Techno-Science Research Journal V14 (Special Issue) (2026) P 1-9

Content list available at ITC

Techno-Science Research Journal

Journal Homepage: http://techno-srj.itc.edu.kh/

Techno-Science Research Journal

Comparison Analysis of Deep Learning and Tree-Based Homogeneous Ensemble Learning Predictive Models on the KHR/USD Daily Official Exchange Rate

Hourpheng Hort*, Sokkhey Phauk, Pheak Neang, Vesal Khean

Department of Applied Mathematics and Statistics, Institute of Technology of Cambodia, Russian Federation Blvd., P.O. Box 86, Phnom Penh, Cambodia

Received: 30 June 2025; Revised: 18 August 2025 Accepted: 19 August 2025; Available online: 31 July 2026

Abstract: *Exchange rate prediction is a crucial aspect of financial analysis, particularly for countries with dual currency systems such as Cambodia. The Khmer Riel (KHR) is widely exchanged with the United States Dollar (USD), making accurate prediction models essential for businesses and policymakers. This study explores ensemble learning and deep learning approaches to predict the USD/KHR daily exchange rate using historical data collected from the National Bank of Cambodia (NBC). This study evaluates the performance of four advanced machine learning models—Long Short-Term Memory (LSTM), Gated Recurrent Units (GRU), Random Forest (RF), and eXtreme Gradient Boosting (XGBoost)—for predicting the KHR/USD daily exchange rate. These models were rigorously compared against a traditional ARIMA statistical baseline. The analysis utilized standard time-series evaluation metrics, including Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and R-squared (R2). The results demonstrate that all advanced machine learning models significantly outperform the ARIMA baseline, which was found to be unsuitable for this dataset. Among the experimental models, the deep learning architectures, GRU and LSTM, performed best. They achieved an RMSE of 6.36 KHR and 6.79 KHR, respectively, along with the highest R2 scores of 0.963 and 0.959. While the RF and XGBoost models were also highly competitive, they had a crucial limitation: their inability to extrapolate beyond their training data range (4110 KHR), which resulted in lower accuracy than the deep learning models. This research confirms that both deep learning and tree-based ensemble methods are highly effective at capturing the complex, non-linear dynamics of financial time series data.*

Keywords: Khmer Riel; Exchange Rate Prediction; Deep Learning; Ensemble Learning

1. INTRODUCTION[1]

Exchange Rate is defined as the measurement of a value of one currency in terms of another or the domestic price of the foreign currency; it is often referenced against the United State dollar (USD) because it is the most used currency in the international market making it the de-facto benchmark for the values of other currency. The Khmer riel or Cambodia's Riel (KHR) is also measured in reference to the U.S. dollar.

The history of the Cambodian Riel, or Khmer Riel (KHR), began with its initial introduction in 1953, coinciding with Cambodia's independence from French colonization. This first iteration of the currency was abolished in 1979 by the Khmer Rouge regime. Following the overthrow of this genocidal regime, the People's Republic of Kampuchea (PRK) reintroduced the Khmer riels into the economy on March 20th, 1980, through the rebuilt of National Bank of Cambodia (NBC) [1]. Initially, the implied value of the Khmer riels in the 1980s was four to one against the United States Dollar. Following political and economic instability, the Khmer riels has experience series events to hyperinflation depreciated its values to 3744.42 riels per U.S in 1997 [2], [3]. Since 2001 the exchange rate became stable fluctuating around 4000 riels per one dollar [1].

The Cambodia's monetary policy is set by the National Bank of Cambodia (NBC) adopting a de-jure managed float regime. The official exchange rate, quoted as riels per dollars (KHR/USD), governs all external transactions made with/by the central government and state enterprises, and serves as accounting rate by the NBC. This official rate is determined daily with adjustment to remain volatile at only one percent of the market rate [3], [4]. Though international foreign exchange market intervention and carefully supply the Khmer

* Corresponding author: *Hourpheng Hort*

E-mail: hourpheng_hort@gsc.itc.edu.kh

riel according to the country's economic condition and its market, the NBC has continued maintaining a stable exchange rate of the currency pair for more than two decades [5].

Despite having a very stable currency exchange rate, Cambodia is known as the most dollarized country in Asia with the dollarization rate calculated by foreign currency deposits to total deposit (RHS) at slightly more than 90% in August 2024 [4]. Consequently, the U.S. dollar remains the preferred currency for large and wholesale transactions, savings, investments, and international trade in Cambodia. The Khmer Riel, on the other hand, is primarily used for smaller, retail transactions. It's common to observe local vendors, market sellers, small businesses, and farmers pricing their goods in Riels but often preferring or using U.S. dollars for purchases and settling larger bills. This creates a significant daily exchange volume between the two currencies, particularly in areas of active trade here in Phnom Penh and across the country.

Given the current stability of the KHR/USD exchange rate, it remains vital to thoroughly study the factors influencing the Khmer Riel's value against the U.S. dollar. Furthermore, as the National Bank of Cambodia continues to implement monetary policies aimed at de-dollarizing the Cambodian economy, this research offers a crucial tool to inform strategies and assess the impact of such efforts for those guiding economic policy.

Existing research on the KHR/USD exchange rate has largely relied on statistical models such as Auto-Regressive Integrated Moving Average (ARIMA) and similar models, to explore its dynamics and internal determinants. While these approaches have been adequate to some extent, there's a clear need for further studies employing more sophisticated techniques like advanced machine learning methods. Such explorations could significantly enhance prediction accuracy and provide a more nuanced understanding of the exchange rate's behavior.

This paper aims to address this gap by exploring advanced machine learning predictive models on the KHR/USD daily official exchange rate. Specifically, it will compare the performance of less computationally intensive tree-based ensemble methods, such as XGBoost and Random Forest, with more complex models like Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRU). The outcomes of this research could offer significant advancements in forecasting capabilities and a more comprehensive analysis of the KHR/USD daily official exchange rate behavior, potentially benefiting financial institutions and policymakers. Therefore, this research makes a specific contribution by being the first to apply a rigorous comparative analysis of tuned tree-based ensemble and deep learning models on the KHR/USD daily exchange rate, utilizing a robust walk-forward validation and hyperparameter optimization strategy to ensure practical relevance.

This study is organized into four distinct chapters. The first chapter serves as an introduction, providing an overview of the research topic. Chapters two, three, and four delineate related works, the research methodology utilized and present the empirical findings, respectively. Lastly, the concluding chapter succinctly summarizes the principal findings and provides insights derived from the study.

2. RELATED WORKS

Initial research on the KHR/USD exchange rate [6] [7] have utilized econometric models like Markov-Switching Autoregressive (MS-AR) and ARIMAX to identify important dynamics, such as distinct volatility regimes and long-term relationships with monetary aggregates. While insightful, these statistical models often struggle to capture the complex non-linear patterns inherent in financial time series, a limitation that has motivated the adoption of machine learning.

Statistical models, often called conventional methods by data scientists, rely on the assumption of stationarity (constant mean and variance) in stochastic processes. However, these models often fail to capture the inherent complexity, non-linear relationship present in the time series [8]. In their research on EUR/USD monthly exchange rate forecasting, a study by [9] utilized an ensemble learning approach with CART ensembles and a bagging predictive model, incorporating macroeconomic indicators like unemployment, inflation, and interest rates. The CART ensembles were the most effective model, with an explanatory power of up to 98.8%, an RMSE of 0.017, and a MAPE of 1%. This study, however, lacked a comparison with other models.

To address this, other research has included comparative analyses. For instance, studies such as [10], [11], and [12] used ARIMA as a baseline to compare against more sophisticated models. The findings revealed that SVM, RNN [10], LSTM [11], and a hybrid ARIMA-LSTM model [12] consistently outperformed the traditional ARIMA model.

Further comparisons have been made among deep learning models themselves. In a study on daily exchange rates for NPR against USD, GBP, and EUR, the authors in [13] found that LSTM performed better than Multi-Layer Perceptron (MLP), Simple Recurrent Neural Network (SRNN), and Gated Recurrent Unit (GRU) based on MAE. Specifically, LSTM had the lowest MAE for USD/NPR (0.013) and EUR/NPR (0.042), while GRU was the top performer for GBP/NPR with an MAE of 0.0177.

The performance of LSTM and GRU has been a recurring theme in other studies as well. Research by [14], [15], and [16] compared these models for predicting 10- and 30-minute interval exchange rates [14], daily exchange rates [15], and cryptocurrencies [16]. Specifically, [14] found that a GRU-LSTM hybrid model surpassed both individual LSTM and GRU models. In their study on daily EUR/USD exchange rate prediction, [15] found that a GRU model with an 80/10/10 training/validation/test data split was the best performer, achieving an RMSE of 0.054, a MAPE of 0.037, and an R-squared of 0.97. For cryptocurrency forecasting, [16] found that GRU consistently outperformed LSTM, demonstrating superior performance across various pairs. For BTC/USD, GRU had an MAE of 0.0325 versus LSTM's 0.0674. Similarly, for ETH/USD and LTC/USD, GRU achieved lower MAE values of 0.0305 and 0.0170, respectively. The study attributed GRU's superior performance to better generalization and a faster convergence rate.

Despite these advancements, there remains a significant gap in the literature regarding the direct comparison of optimized tree-based ensembles (Random Forest, XGBoost) against deep learning models (LSTM, GRU), particularly for the KHR/USD daily official exchange rate. This study aims to fill this gap.

3. METHODOLOGY

3.1. Dataset

Data for this research were collected from the National Bank of Cambodia (NBC) website. Specifically, the target datasets were utilized: "Daily Exchange Rate (Riel/U.S. $)" [17], ranging from 1st January, 2009 to 31st December, 2024 accounting for 5844 days. With datetime index corresponds to a specific day of the month of a specific year, the dataset has 5 columns:

a. Market Exchange Rate - Purchase
b. Market Exchange Rate - Sale
c. Market Exchange Rate - Midpoint
d. Official Exchange Rate
e. Spread Market (SM)

The official exchange rate is chosen to be the target variable to be put in the predictive model consisting of only endogenous variables derived from the target. A comprehensive summary of this statistical analysis is presented in **Table 1**.

Table 1: Summary Statistics of KHR/USD daily exchange rate.

	KHR/USD daily Exchange Rate
Mean	4072.63 KHR
Median	4079.50 KHR
Maximum	4241 KHR
Minimum	3980 KHR
Std. Dev.	52.015 KHR
Skewness	0.96
Kurtosis	1.20
Observations	5844 days

3.2. Feature Engineering and Data Preprocessing

3.2.1. Feature Engineering

The dataset for the exchange rate was clean, with no missing values or duplicates identified. Furthermore, the Official Exchange Rate Midpoint was used as the primary raw time series for our target variable (KHR/USD). As highlighted in [18], raw time series data is often a crucial component of the feature vector, particularly in RNN-based models for financial time series forecasting. To incorporate elements similar to statistical models like ARIMA along with some other meaningful statistics, we extracted the following additional endogenous features directly from the target time series, where **Table 2** provides a comprehensive overview of engineered features and their derivations.

a. Lag Features (1 to 30)
b. Rolling Means (2 to 30)
c. Rolling Standard Deviation (2 to 30)
d. Exponential Moving Average (2 to 30)
e. Day of week
f. Month of the year

Following the generation of endogenous input features, we had a total of 136 input features. Due to the inclusion of lagged components up to 30-degree in our feature set, we anticipated a reduction in the number of usable observation points for model training to the period of 1st January 2011 to 31st December 2024, resulting in a current dataset of 5114 daily observations.

Table 2: Engineered Features and Their Derivations.

Feature Category	Specific Feature	Derivation
Temporal Features	Day of the Week Month of Year	Extracted from date index.
Lags	Lagged Values (1 to 30 days)	Applied to the target.
Moving Averages	Moving Averages (2 to 30 days)	Applied to the target.
Rolling std.	Moving Averages (2 to 30 days)	Applied to the target.
Exponential Moving Averages	Moving Averages (2 to 30 days)	Applied to the target.

3.2.2. Data Preprocessing

- Data Scaling

Applying scaling to input features is crucial before training deep learning models. This step ensures better generalization, avoids bias, promotes faster convergence, and is particularly important in preventing vanishing gradients during backpropagation. Following the guidelines in [19] for optimal input transformation—namely centering the average near zero, ensuring similar covariances, and ideally decorrelating inputs—we opted for normalization. Specifically, for our recurrent neural network (RNN) and related models, normalization to the range of (-1, 1) is generally favored over (0, 1).

- Train-Test Split

For time series analysis, random data splitting is inappropriate due to the inherent temporal dependencies within the data. Consequently, this research employs a chronological split strategy.
The dataset will first be divided into three distinct sets:

a. **Training Set:** The initial **3287** daily obs. (9 years).
b. **Validation Set:** The subsequent **730** daily obs. (2 years).
c. **Test Set:** The remaining **1097** daily obs. (3 years).

3.3. Statistical Model as the baseline model

3.3.1. Autoregressive Integrated Moving Average with (ARIMA)

The AutoRegressive Integrated Moving Average (ARIMA) model is a fundamental statistical model used for time series forecasting. It combines an Auto-Regressive (AR) component, a Moving Average (MA) component, and an Integrated (I) component that handles non-stationarity through differencing. In this research, ARIMA will serve as the *baseline model* to provide a robust comparison, allowing us to assess the performance improvements offered by more advanced and complex predictive models, such as tree-based ensemble learning and deep learning models.

The ARIMA modeling process in this research is implemented using the `auto_arima` function from the *`pmdarima`* library in Python. This algorithm automates the identification of optimal non-seasonal (p, d, q) and seasonal (P, D, Q) orders for the AutoRegressive (AR), Integrated (I), and Moving Average (MA) components, respectively. The model selection is systematically conducted by evaluating various specifications based on the Bayesian Information Criterion (BIC)**,** as specified.

3.4. Tree-based Homogenous Ensemble Learning Models

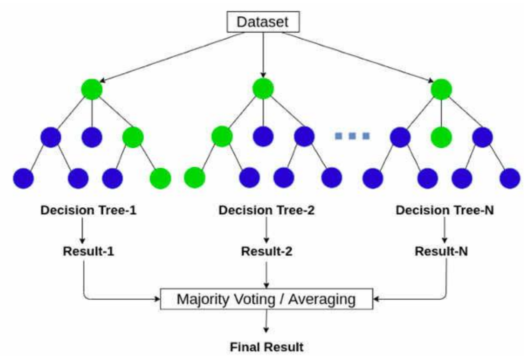
Ensemble learning is a machine learning paradigm that combines the predictions of multiple individual models, often referred to as 'base learners,' to produce a more robust and accurate output [20]. Diversity among these base learners is typically achieved through two primary strategies. Homogenous ensembles utilize multiple instances of the same base learning algorithm. Diversity is often introduced by training on different subsets of the data (as seen in bagging methods like Random Forest) or by iteratively re-weighting data points to focus on where larger prediction errors were made (as in boosting methods like XGBoost). In contrast, heterogeneous ensembles combine different types of base learners which are typically trained on the same dataset. The strength of this approach lies in leveraging the complementary strengths and weaknesses of diverse model

types. In this study, our exploration of ensemble learning will exclusively focus on homogeneous ensembles that utilize decision trees as their base learners, specifically employing bagging and boosting ensemble methods.

3.4.1. Random Forest

Random Forest is an exemplary bagging-based ensemble method that utilizes decision trees as its base learners. These individual trees are generated in parallel with no inter-tree data dependency, meaning each tree is built independently on a bootstrapped subset of the training data. For time series forecasting, implemented as a regression tasks, the final prediction from the ensemble is obtained by averaging the individual predictions of all constituent decision trees.

Fig. 1 Structure of the random forest regressor algorithm. Source: [21].



3.4.2. Extreme Gradient Boosting

XGBoost (eXtreme Gradient Boosting) is a highly optimized and widely used boosting-based ensemble method. Unlike bagging, XGBoost builds its decision tree base learners sequentially, with each new tree attempting to correct

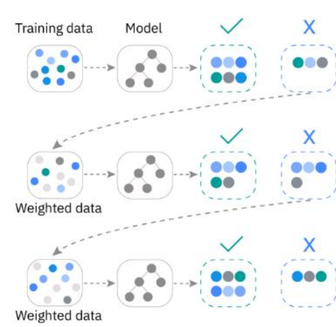


Fig. 2 Structure of the XGboost algorithm. Source: [22].

the residuals (errors) of the predictions made by the preceding trees. This iterative process focuses on instances where the model previously performed poorly, thereby incrementally improving the overall accuracy. For time series forecasting, implemented as a regression tasks, the final prediction is obtained from taking a weighted sum of the predictions from all individual decision trees in the ensemble.

3.5. Deep Learning Models

A semantic literature review by on deep learning applications in financial time series forecasting (2005-2019) highlighted the prevalence of Recurrent Neural Networks (RNNs), which were utilized in 51% of the surveyed studies. Multilayer Perceptrons (MLPs) and Convolutional Neural Networks (CNNs) were employed in 20.8% and 14.1% of the reviewed literature, respectively. Within the RNN category, Long Short-Term Memory (LSTM) networks emerged as the most popular architecture (60.4%), followed by Vanilla RNNs (29.7%) and Gated Recurrent Units (GRUs) (9.89%). Given the sequential nature of time series data, Recurrent Neural Networks (RNNs) are a natural choice for analysis, as they can leverage internal memory to process incoming inputs and capture dependencies over longer time horizons [18]. While Long Short-Term Memory (LSTM) networks have been the prevalent RNN architecture in financial time series forecasting research, Gated Recurrent Units (GRUs), which represent a more streamlined version of LSTMs, offer an intriguing alternative. Interestingly, Gated Recurrent Units (GRUs), despite their release in 2014 (later than LSTM's 1998 introduction), were likely underrepresented in the semantic review by [18]. They have shown promising results, often outperforming LSTMs in tasks characterized by lower computational costs, shorter sequence lengths, and limited datasets.

3.5.1. Long Short-Term Memory (LSTM)

Long Short-Term Memory (LSTM) networks are a specialized type of RNN engineered to effectively handle both short-term and long-term dependencies in sequential data [23]. They achieve this through the introduction of memory cells, which can maintain information over extended time intervals. Crucially, the flow of information into and out

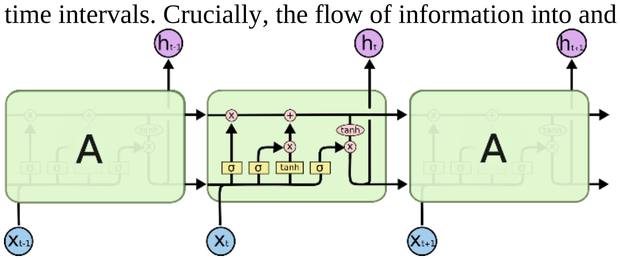


Fig. 3 The Repeating module in an LSTM contains four interacting layers. Source: [23].

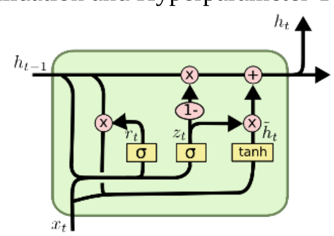
of these memory cells (c) is carefully managed by three key gates: the input gate (i), the forget gate (f), and the output gate (o). These gates selectively regulate the information, allowing the network to learn and retain relevant values across arbitrary time spans [23].

3.5.2. Gated Recurrent Units (GRU)

Gated Recurrent Units (GRU) emerging as a potent yet alternative to the LSTM is the Gated Recurrent Unit, or GRU, prioritizes the hidden state (h) over a distinct cell state (C) and streamlines the gating mechanism by combining the input (i) and forget gates (f) into an "update gate" (z). This gate determines how much information of the past hidden state should be retained and pass on to the future timestep. Complementing this, the reset gate (r), determines how much of the past information of the past hidden state need to reset or discard.

Fig. 4 GRU architecture. Source: [23].

3.6. Cross-Validation and Hyperparameter Tuning



In this study, a robust methodology was employed to ensure model reliability and optimal performance. For time series data, Walk-Forward cross-validation was used instead of traditional random data splits to preserve the temporal order of observations. This method trains a model on an initial segment of historical data and then predicts on a subsequent, fixed-size period. The training window then "walks forward" to include this newly tested data, and the process is repeated. This approach realistically simulates real-world predicting and is crucial for detecting changes in data patterns over time, known as concept drift.

To optimize model performance, hyperparameter tuning was conducted using the Optuna framework. Unlike grid or random search, Optuna uses a dynamic, adaptive approach to efficiently explore the hyperparameter space. This process involves defining an objective function, which for this study was the Root Mean Squared Error (RMSE), and then systematically searching for the combination of hyperparameters that minimizes this metric.

The tuning process was applied to all models. The specific hyperparameters and their optimal values for the ensemble learning models (Random Forest and XGBoost) are detailed in **Table 3**, while the values for the deep learning models (LSTM and GRU) are in **Table 4**. This comprehensive tuning and validation process was essential for ensuring the predictive models were both accurate and robust.

3.7. Evaluation Metrics

Common evaluation metrics for non-classification prediction tasks, relevant to the predicting of the KHR/USD exchange rate and broad money, include Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), Root Mean Squared Error (RMSE), and the coefficient of determination (R^2) [8]. For this paper, we explore all metrics; however, the R^2 is used to decide the best fit model. The equations (Eq.12-Eq.15) illustrate how all evaluation metrics are calculated.

$$MAE = \frac{1}{n}\Sigma_{i=1}^{n}|y_i - \hat{y}_i| \qquad \text{(Eq.12)}$$

$$MAPE = \frac{1}{n}\Sigma_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \times 100\% \qquad \text{(Eq.13)}$$

$$RMSE = \sqrt{\frac{1}{n}\Sigma_{i=1}^{n}(y_i - \hat{y}_i)^2} \qquad \text{(Eq.14)}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y}_i)^2} \qquad \text{(Eq.15)}$$

Where,

n is the number of observations
y_i is the actual value at i^{th} data point
$\hat{y}_i$ is the predicted value at i^{th} data point
$\bar{y}_i$ is the mean of actual values

4. RESULTS AND DISCUSSION

4.1. Baseline Model (ARIMA)

In statistical time series analysis, achieving stationarity, often confirmed by unit root tests such as the Augmented Dickey-Fuller (ADF) or Kwiatkowski-Phillips-Schmidt-Shin (KPSS) tests, is a prerequisite for many modeling approaches. The Auto-Regressive Integrated Moving Average (ARIMA) model, recognized for its simplicity, computational efficiency, and interpretability rooted in statistical foundations, is widely adopted as a baseline or "benchmark" model in time series studies.

In this study, pmdarima was utilized to automate the selection of ARIMA model components, allowing for a maximum order of 7 for both Auto-Regressive and Moving Average parts, with the degree of differencing automatically determined to achieve stationarity. Applied to the feature-engineered time series for consistent comparison with deep learning models, the optimized ARIMA (7, 1, 0) model, achieving an AIC 14960, demonstrated significant limitations. Its evaluation metrics (RMSE: 33.65, and R2: -0.033), particularly the highly negative R2, unequivocally indicated that this linear model was not adequately fitted for the prediction task and failed to generalize effectively. This poor performance was further supported by residual analysis, which suggested the underlying data was essentially white noise, confirming the unsuitability of a linear approach as shown in Fig. 5.

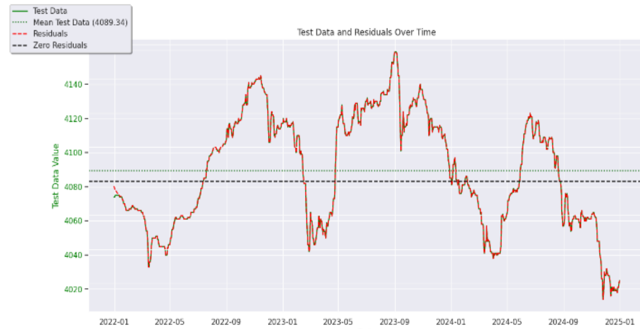


Fig. 5 Residuals obtained from ARIMA

4.2. Hyperparameters Tuning for all Models

To represent homogeneous ensemble methods, Random Forest (RF) and Extreme Gradient Boosting (XGB) algorithms were employed, embodying the bagging and boosting paradigms, respectively. A direct comparison of their performance considering computational cost proved challenging. The tuned hyperparameters for both models, are detailed in **Table 3**.

Table 3: Optimal Hyperparameters of Tree-Based Ensemble Learning Models.

Model	Hyperparameter	Search Space	Optimal Value
Random Forest	*n estimators*	50 to 500	96
	max depth	5 to 25	15
	min sample split	2 to 20	19
	min sample leaf	1 to 15	8
	max features	['sqrt', 'log2', 0.6, 0.8, 1.0]	0.8
XGBoost	*n estimators*	50 to 1000	975
	max depth	3 to 15	7
	learning rate	Log scale 0.0001 to 0.01	0.046
	subsample	0.5 to 1.0 (step 0.1)	0.8
	colsample by tree	0.5 to 1.0 (step 0.1)	1.0
	gamma	0.0 to 0.5 (step 0.1)	0.1
	reg alpha	Log scale 1e-8 to 10	0.005
	reg lambda	Log scale 1e-8 to 10	4.15
	min child weight	1 to 10	3

For both the deep learning LSTM and GRU models, training incorporated Huber loss and early stopping with a 20-epoch patience based on validation loss. Hyperparameter optimization was performed using Optuna, with each model undergoing 30 trials to refine the parameters and the configurations and performance of the best-tuned models are also summarized in **Table 4**. A comparison of model complexity revealed that the tuned LSTM had 805,384 trainable and optimizer parameters, whereas the tuned GRU had 306,825 parameters, representing approximately 2.6 times fewer parameters.

Table 4: Hyperparameter Tuning Ranges, Best Model Configurations for GRU and LSTM Models.

Hyperparameter	Search Space	GRU	LSTM
Batch size	[32, 64]	32	64
RNN layers	1 to 3 (step 1)	1	1
RNN neurons per layer	[64, 128, 256]	128	256
Dropout rate	0.1 to 0.3 (step 0.1)	0.1	0.3
RNN layers activation function	["relu", "tanh"]	Relu	Tanh
Output layer activation function	["linear", "relu"]	Linear	Linear
Optimizer	["Adam", "RMSprop"]	Adam	RMSprop
Learning rate	Log scale 1e-4 to 1e-2	47e-4	6e-4
Huber delta	1.0 to 1.2 (step 0.1)	1.2	1.2

4.3. Overall Results

The evaluation of predictive models reveals a clear performance hierarchy, with advanced machine learning models significantly outperforming the traditional statistical approach. The GRU (Gated Recurrent Unit) model emerged as the top performer, excelling at capturing overall data variance and avoiding major prediction failures. This is evidenced by its highest R2 score of 0.963 and lowest RMSE (Root Mean Square Error) of 6.36 KHR. Following closely behind was a deep learning model counterpart the LSTM (Long Short-Term Memory), which, despite a higher computational cost, achieved an R2 score of 0.959 and an RMSE of 6.79 KHR. This superior performance confirms that these complex deep learning models have an edge over the ensemble model.

Due to their fundamental limitation of not being able to extrapolate beyond their training data (interpolation), the RF (Random Forest) and XGBoost (eXtreme Gradient Boosting) models did not achieve the highest scores but only fell behind by a slight margin, reaching impressive R2 scores of 0.92 and 0.90, respectively.

In contrast, the traditional ARIMA (Autoregressive Integrated Moving Average) model performed the worst by a substantial margin, indicated by a negative R2 score (-0.033) and the highest RMSE and MAE (Mean Absolute Error) values. This poor performance confirms its unsuitability for this particular dataset, which likely contains complex relationships that a linear, statistical model cannot adequately capture. Overall, the results underscore the effectiveness of modern machine learning models for this predicting task when compared to the ARIMA statistical baseline, with the GRU model providing the most accurate and efficient predictions.

Table 5: Evaluation Metrics Comparison across all Models.

	ARIMA	RF	XGBoost	LSTM	GRU
RMSE	33.65	9.72	10.51	6.79	6.36
R2	-0.033	0.915	0.901	0.959	0.963
MAE	29.20	5.46	6.07	4.89	4.65
MAPE	72%	0.13%	0.15%	0.12%	0.11%

e

Fig. 6, the machine learning models (RF, XGBoost, LSTM, and GRU) exhibit a clear hierarchy of performance in predicting the KHR/USD exchange rate. The deep learning models (GRU and LSTM) demonstrate superior accuracy, with their predictions (represented by the green and purple dotted lines) closely following the actual values (the solid blue line) and effectively capturing the major peaks and troughs of the time series.

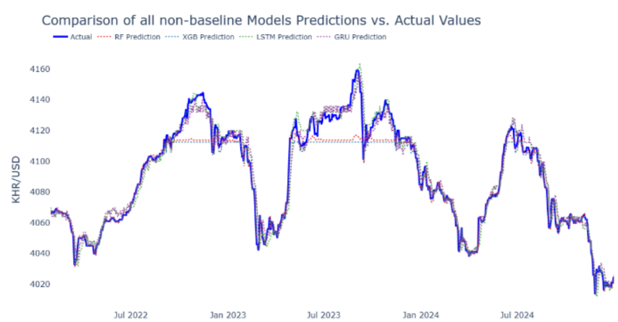


Fig. 6 Actual vs Prediction of all models

The ensemble models (RF and XGBoost), represented by the red and blue dotted lines respectively, also perform well but show slightly larger deviations during periods of high volatility. They are notably bounded by a value of 4110 KHR, which is a definite result of their fundamental limitations. This visual evidence strongly supports the quantitative findings that the deep learning models are the most effective for this forecasting task, with the GRU model being more efficient due to its lower computational cost compared to the LSTM.

5. CONCLUSIONS

This study successfully applied and evaluated four advanced machine learning models—Random Forest (RF), XGBoost, Long Short-Term Memory (LSTM), and Gated Recurrent Unit (GRU)—against a traditional ARIMA statistical baseline for forecasting the KHR/USD exchange rate. The results unequivocally demonstrate the superiority of the machine learning models, with ARIMA performing the worst across all metrics, highlighted by a negative R2 score of -0.033. Among the machine learning models, the deep learning models were the top performers specifically GRU and LSTM, are the most effective for predicting the KHR/USD exchange rate, significantly outperforming traditional ARIMA and tree-based models. The key takeaway is that the deep learning models (GRU and LSTM) achieved the lowest test RMSEs of 6.36 KHR and 6.79 KHR respectively, and the highest R2 among all models of 0.963 and 0.959, indicating their exceptional ability to generalize to unseen data. This superior performance is attributed to their capacity to capture complex, non-linear dynamics and to extrapolate beyond the range of historical training data, a fundamental limitation observed in the Random Forest and XGBoost models which could not predict values past 4110 KHR. The study concludes that the GRU model is the best overall performer due to its high accuracy and computational efficiency.

Despite the successful identification of high-performing models, this study faced several key limitations that impact the generalizability of our findings. A significant oversight was the omission of a thorough residual analysis, which prevented us from identifying systematic errors or uncaptured patterns within our models. Furthermore, the inherent complexity of forecasting the KHR/USD exchange rate, influenced by Cambodia's managed float regime and unmodeled macroeconomic factors, posed a substantial challenge. Methodologically, the combination of Walk-Forward cross-validation with hyperparameter tuning was computationally intensive, requiring significant time and resources to optimize model performance.

Future research should build on these findings by addressing the limitations of this study. First, a residual analysis is essential to enhance model reliability by uncovering and correcting systematic errors. Second, a transition from a univariate to a multivariate framework is recommended to improve predictive accuracy by incorporating external factors such as macroeconomic indicators and geopolitical events. Finally, exploring more advanced deep learning architectures, such as Transformer networks for capturing long-range dependencies or Bayesian Neural Networks for quantifying prediction uncertainty, could provide more robust and precise forecasting capabilities.

ACKNOWLEDGMENTS

My sincere appreciation and deepest gratitude extend to Dr. Sokkhey PHUAK, Dr. Pheak NEANG, and Mr. Vesal KHEAN for their invaluable support throughout this research. Their profound insights, exceptional expertise, and unwavering encouragement with great patience, were instrumental in not only completing this work, but also in fostering a comprehensive understanding of the subject matter. Their supervision has been a great source of direction, and I am honored to have had the opportunity to contribute to this project under their supervision.

REFERENCES

[1] H. Okuda and S. Chea, *Cambodian Dollarization: Its Policy Implications for LDCs' Financial Development*, 1st ed. London: Routledge, 2023. doi: 10.4324/9781003391418.

[2] C. Hay, "Dollarization and macroeconomic performance in Cambodia since the first 1993 general election: a historical perspective," *IJFBS*, vol. 10, no. 2, pp. 27–46, Jun. 2021, doi: 10.20525/ijfbs.v10i2.1179.

[3] H. Chanthol, "Money Demand and Inflation in a Highly Dollarized Economy: Fighting Inflation in Cambodia".

[4] International Monetary Fund. Asia and Pacific Dept, "Cambodia," *IMF Staff Country Reports*, vol. 2025, no. 022, p. 1, Jan. 2025, doi: 10.5089/9798400299711.002.

[5] M. Sean, P. Pastpipatkul, and P. Boonyakunakorn, "Money Supply, Inflation and Exchange Rate Movement: The Case of Cambodia by Bayesian VAR Approach," *JOMEINO*, pp. 63–81, Jan. 2019, doi: 10.31039/jomeino.2019.3.1.5.

[6] S. Lim, "Modelling Cambodia's Foreign Exchange Rate Dynamics: A Markov-Switching Autoregressive Model," *IJEFI*, vol. 15, no. 1, pp. 330–336, Dec. 2024, doi: 10.32479/ijefi.17618.

[7] S. Lim, "Monetary Policy and Foreign Exchange Fluctuations in Cambodia," *14/09/2023*, vol. 2, no. 9, pp. 30–37.

[8] S. Ghahremani and U. T. Nguyen, "Prediction of Foreign Currency Exchange Rates Using an Attention-Based Long Short-Term Memory Network," 2024. doi: 10.2139/ssrn.5038611.

[9] I. V. Boyoukliev, H. N. Kulina, and S. G. Gocheva-Ilieva, "Modelling and Forecasting of EUR/USD Exchange Rate Using Ensemble Learning Approach," *Cybernetics and Information Technologies*, vol. 22, no. 4, pp. 142–151, Nov. 2022, doi: 10.2478/cait-2022-0044.

[10] R. Liao, P. Boonyakunakorn, N. Harnpornchai, and S. Sriboonchitta, "Forecasting the Exchange Rate for USD to RMB using RNN and SVM," *J. Phys.: Conf. Ser.*, vol. 1616, no. 1, p. 012050, Aug. 2020, doi: 10.1088/1742-6596/1616/1/012050.

[11] S. S. Namin and A. S. Namin, "FORECASTING ECONOMIC AND FINANCIAL TIME SERIES: ARIMA VS. LSTM".

[12] F. García, F. Guijarro, J. Oliver, and R. Tamošiūnienė, "Foreign Exchange Forecasting Models: ARIMA and LSTM Comparison," in *ITISE 2023*, MDPI, Jul. 2023, p. 81. doi: 10.3390/engproc2023039081.

[13] S. Ranjit, S. Shrestha, S. Subedi, and S. Shakya, "Comparison of algorithms in Foreign Exchange Rate Prediction," in *2018 IEEE 3rd International Conference on Computing, Communication and Security (ICCCS)*, Kathmandu: IEEE, Oct. 2018, pp. 9–13. doi: 10.1109/CCCS.2018.8586826.

[14] M. S. Islam and E. Hossain, "Foreign exchange currency rate prediction using a GRU-LSTM hybrid network," *Soft Computing Letters*, vol. 3, p. 100009, Dec. 2021, doi: 10.1016/j.socl.2020.100009.

[15] M. R. Pahlevi, K. Kusrini, and T. Hidayat, "Comparison of LSTM and GRU Models for Forex Prediction," *SinkrOn*, vol. 8, no. 4, pp. 2254–2263, Oct. 2023, doi: 10.33395/sinkron.v8i4.12709.

[16] R. Kaur *et al.*, "Development of a cryptocurrency price prediction model: leveraging GRU and LSTM for Bitcoin, Litecoin and Ethereum," *PeerJ Computer Science*, vol. 11, p. e2675, Mar. 2025, doi: 10.7717/peerj-cs.2675.

[17] Statistics Department, "Monthly Average Exchange Rate (Riel/U.S. $)." National Bank of Cambodia, Apr. 21, 2025. Accessed: May 10, 2025. [Online]. Available: https://www.nbc.gov.kh/download_files/data/english/5.monthlyaverageexchangeratemar-25_2192.xlsx

[18] O. B. Sezer, M. U. Gudelek, and A. M. Ozbayoglu, "Financial Time Series Forecasting with Deep Learning : A Systematic Literature Review: 2005-2019," Nov. 29, 2019, *arXiv*: arXiv:1911.13288. doi: 10.48550/arXiv.1911.13288.

[19] Y. A. LeCun, L. Bottou, G. B. Orr, and K.-R. Müller, "Efficient BackProp," in *Neural Networks: Tricks of the Trade*, vol. 7700, G. Montavon, G. B. Orr, and K.-R. Müller, Eds., in Lecture Notes in Computer Science, vol. 7700. , Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 9–48. doi: 10.1007/978-3-642-35289-8_3.

[20] A. Mohammed and R. Kora, "A comprehensive review on ensemble deep learning: Opportunities and challenges," *Journal of King Saud University - Computer and Information Sciences*, vol. 35, no. 2, pp. 757–774, Feb. 2023, doi: 10.1016/j.jksuci.2023.01.014.

[21] V. T. Ha, "Experimental Study on Remaining Useful Life Prediction of Lithium-Ion Batteries Based on Three Regressions Models for Electric Vehicle Applications," Jun. 14, 2023. doi: 10.20944/preprints202306.0999.v1.

[22] E. Kavlakoglu and E. Russi, "What is XGBoost?," What is XGBoost? Accessed: Jun. 28, 2025. [Online]. Available: https://www.ibm.com/think/topics/xgboost

[23] C. Olah, "Understanding LSTM Networks -- colah's blog," Understanding LSTM Networks. Accessed: May 27, 2025. [Online]. Available: https://colah.github.io/posts/2015-08-Understanding-LSTMs/