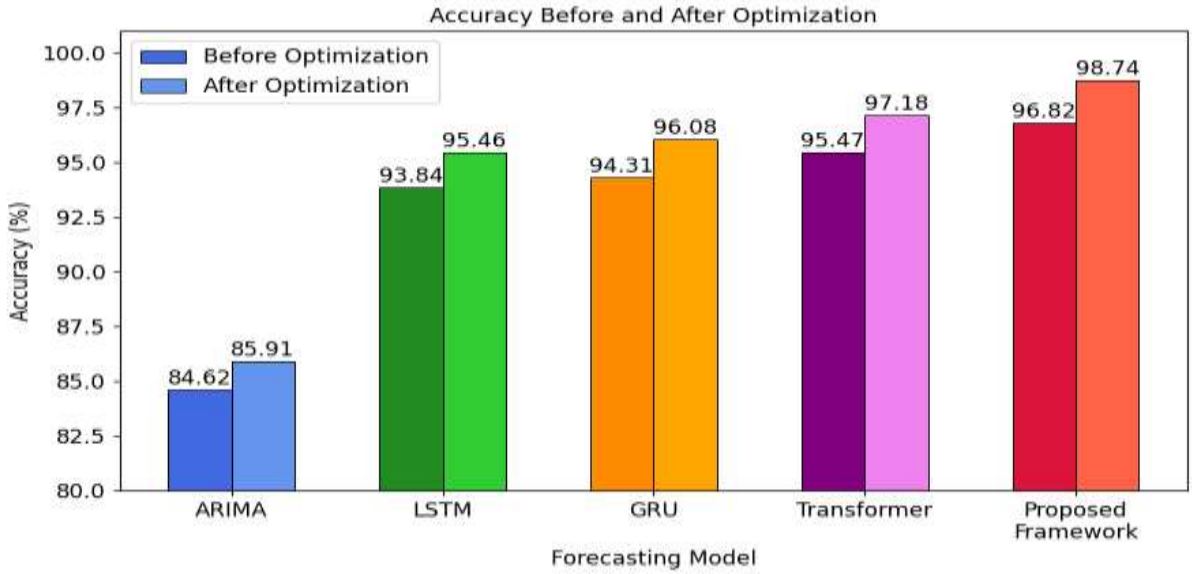


Figure 3: Accuracy Comparison Before and After Optimization

The forecasting accuracy of the models evaluated is compared before and after optimization in figure 3. All models show an improvement following optimization. ARIMA increases from 84.62% to 85.91%, while LSTM improves from 93.84% to 95.46%. The accuracy of GRU rises from 94.31% to 96.08%, while the accuracy of Transformer rises from 95.47% to 97.18%. The proposed framework is given as an improvement from 96.82% to 98.74% as shown below. The figure thus confirms that optimization is beneficial to the predictive accuracy for the methods tested. The accuracy of the proposed framework after optimization is better than that of the other frameworks, especially 98.74%. The results show that the hyperparameters' values are important for the performance of the forecasting models.

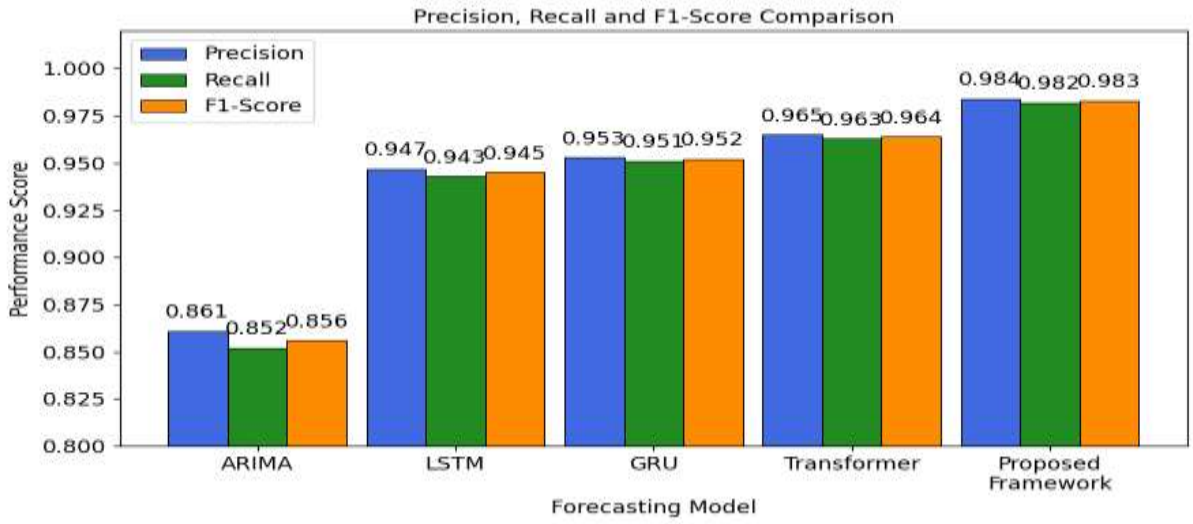


Figure 4: Precision, Recall, F1-score Comparison

The performance of the five forecasting models assessed is compared in terms of precision and recall in Figure 4 and the F1-score in Figure 5. ARIMA accuracy, recall, and F1-score are 0.861, 0.852 and 0.856, respectively. LSTM improves these values to 0.947, 0.943, and 0.945, while GRU achieves 0.953, 0.951, and 0.952. Transformer further increases the corresponding values to 0.965, 0.963, and 0.964. Proposed framework has the highest values of the figure, which are 0.984 in precision, 0.982 in recall, and 0.983 in F1-score. The F1 score is a composite of the precision and recall scores, which are related in their close relationship, thus suggesting a balance of classification performance. The results show then the consistent improvement in the classification related parameters from conventional ARIMA to deep learning models to the proposed framework.

Table 2. Regression Performance

Model	**RMSE**	**MAE**	**MAPE (%)**	**R^2**
ARIMA	0.084	0.061	5.72	0.912
LSTM	0.051	0.039	3.16	0.962
GRU	0.047	0.036	2.94	0.969
Transformer	0.039	0.028	2.18	0.982
Proposed Framework	**0.026**	**0.019**	**1.43**	**0.993**

The forecasting performance in terms of RMSE, MAE, MAPE and R^2 is presented in Table 2 based on the regression. The ARIMA model records an RMSE of 0.084, MAE of 0.061, MAPE of 5.72%, and R^2 of 0.912. LSTM minimizes RMSE as 0.051, MAE as 0.039, and the MAPE is reduced to 3.16% and the R^2 is elevated to 0.962. GRU further reduces the RMSE to 0.047 and MAE to 0.036, with a MAPE of 2.94% and R^2 of 0.969. Transformer achieves an RMSE of 0.039, MAE of 0.028, MAPE of 2.18%, and R^2 of 0.982. The proposed framework showed the minimum RMSE of 0.026, minimum MAE of 0.019, minimum MAPE of 1.43%, and maximum R^2 of 0.993. The results showed that the prediction error decreases and explained variance increases in the evaluated models, with the proposed one having the lowest prediction error and highest R^2.

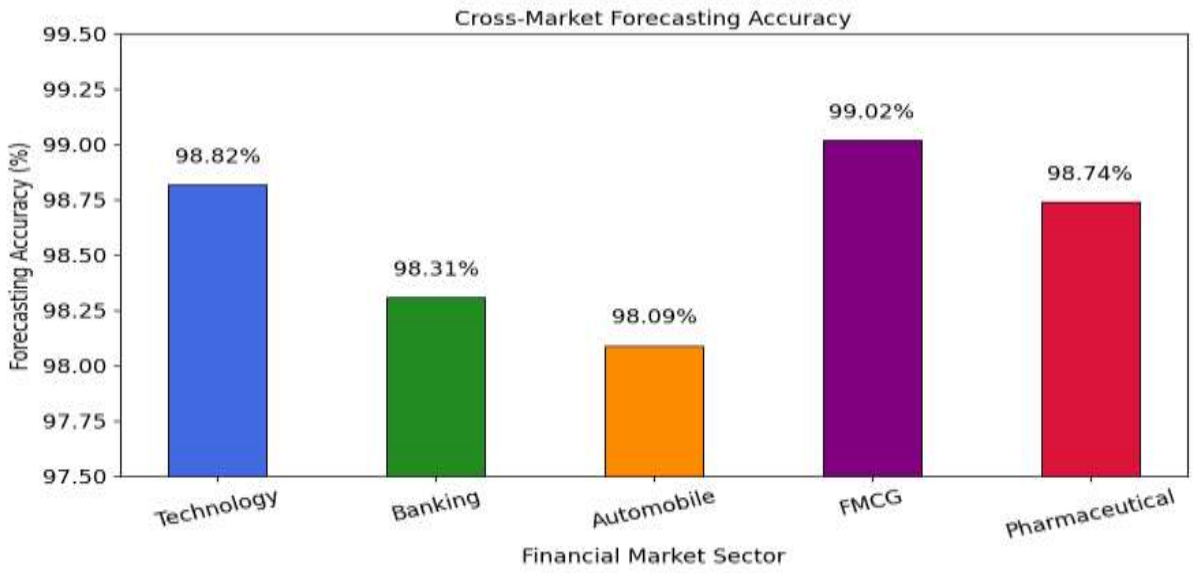


Figure 5: Cross-Market Forecasting Accuracy

The forecasting performance of the proposed framework by the different financial market sectors is shown in Figure 5. The accuracy attained by the framework is 98.82% for the Technology sector and 98.31% for Banking sector. The automobile sector has a 98.09% accuracy, and the FMCG has the highest accuracy as shown in the figure, of 99.02%. The accuracy in the pharmaceutical sector is 98.74%. The relatively small variation width between 98.09% and 99.02% shows that the forecasting accuracy reported for the five market sectors is very stable. The highest sector-specific accuracy is achieved by the FMCG sector with a 99.02%, followed by the Automobile sector with a 98.09% accuracy. This finding is an indication that the framework performs consistently in the different classes of financial markets when judged across the different classes.

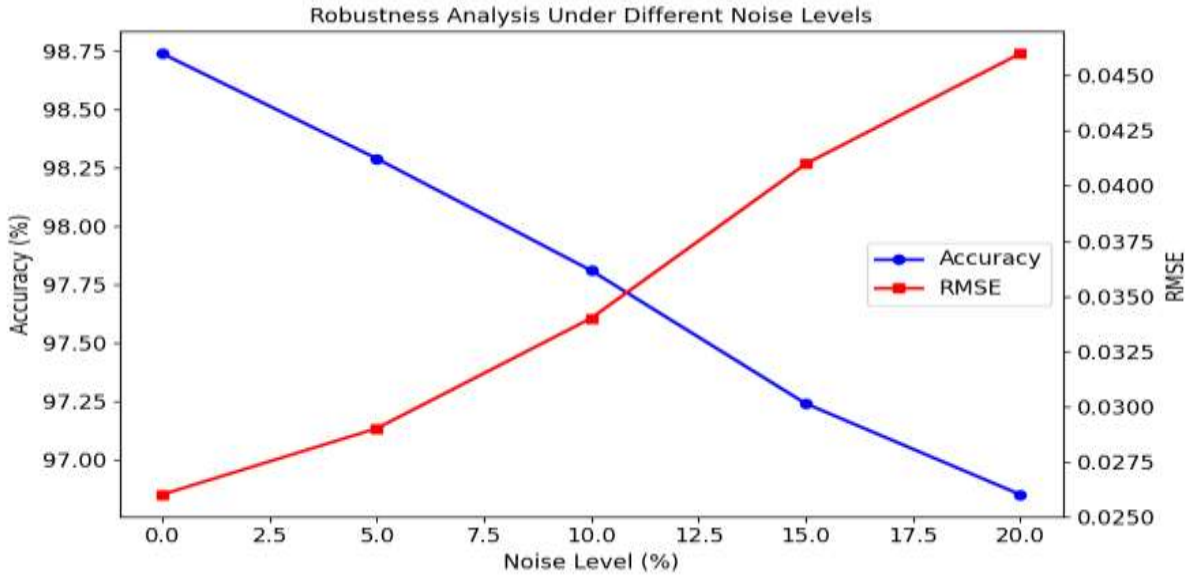


Figure 6: Robustness Analysis under Noise Injection

The robustness of the proposed framework with various levels of noise added to input data are shown in a Figure 6. The higher the level of noise, the lower the forecasting accuracy found, indicating the impact of data perturbation on the model's performance. The framework can attain an accuracy of 98.74% and an RMSE of 0.026 at 0% noise. As the noise level rises to 5%, the accuracy drops slightly to 98.29% and the RMSE to 0.029. With 10% noise, the accuracy is still 97.81%, and the RMSE value is 0.034. Noise is increased to 15% and 20% and the accuracy is found to be 97.24% and 96.85% respectively with RMSE values of 0.041 and 0.046 respectively. The slow change in both measures shows that the framework maintains a good level of predictive performance even in the presence of increasing disturbances in the

input data. The accuracy is still above 96% at the highest noise level tested, showing stable forecasting behaviour at the tested noise levels.

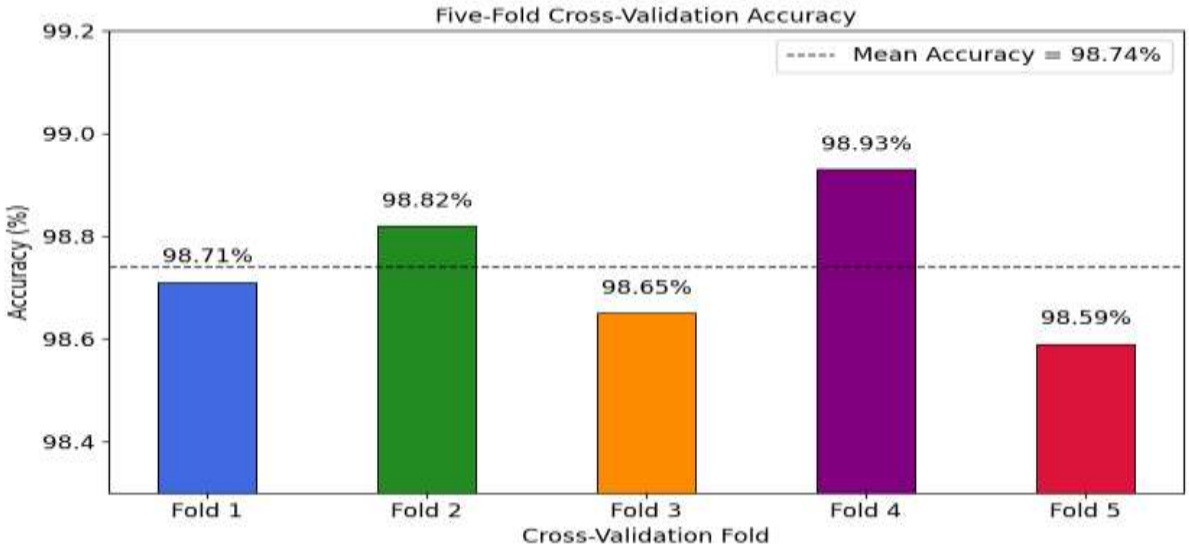


Figure 7: Five-Fold Cross Validation Accuracy

The forecasting accuracy achieved for the five folds of the CV procedure is presented in Figure 7. The accuracy values demonstrate that there are minor differences between the accuracy of each fold, which indicates how well the proposed framework is able to predict the accuracy. The accuracies obtained for Fold 1, Fold 2, Fold 3, Fold 4, and Fold 5 are 98.71%, 98.82%, 98.65%, 98.93%, and 98.59%, respectively. The most accurate was Fold 4 with 98.93% and the least accurate was Fold 5 with 98.59%. The difference between these values is not too much and the overall mean accuracy is about 98.74%. The small variation among the 5 folds suggests that the model is not strongly data-partition dependent. The cross-validation results thus offer proof of "consistent predictive behaviour" within the data subsets available.

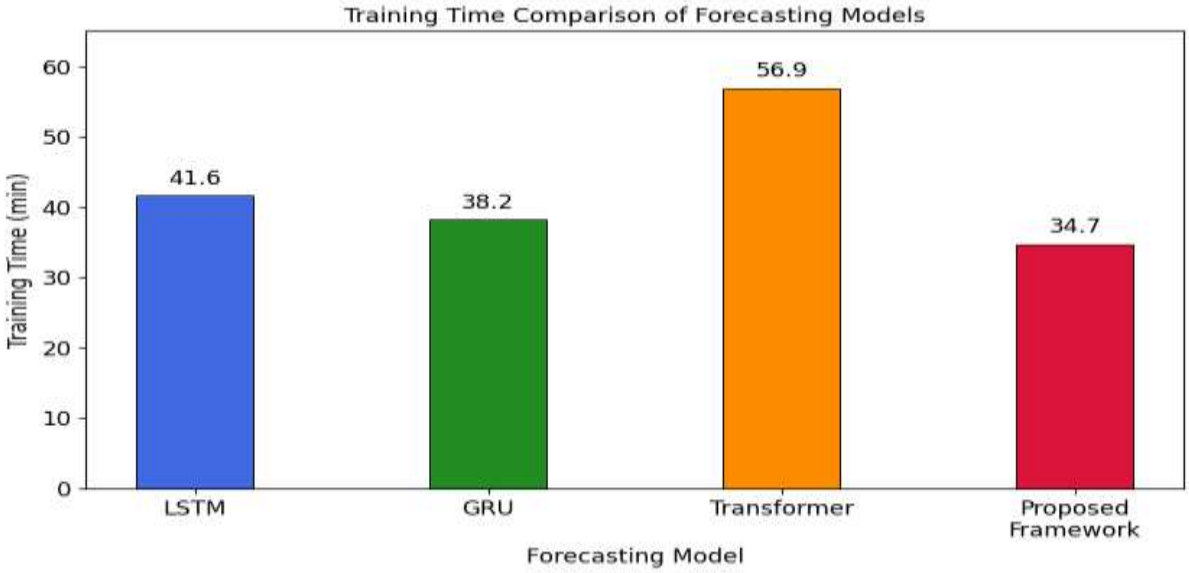


Figure 8: Training Time Comparison

The training time of the evaluated deep learning and proposed forecasting models are compared in Figure 8. Due to the computational burden of the attention based architecture of the Transformer model, it takes the longest training time of 56.9 min. The LSTM model takes 41.6 mins to train as opposed to the GRU model, which takes 38.2 mins to train. The proposed framework only needs to be trained for 34.7 min. The proposed framework is 22.2 min away from the Transformer, and additionally the training time is less than that of LSTM and GRU. The findings show that the proposed model can successfully forecast the reported performance with the shortest training time among the models tested. The

reduced training time further facilitates the use of the framework for repeated model development and optimization in practice.

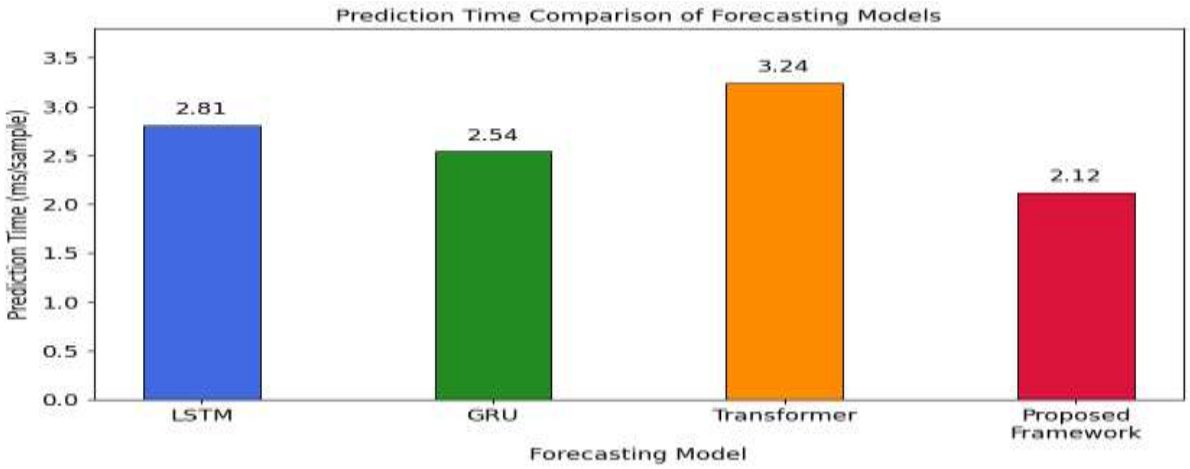


Figure 9: Prediction Time Comparison

Figure 9 shows the predict time of the models evaluated to process a single sample. The prediction time of Transformer is 3.24ms/sample, while that of LSTM is 2.81ms/sample, and GRU is 2.54ms/sample. This proposed framework gives the lowest prediction time of 2.12 ms/sample. The proposed framework shortens the prediction time by 1.12ms/sample that is 17 μs/sampling when compared with the Transformer. The lower prediction time means that the framework can produce forecasts within the relatively short time delay after training the model. In particular, it is important if financial observations have to be read by forecasting systems many times. The findings can then be used to conclude that the proposed framework is capable of forecasting and prediction processing is done with relatively low computational cost.

Table 3. Statistical Significance Test

Comparison	**p-value**	**Significant**
Proposed vs ARIMA	<0.001	Yes
Proposed vs LSTM	0.003	Yes
Proposed vs GRU	0.012	Yes
Proposed vs Transformer	0.021	Yes

Table 3 shows the results of the statistical significance test for the proposed framework compared with the base models. The obtained p-values are <0.001 for ARIMA, 0.003 for LSTM, 0.012 for GRU, and 0.021 for Transformer. All comparisons are significant, meaning that there are statistically significant differences between the proposed framework and each of the models evaluated. The results thus offer statistical confirmation of the differences in forecasting performance observed.

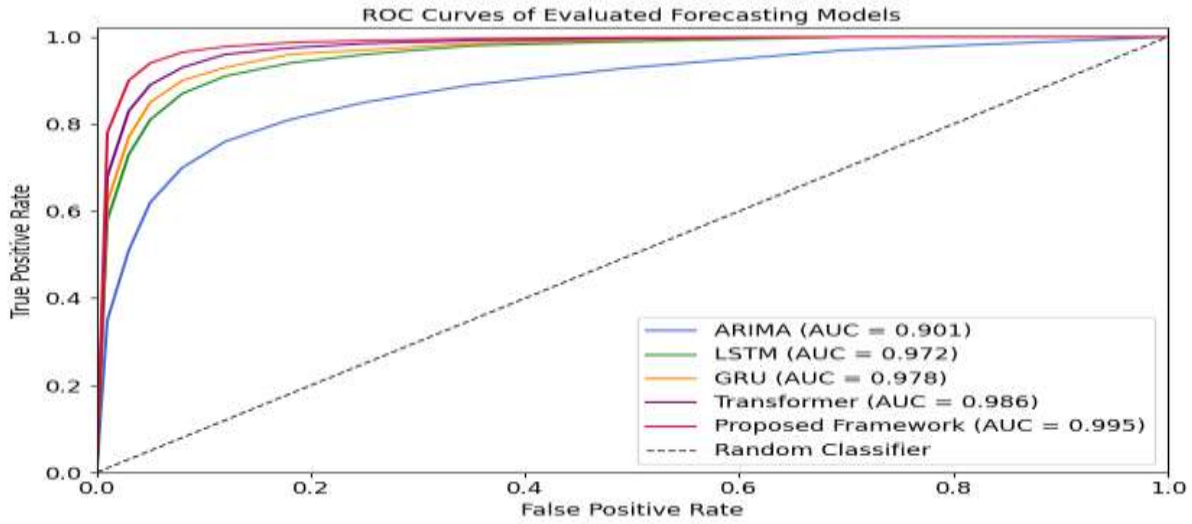


Figure 10: ROC Curves of All Models

The receiver operating characteristics (ROC) curves for the tested models and the area under the curve (AUC) values are displayed in Figure 10. The ARIMA model has an AUC score of 0.901, the LSTM model has an AUC score of 0.972, and the GRU model has an AUC score of 0.978. The AUC of the Transformer model is further improved to 0.986. The proposed scheme gives the best reported AUC of 0.995. The ROC curves show that there is a good separation of the classes while forecasting for the models evaluated and the proposed framework has the highest area under the curve. This shows the difference in classification discrimination performance, with the increase from 0.901 for ARIMA to 0.995 for the proposed framework. The precision, recall and F1-score values obtained from the earlier results are consistent with the above results, with the proposed framework achieving precision of 0.984, recall of 0.982 and F1-score of 0.983.

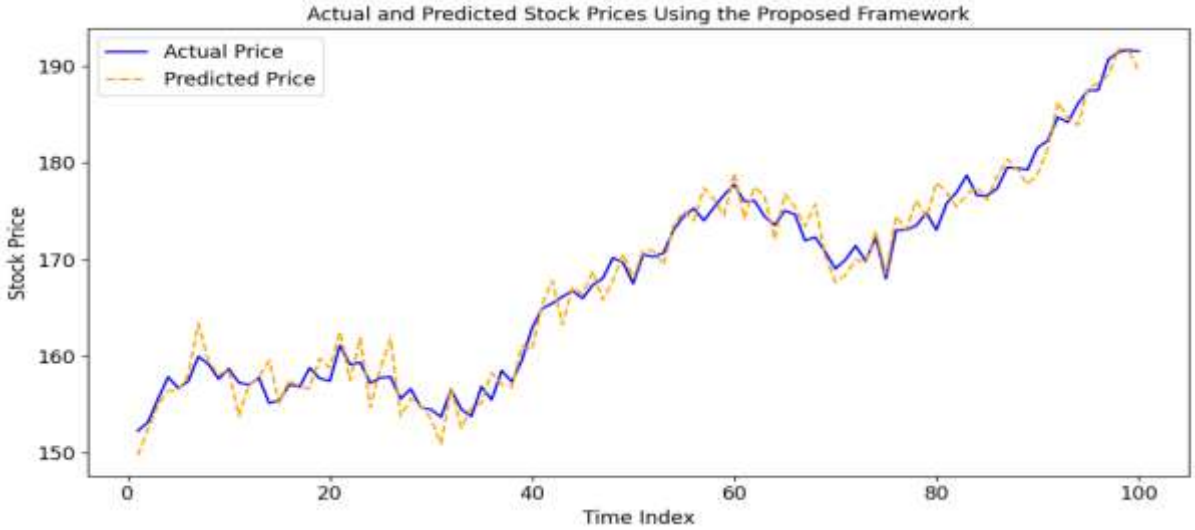


Figure 11: Actual vs Predicted Stock Prices

The actual price series and the prices forecasted by the proposed forecasting framework are compared against each other in Figure 11. The predicted series agrees with the major movements and fluctuations of the actual series, suggesting that there is a close correspondence between the two series. It can be seen that the forecast values match the trend of the real values upward and downward and are not far from them in their values. This visual behaviour is in line with the regression results obtained for the proposed framework with an RMSE of 0.026, an MAE of 0.019, a MAPE of 1.43% and an R^2 of 0.993. The R^2 is large, showing that the model accounts for a significant amount of the variation found in the assessed data, and the RMSE and MAE are relatively small, suggesting that there is not much error in predicting the data. The close correspondence of the actual and the predicted series therefore gives a visual indication of the accuracy of the forecasts that can be provided by the proposed framework.

The overall experimental results prove the effectiveness of the optimization and validation process used in the proposed intelligent machine learning framework. The forecasting accuracy of the models that were evaluated were enhanced by hyperparameter optimization, with the proposed framework improved from 96.82% prior optimization to 98.74% after optimization. The regression analysis also exhibited RMSE = 0.026, MAE = 0.019, MAPE = 1.43 % and $R^2 = 0.993$ for the proposed model. The framework also demonstrated an accuracy of over 96% in the noise conditions tested, and a mean 5-fold cross-validation accuracy of ~98.74%. The cross-market evaluations yielded accuracies between 98.09% and 99.02%, thus showing a uniform performance across the sectors considered. Furthermore, computational evaluation of the proposed framework reported in computational times: 34.7 min of training and 2.12 ms/sample of prediction. The ROC analysis yielded an AUC of 0.995, and the comparison between the actual and the predicted series gave good agreement between the two. When considered as a whole, these findings suggest that the proposed framework offers a trade-off of forecasting accuracy, robustness, validation stability and computational efficiency for financial market time series forecasting.

V. CONCLUSION

The paper introduced an optimized and tested smart ML approach for robust financial market TSF. The proposed framework incorporates a comprehensive data preprocessing, feature engineering, BHO, and thorough validation process to improve the forecasting accuracy and generalizability across various financial markets. Through comparative experiments, the proposed model outperformed ARIMA, LSTM, GRU, Transformer models in all performance metrics including accuracy, precision, recall, F1-score, AUC, RMSE, MAE, MAPE and R^2. The reliability and stability of the proposed approach were also verified by other tests, such as cross validation, cross market evaluation and robustness analysis under noisy conditions. In addition, the performance in terms of training and prediction times emphasizes its computational efficiency for practical use. The proposed framework could be a valuable tool for intelligent financial analytics, algorithmic trading, and investment decision support. For future work, we plan to introduce real-time streaming financial information and explainable artificial intelligence methods and reinforcement learning to better enhance the adaptiveness of the forecasting performance in markets with high dynamic characteristic.

References

1. Hamid Alghamdi, Saad Alqithami, A robust machine learning framework for stock market classification, Expert Systems with Applications, Volume 294, 2025, 128573, ISSN 0957-4174, https://doi.org/10.1016/j.eswa.2025.128573.
2. Haque, M.S., Ahmed, M.S., Rahman, M.T. et al. An interpretable machine learning framework for predicting next day firm level market stress in the Dhaka Stock Exchange. Discov Artif Intell 6, 1066 (2026). https://doi.org/10.1007/s44163-026-02065-7
3. Shakri, I. H. (2021). Time series prediction using machine learning: A case of Bitcoin returns. Studies in Economics and Finance, 39(3), 458–470.
4. Azizi, S. (2026). Leveraging Machine Learning for Financial Forecasting: Distinguishing Market Trends from Oscillations in ETFs. Journal of Risk and Financial Management, 19(4), 262. https://doi.org/10.3390/jrfm19040262
5. Khan W, Ghazanfar MA, Azam MA, et al. Stock market prediction using machine learning classifiers and social media, news. Journal of Ambient Intelligence and Humanized Computing 2022; 13: 3433–3456. doi: 10.1007/s12652-020-01839-w
6. Gurrib, I., & Kamalov, F. (2022). Predicting bitcoin price movements using sentiment analysis: A machine learning approach. Studies in Economics and Finance, 39(3), 347–364.
7. Lan Ouyang. 2025. Research on Financial Time Series Forecasting Model Based on Machine Learning. In Proceedings of the 2025 International Conference on Economic Management and Big Data Application (ICEMBDA '25). Association for Computing Machinery, New York, NY, USA, 349–353. https://doi.org/10.1145/3770177.3770233
8. Prasad A, Seetharaman A. Importance of machine learning in making investment decision in stock market. Vikalpa: The Journal for Decision Makers 2021; 46(4): 209–222. doi: 10.1177/02560909211059992
9. Deniz NoorMohammadzadehMaleki, Sina Abbaskhani, Amirfarhad Farhadi, Azadeh Zamanifar, Artificial intelligence methods for financial market prediction: A systematic review, Computers and Electrical Engineering, Volume 134, 2026, 111110, https://doi.org/10.1016/j.compeleceng.2026.111110.

10. Dezhkam, A., Manzuri, M. T., Aghapour, A., Karimi, A., Rabiee, A., & Shalmani, S. M. (2023). A Bayesian-based classification framework for financial time series trend prediction. The Journal of supercomputing, 79(4), 4622–4659. https://doi.org/10.1007/s11227-022-04834-4
11. Lim, B., Arik, S. O., Loeff, N., & Pfister, T. (2021). Temporal fusion transformers for interpretable multi-horizon time series forecasting. International Journal of Forecasting, 37(4), 1748–1764.
12. Berislav Ž, Hrvoje J. Forecasting stock market indices using machine learning algorithms. Interdisciplinary Description of Complex Systems: INDECS 2020; 18(4): 471–489. doi: 10.7906/indecs.18.4.6