Systems and Soft Computing 7 (2025) 200411

Contents lists available at ScienceDirect

Systems and Soft Computing

journal homepage: www.journals.elsevier.com/soft-computing-letters

Value-at-risk estimation in the Tadawul exchange: A comparison of recurrent neural networks and GARCH models

Abdaljbbar B.A. Dawod [a,b], Mohammed Nsaif [c], Dhafer G. Honi [a,d], Ismail H. Abdi [e], Husam A. Neamah [f,*]

[a] Doctoral School of Informatics, University of Debrecen, Kassai ut 26, Debrecen, 4028, Hungary
[b] Department of Business Management, Al-imam University College, Balad, Iraq
[c] Department of Computer Science, University of Kufa, Kufa, Iraq
[d] Computer Science Department, College of Education for Pure Sciences, University of Basrah, Basrah, Iraq
[e] Department of Mathematics and Natural Sciences, Garissa University, Garissa, Kenya
[f] Department of Electrical Engineering and Mechatronics, Faculty of Engineering, University of Debrecen, Debrecen, Hungary

ARTICLE INFO

Keywords:
Tadawul
GARCH
LSTM
BiLSTM
GRU
BiGRU
VaR
ES

ABSTRACT

Value at Risk (VaR) and Expected Shortfall (ES) are two of the most established measures of market risk, yet both can lose reliability in volatile and nonlinear markets. This study examines how econometric models compare with modern recurrent neural networks in forecasting risk for the Saudi stock market. Using 4,340 daily closing prices of the Tadawul All Share Index (TASI) from March 2007 to July 2024, the analysis applies a sliding-window framework. Forecast accuracy is assessed through Root Mean Square Error (RMSE) and information criteria, while VaR and ES forecasts are validated using unconditional and conditional coverage tests. The study makes three key contributions: (i) it systematically contrasts GARCH specifications with recurrent models (LSTM, BiLSTM, GRU, BiGRU) in a long-horizon dataset, (ii) it extends backtesting of VaR and ES across 17 years of data, and (iii) it provides practical insights for combining econometric and machine-learning methods in emerging markets. Results show that BiGRU achieved the highest predictive accuracy, while GRU offered the best balance between accuracy and parsimony. Heavy-tailed GARCH models produced the most reliable tail-risk forecasts, consistently passing tests at the 95% and 99% confidence levels. Although recurrent networks were less accurate at extreme quantiles, they effectively captured nonlinear patterns and volatility clustering, making them a valuable complement for practical risk forecasting.

1. Introduction

Financial institutions face inherent risks from daily operations, including business activities, new ventures, mergers, and external factors like competition. These threats are broadly categorized as market risk (e.g., fluctuations in interest or exchange rates), transaction risk (e.g., dealings with counterparties), and operational risk (e.g., internal system or human failures). Value at Risk (VaR) is the predominant measure for quantifying such risks at both individual trade and portfolio levels. Given the complexity of financial markets, numerous statistical and machine learning approaches have been developed to improve forecasts of returns, volatility, and associated risk [1,2]. Hence, VaR can be calculated either parametrically, relying on Gaussian assumptions, or non-parametrically through historical simulation. Many practitioners rely heavily on historical data due to its availability and the confidence it provides in reflecting actual market behavior [3].

However, various statistical models have been employed to generate historically simulated data for calculating VaR over a targeted risk horizon, yet most of these methods struggle to capture the high volatility inherent in stock market data [4]. The Generalized Autoregressive Conditional Heteroskedastic (GARCH) family of models is a stationary, nonlinear framework for fitting econometric data that is characterized by high volatility [5]. Whereas some GARCH models may struggle to fit markets with extreme volatility, neural networks can effectively model nonlinear relationships by identifying complex patterns in data, making them well-suited for financial forecasting. Several studies have employed architectures such as long short-term memory (LSTM), Bi-Directional LSTM (BiLSTM), and gradient boosting to predict stock prices and market trends with high accuracy. While these methods

* Corresponding author.
E-mail addresses: dawod.abdaljbbar@inf.unideb.hu (A.B.A. Dawod), mohammed.nsaif@unkufa.edu.iq (M. Nsaif), dhafer.honi@inf.unideb.hu (D.G. Honi), ismail.abdi@gau.ac.ke (I.H. Abdi), husam@eng.unideb.hu (H.A. Neamah).

https://doi.org/10.1016/j.sasc.2025.200411
Received 13 July 2025; Received in revised form 23 October 2025; Accepted 25 October 2025
Available online 30 October 2025

show promising results, further refinement of network structures—through experimentation with depth, layer configuration, and model parameters—could enhance predictive performance and reliability [6,7].

Accurately estimating VaR in volatile markets remains a central challenge in financial risk management. Several studies have attempted to refine volatility modeling through diverse approaches: copula–GARCH frameworks have improved portfolio VaR accuracy compared to traditional methods [8]; volatility adjustments via exponentially weighted moving averages enhanced the reliability of the historically simulated VaR in the Khartoum Stock Exchange [4]; and GARCH-family extensions such as EGARCH and TGARCH demonstrated effectiveness in capturing medium-, long-, and short-term volatility dynamics in the Saudi stock market [9]. Volatility in commodity prices during crisis periods, such as the global financial crisis and COVID-19, has further underscored the sensitivity of these markets to negative shocks [10]. More recent research has proposed hybrid and extended econometric frameworks, including post-selection ARCH, skewed error distributions, and GARCH-Mixed Data Sampling (GARCH-MIDAS), to address the limitations of conventional GARCH models in handling long-term dynamics and multi-frequency data [11–13]. Despite these advancements, the accurate estimation of VaR under conditions of heightened volatility and structural breaks continues to pose significant challenges, motivating the need for improved methodologies that can capture both short-term fluctuations and long-term risk drivers.

Over the past decade, methods such as Support Vector Machines (SVMs) and Multi-Layer Perceptrons (MLPs) have been applied to stock market data, demonstrating improved predictive performance in markets such as the Karachi Stock Exchange and NIFTY-50 [14,15]. Similarly, the Two-stream Gated Recurrent Unit (TGRU) has also shown robustness to market fluctuations and the ability to address market-related risks, achieving higher average accuracy compared to LSTM and GRU models [16]. More broadly, approaches including Recurrent Neural Networks (RNNs), Gradient Boosting (GBoost), Decision Trees, Random Forests, Light Gradient Boosting Machine (LightGBM), extreme GBoost (XGBoost), Extreme Learning Machines (ELM), and Backpropagation Neural Networks (BPNN) have been employed to predict financial markets, with studies reporting accuracies ranging from 50.9% to 56.0% in binary market movement prediction [17,18]. Shah et al. [15] showed that combining CNN with LSTM improves the prediction of NIFTY-50 stock prices compared to using LSTM alone.

Moreover, several studies have proposed hybrid models to improve forecasting accuracy. For instance, Dave et al. [19] combined the integrated Autoregressive Moving Average (ARIMA) and LSTM for housing sales in Turkey, where ARIMA captured linear patterns and LSTM modeled nonlinearities, outperforming individual models; Siami et al. [20] showed that BiLSTM, despite slower training, achieved higher accuracy than LSTM and ARIMA on Yahoo Finance time series. Binmakhashen et al. [21] compared recurrent neural networks with traditional models such as linear regression, decision trees, and random forests for forecasting Tadawul stock prices. They found that simpler models generally performed better for midterm index prediction, and that shallow LSTM architectures outperformed deeper designs. Similarly, Alshammari et al. [22] integrated wavelet transforms with the EGARCH model, demonstrating superior performance in forecasting Tadawul volatility.

This study evaluates the performance of the GARCH model and recurrent network structures, including LSTM, BiLSTM, GRU, and Bi-GRU, for improving the estimation of the historically simulated VaR and Expected Shortfall (ES). A comparative analysis was conducted to examine the ability of these models to capture volatility clustering and nonlinear market dynamics, building on earlier evidence of the effectiveness of GARCH-family models and RNNs in financial risk measurement as found by Hansen and Lunde [13] and kosapattarapim et al. [12]. The findings aim to determine the most reliable approach for forecasting stock market volatility and enhancing the accuracy of risk metrics, thereby offering practical insights for investors, regulators, and financial analysts. Besides the root mean square error (RMSE), the Akaike (AIC) and Bayesian (BIC) information criteria have been adopted to determine the goodness of fit of the different approaches. Additionally, the Backtesting approaches were subsequently used to test the accuracy of the estimated VaR. The paper is organized as follows: Section 2 describes the dataset, methodology, and evaluation metrics; Section 3 presents the computational analysis and results; Section 4 concludes the discussion.

2. Methodology

This section presents the framework and evaluation metrics for estimating 1-day VaR and ES of Tadawul index daily prices, with Fig. 1 illustrating the main components of the estimation process.

In the data preparation stage, we measure stock market volatility using returns rather than raw closing prices (x_t), as returns are scale-free and stationary under standard assumptions, and are more suitable for statistical modeling and comparison across assets [23]. The return at time t is defined as

$$r_t = \frac{\left(x_t - x_{t-1}\right)}{x_{t-1}}, \tag{1}$$

where x_t and x_{t-1} denote consecutive daily closing prices.

Hence, our target is to find a model that precisely mimics the returns r_t. The following subsections highlight the parametric statistical models used for this task and various deep-learning models that show their excellence in the literature. For the RNN models, the input data are scaled before training to improve numerical stability and accelerate convergence [24]. We apply min–max normalization, defined as

$$scaled(x_t) = \frac{x_t - \min\left(x_t\right)}{\max\left(x_t\right) - \min\left(x_t\right)} \tag{2}$$

which maps the values into the $[0, 1]$ range, ensuring consistent magnitude across features.

2.1. Dataset

Tadawul is the sole stock exchange in Saudi Arabia and one of the largest financial institutions in the Gulf Cooperation Council (GCC) region [25]. Formally established in March 2007 as a joint stock company, its origins trace back to 1954 as an informal financial market [25,26]. Tadawul plays a central role in the Saudi economy by providing a platform for trading securities (e.g., stocks and bonds) in line with the Kingdom's Vision 2030 [27]. It is also recognized for its comprehensive indices, most notably the Tadawul All Share Index (TASI), which tracks the performance of listed companies. The exchange supports capital market development and attracts foreign investment [26,27].

The dataset of Tadawul indices was obtained from Yahoo Finance [28], covering the period from 19 March 2007 to 24 July 2024. The dataset includes the variables Open, High, Low, Close, Adj. Close, and Volume. The analysis focuses on the closing prices. After removing incomplete cases, the data was divided into training (80%) and validation (20%) sets.

The returns were computed from the closing prices and are shown in Fig. 2. Subsequently, Fig. 3 illustrates higher positive kurtosis relative to the normal distribution. Since kurtosis reflects the likelihood of extreme outcomes, elevated values indicate greater risk due to heavier tails. Descriptive statistics of closing prices and returns are reported in Table 1, which also provides insight into normality assumptions.

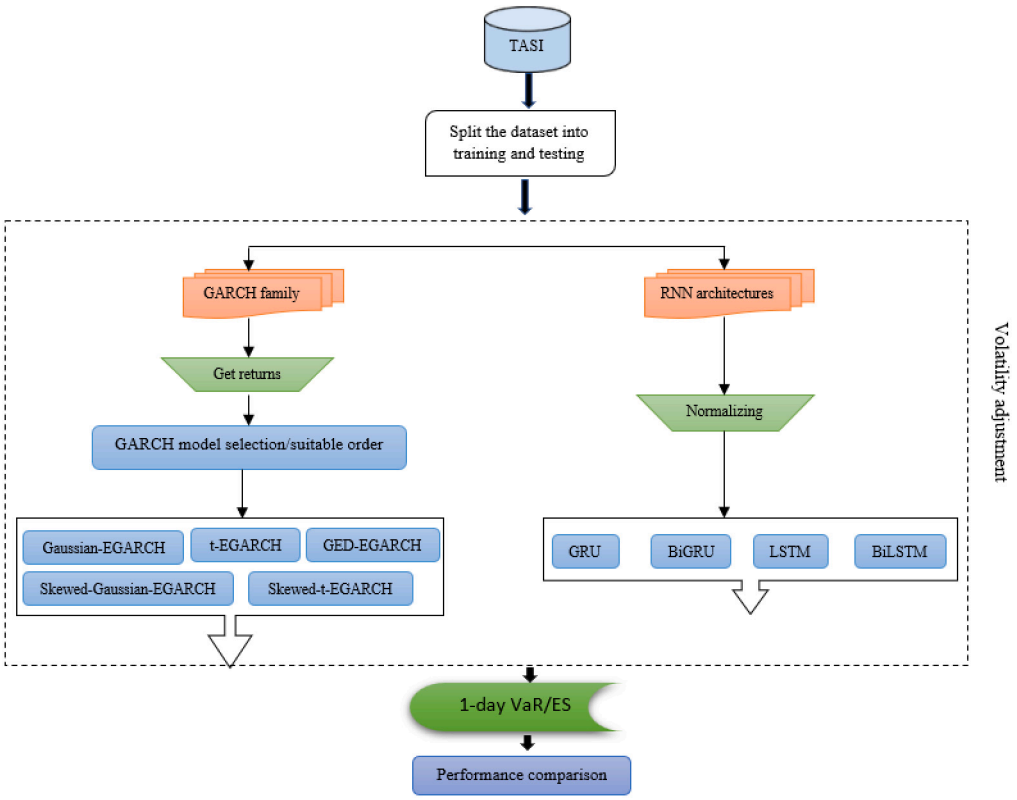

Fig. 1. Proposed methodological framework outlining the study's workflow and key steps.

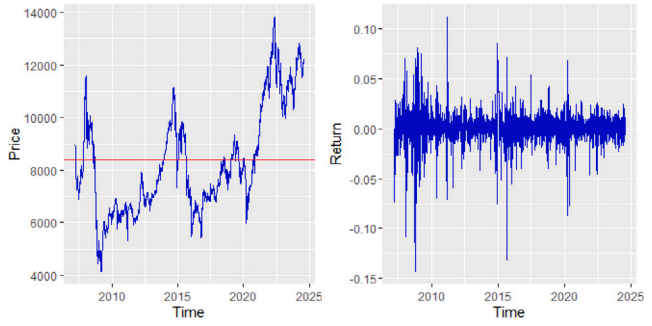

Fig. 2. Historical closing prices of the Tadawul index and the corresponding daily returns over the years.

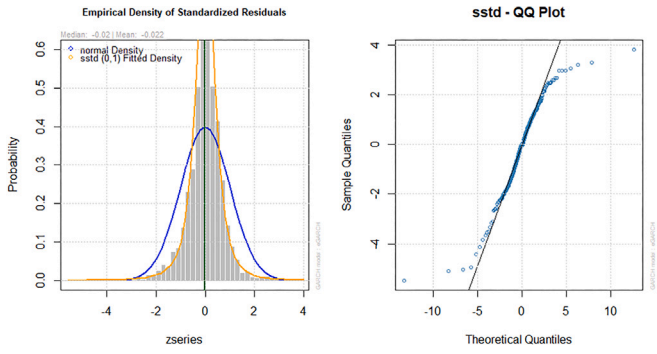

Fig. 3. Empirical density and QQ plot of standardized residuals, assessing normality assumptions.

Table 1
Summary of descriptive statistics for Tadawul closing prices and their corresponding returns over the period 19 March 2007 to 27 July 2024, providing insight into the distribution and variability of the data.

	mean	sd	min	max	skew	kurt.
Returns	0.00	0.01	−0.14	0.11	−1.29	16.84
Prices	8393.85	2042.17	4130.01	13 820.35	0.54	−0.66

2.2. Value at Risk (VaR)

VaR is a quantitative risk measure that quantifies the maximum expected loss a portfolio might encounter over a specified time horizon. It represents a probability that the losses will not exceed that value within a given confidence level [29]. Assuming that R is the returns, VaR at level α is the smallest number l such that the probability that $L = -R$ does not exceed l is at least $(1-\alpha)$. Mathematically, $\text{VaR}_\alpha(R)$ represents $(1-\alpha)$-quantile of L such that

$$\begin{aligned}\text{VaR}_\alpha(R) &= \inf\{l \in \mathbb{R} : F_R(r) > \alpha\} \\ &= F_L^{-1}(1-\alpha)\end{aligned} \tag{3}$$

For any real random variable R, its cumulative distribution function F_X is well defined.

In practice, the historical Simulations (HS) approach uses the historical returns of the portfolio over a specific horizon to estimate VaR. It only relies on actual past performance, regardless of any particular pattern in returns. To calculate VaR, historical data is collected, and its returns are extracted using (1), then these returns are sorted in ascending or descending order, and the empirical quantile corresponding to the desired confidence level (α) is determined and multiplied by the volume of the portfolio [3]. Fortunately, Pérignon and Smith [30] reported that more than three-quarters of bankers prefer this method to calculate VaR.

On the other hand, ES or Conditional VaR (CVaR) represents the weighted arithmetic mean of the 'extreme' losses in the tail of the distribution of returns beyond the VaR cutoff point. The estimated ES of the portfolio with returns R and estimated VaR at significance level α is given mathematically as follows

$$\begin{aligned} ES_{\alpha}(R) &= E\left[R|R \leq -VaR_{\alpha}(R)\right] \\ &= -\frac{1}{\alpha}\int_{-\infty}^{-VaR(R)} rf(r)\,dr, \end{aligned} \tag{4}$$

where $f(r)$ is the probability density of the returns

2.3. GARCH-family models

GARCH models and their variants have been thoroughly adopted in financial econometrics to forecast volatility in financial time series. GARCH models overcame the limitation of ARIMA models by explicitly modeling the conditional variance of the data. Engle [31] proposed the ARCH model to fit the conditional variance of a time series, particularly in the presence of volatility clustering, where periods of high volatility tend to cluster. The daily returns can be represented by the ARCH model of order p as

$$\begin{aligned} r_t &= \sqrt{\sigma_t^2}z_t, \\ \sigma_t^2 &= \alpha_0 + \sum_{i=1}^{p}\alpha_i r_{t-i}^2. \end{aligned} \tag{5}$$

where $z_t \sim D(0,1)$ is a white noise error term, and D is a specified probability distribution, σ_t^2 is the conditional variance of the returns at time t, and the coefficients α_i are non-negative parameters. Subsequently, Bollerslev [5] generalized the ARCH (i.e., GARCH) by incorporating the past squared observations of the error term and past variances into the conditional variance. The conditional variance of the returns at time t is σ_t^2 and given by

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^{p}\alpha_i r_{t-i}^2 + \sum_{j=1}^{q}\beta_j\sigma_{t-j}^2, \tag{6}$$

where the coefficients α_i and β_j are non-negative parameters. The conditional variance of the time series in (6) is equivalent to the ARMA model of orders p and q. The Exponential GARCH (EGARCH) model, introduced by Nelson [32], extends the standard GARCH framework by allowing asymmetric effects in volatility without imposing positivity constraints on conditional variances. It captures key empirical features, such as small positive returns exerting a stronger influence on volatility than small negative returns, while large negative shocks have a greater impact than large positive ones. The conditional variance in EGARCH(p, q) model is given by

$$\begin{aligned} \ln\sigma_t^2 &= \alpha + \sum_{i=1}^{q}\alpha_i g\left(z_{t-i}\right) + \\ &\quad \sum_{j=1}^{p}\beta_j \ln\left(\sigma_{t-j}^2\right) \\ g\left(z_{t-i}\right) &= \left\{\gamma\frac{r_{t-i}}{\sqrt{\sigma_{t-1}^2}} + \frac{|r_{t-1}|}{\sqrt{\sigma_{t-1}^2}} - \sqrt{\frac{2}{\pi}}\right\} \end{aligned} \tag{7}$$

where α_i, β_j, and γ are the parameters, and the function $g(z_{t-i})$ allows the sign and the magnitude of Z_t to have separate effects on the volatility. Since the logarithmic value of σ_t^2 might have a negative sign, there will be no sign restrictions for the parameters. Glosten et al. [33] introduced GJR-GARCH to capture the differential effects of positive and negative shocks on volatility. The conditional variance is given by

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^{p}\alpha_i r_{t-i}^2 + \gamma D\left(\Psi_{t-1}\right) r_{t-1}^2 + \sum_{t-j}^{q}\beta_j\sigma_{t-j}^2. \tag{8}$$

where α_i, β_i and γ are constant parameters and $D\left(\Psi_{t-1}\right)$ is a dummy variable that given by

$$D\left(\Psi_{t-1}\right) = \begin{cases} 1, & \text{if } r_t \geq 0. \\ 0, & \text{if } r_t < 0 \end{cases}$$

The strictly positive condition of the variance is guaranteed when $p = 1, \alpha_0, \alpha_1 > 0, \alpha_1 + \gamma_1 > 0$, and $\beta_1 > 0$. Other commonly used GARCH variants include IGARCH, TGARCH, and APARCH [34,35]. Before applying these models, we assessed return stationarity using the Dickey–Fuller test [36] and verified the presence of ARCH effects by testing the null hypothesis of i.i.d. Gaussian residuals against evidence of conditional heteroskedasticity.

2.4. Deep learning models

RNNs have been suggested for sequential data series, leveraging their inherent flexibility and capacity to analyze time series data, even in the presence of nonlinear and non-stationary characteristics [37,38]. These strengths have led practitioners to adopt RNN for predicting market returns. For example, LSTM was developed by Schmidhuber and Hochreiter [39] and is characterized by its capacity to handle vanishing and exploding gradient problems and capture long-term dependencies. LSTMs are designed to address the shortcomings of conventional RNNs in terms of retaining information over extended sequences by incorporating an advanced gating mechanism that consists of three gates: input, forget, and output. The input gate controls the flow of new information into the cell state, as shown in (9). The forget gate decides what information to discard from the cell state, as shown in (10). The output gate determines the output of the cell based on the current input and the memory stored in the cell state, as shown in (11). These gates regulate the information flow through the cell state, as illustrated in Fig. 4 (see [40]).

$$i^{(t)} = \sigma\left(W_i u_{i,j}^t + Q_i h^{(t-1)} + b_i\right) \tag{9}$$

$$f^{(t)} = \sigma\left(W_f u_{i,j}^t + Q_f h^{(t-1)} + b_f\right) \tag{10}$$

$$o^{(t)} = \sigma\left(W_o u_{i,j}^t + Q_o h^{(t-1)} + b_o\right) \tag{11}$$

In these equations, W_i, W_f, and W_o are weight matrices for the current input data, $u_{i,j}^t$. On the other hand, Q_i, Q_f, and Q_o are weight matrices for the current hidden state, $h^{(t-1)}$. σ denotes the sigmoid activation function, and b_i, b_f, and b_o are bias terms. Each LSTM cell receives a candidate cell state from the previous LSTM cell:

$$\tilde{c}^{(t)} = \tanh\left(W_c u_{i,j}^t + Q_c h^{(t-1)} + b_c\right) \tag{12}$$

Then the memory cell state, $c^{(t)}$, and the final hidden state, $h^{(t)}$, are updated as follows:

$$c^{(t)} = f^{(t)} \odot c^{(t-1)} + i^{(t)} \odot \tilde{c}^{(t)} \tag{13}$$

$$h^{(t)} = o^{(t)} \odot \tanh(c^{(t)}) \tag{14}$$

The proposed architecture enables LSTMs to selectively remember or forget information at each time step, which makes LSTM an efficient approach with sequential data, such as time series prediction, speech recognition, and natural language processing (NLP). Consequently, the network has been widely used in various applications, such as speech recognition, machine translation, and time series forecasting [41]. Moreover, BiLSTM extends the capabilities of LSTM by processing input sequences in two directions, forward and backward [42]. It captures contextual information from both past and future time steps, enabling them to better understand the context and dependencies within the sequence. Obtaining gradient information from both directions during training resulted in more stable and effective training processes for BiLSTM. Hence, it is an incredibly efficient neural network for various NLP tasks [43].

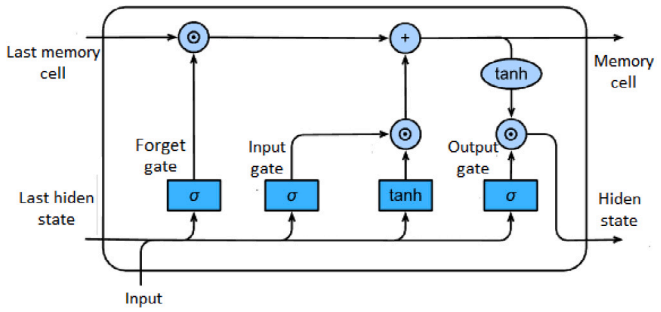

Fig. 4. The structure of the LSTM cell. the ⊙, and σ refer to element-wise production and sigmoid function.

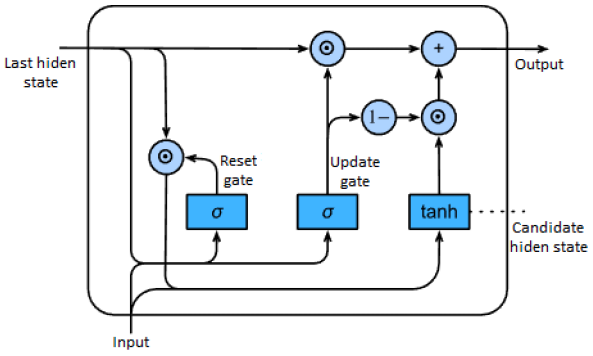

Fig. 5. The structure of a GRU cell, where ⊙ and σ refer to element-wise multiplication and sigmoid functions, respectively.

Moreover, GRU was proposed by [44] as a simplified version of LSTM, utilizing fewer parameters and achieving faster training times. Consequently, GRUs can handle long-term dependencies in sequential data with efficient computational power. This efficiency is inherent due to their structure, which consists of update and reset gates, as shown in Fig. 5. The update gate controls the flow of past information that should be kept for future time steps, as shown in (15).

$$z^{(t)} = \sigma\left(W_z u_{i,j}^t + Q_z h^{(t-1)} + b_z\right) \tag{15}$$

In this equation, W_z is a weight matrix for the current input, Q_z is a weight matrix for the last hidden state, $h^{(t-1)}$, σ denotes the sigmoid activation function, and b_z is a bias term. The reset gate decides what information to forget at the current time step, as shown in (16).

$$r^{(t)} = \sigma\left(W_r u_{i,j}^t + Q_r h^{(t-1)} + b_r\right) \tag{16}$$

In this equation, W_r is a weight matrix for the current input, Q_r is a weight matrix for the last hidden state, $h^{(t-1)}$, σ denotes the sigmoid activation function, and b_r is a bias term.

The candidate hidden state is calculated as follows in (17):

$$h'^{(t)} = \tanh\left(W u_{i,j}^t + r^{(t)} \odot Q h^{(t-1)} + b\right) \tag{17}$$

Finally, the new hidden state is updated as shown in (18):

$$h^{(t)} = z^{(t)} \odot h^{(t-1)} + \left(1 - z^{(t)}\right) \odot h'^{(t)} \tag{18}$$

GRU has been widely adopted in NLP, speech recognition, and time series analysis due to its effectiveness and efficiency in modeling sequential data. Additionally, the network is capable of handling larger and nonlinear datasets [44,45].

BiGRU was built on the architecture of GRUs by combining the strengths of bidirectional processing and gated recurrent units to enhance sequential data modeling. BiGRU merges the hidden states from both directions, similar to BiLSTMs, to produce a more thorough representation of the input sequence. Owing to their ability to process in both directions, BiGRUs are particularly useful for tasks such as machine translation, named entity identification, and sentiment analysis, as they excel at capturing context and long-term dependencies.

2.5. Evaluation metrics

To evaluate model performance, the complete dataset was partitioned into 80% as a training set and 20% for validation. In addition, the accuracy of the prediction models is evaluated using a range of metrics, with RMSE being the most widely adopted. Practically, RMSE is calculated as

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

where y is a tensor of the target values, and $\hat{y}$ is a tensor of the predictions and n is the length of the tensors.

Moreover, AIC and BIC are used to estimate the prediction error and report the goodness of fit of the model relying on the maximum likelihood estimate (MLE). Considering the targeted model has MLE L with k estimated parameters, the AIC/BIC are given as follows

$$AIC = 2k - 2\ln(L)$$
$$BIC = 2k\ln(n) - 2\ln(L)$$

where n is the size of our training set. Backtesting methods are adopted to evaluate the accuracy of the estimated VaR. Daily returns are assumed to be independent and identically distributed (i.i.d.) from the Bernoulli distribution. It is given by the indicator function as follows

$$I_{\alpha,t+1} = \begin{cases} 1 & \text{if } r_{t+1} < -\text{VaR}_{1,\alpha,t} \\ 0 & \text{otherwise.} \end{cases}$$

where r_{t+1} is the daily return at time $t+1$.

From the definition of the indicator function $\{I_{\alpha,t}\}$, let the process of success/failure be defined by the random variable $X_{n,\alpha}$, and it will follow a Binomial distribution with parameters n and α.

The most commonly used Backtesting method is the Kupiec test (Unconditional Coverage) [46], the method considers the estimated VaR to be accurate when the observed frequency of returns exceeding the VaR threshold matches the expected frequency based on the assumed binomial distribution with a success probability equal to the VaR significance level. The test statistic for the method is given as

$$LR_{uc} = \frac{\Pi_{exp}^{n_1}(1-\Pi_{exp})^{n_0}}{\Pi_{obs}^{n_1}(1-\Pi_{obs})^{n_0}} \sim \chi_\alpha^2(1) \tag{19}$$

given that $n_0 = n - n_1$, where n represents the number of the training set, n_1 presents the frequency of returns exceeding VaR_α. In addition, $\Pi_{obs} = \frac{n_1}{n}, \Pi_{exp} = \alpha$. The null hypothesis of LR_{uc} states that the observed failure rate is consistent with the expected failure rate (i.e., the VaR model is accurate).

Later, Christoffersen [47] proposed the Independence test, which considers an estimated VaR to be accurate if a VaR violation on a given day is independent of whether a violation occurred on the previous day. The test statistic is given by

$$LR_{ind} = \frac{\Pi_{obs}^{n_1}\left(1-\Pi_{obs}\right)^{n_0}}{\Pi_{01}^{n_{01}}\left(1-\Pi_{01}^{n_{00}}\right)\Pi_{11}^{n_{11}}\left(1-\Pi_{11}\right)^{n_{10}}} \sim \chi_\alpha^2(1) \tag{20}$$

n_{ij} is the number of returns with indicator i followed by indicator j, given that $i, j = 0, 1$. $n_0 = n_{10} + n_{00}$, $n_1 = n_{11} + n_{01}$, $\Pi_{01} = \frac{n_{01}}{n_{00}+n_{01}}$, and $\Pi_{11} = \frac{n_{11}}{n_{10}+n_{11}}$.

To jointly test the correct success/failure rate and the independence of exceedances, the combined test statistic is calculated as the sum of the Kupiec and Independence test statistics. This new test is referred to as the Conditional Coverage test, and it is given as follows

$$LR_{cc} = LR_{uc} + LR_{ind} \sim \chi_\alpha^2(2)$$

Table 2
Results of the ARCH effect tests, indicating the presence of conditional heteroskedasticity in the series.

Testing method	Statistics	P-value
Q test	810.7	2.2e−16
Lagrange multiplier (LM) test	478.97	2.2e−16

Table 3
Selection criteria for GARCH models, used to identify the most appropriate model specification for the data.

Model	AIC	BIC	SIC	Hannan
norm - sGARCH(1, 1)	−6.22	−6.21	−6.22	−6.22
snorm - sGARCH(1, 1)	−6.24	−6.23	−6.24	−6.23
std - sGARCH(1, 1)	−6.44	−6.43	−6.44	−6.44
sstd - sGARCH(1, 1)	−6.44	−6.43	−6.44	−6.44
norm - eGARCH(1, 1)	−6.23	−6.22	−6.23	−6.23
snorm - eGARCH(1, 1)	−6.24	−6.23	−6.24	−6.23
std - eGARCH(1, 1)	−6.46	−6.45	−6.46	−6.45
sstd - eGARCH(1, 1)	−6.46	−6.45	−6.46	−6.45
norm - gjrGARCH(1, 1)	−6.25	−6.24	−6.25	−6.24
snorm - gjrGARCH(1, 1)	−6.26	−6.25	−6.26	−6.26
std - gjrGARCH(1, 1)	−6.46	−6.45	−6.46	−6.45
sstd - gjrGARCH(1, 1)	−6.46	−6.45	−6.46	−6.45
norm - iGARCH(1, 1)	−6.22	−6.21	−6.22	−6.22
snorm - iGARCH(1, 1)	−6.23	−6.23	−6.23	−6.23
std - iGARCH(1, 1)	−6.44	−6.44	−6.44	−6.44
sstd - iGARCH(1, 1)	−6.44	−6.44	−6.44	−6.44

3. Result analysis

In financial time series, the actual prices have neither a constant mean nor a constant variance, which violates the modeling assumptions (see Fig. 2). The transformation described in (1) extracts returns from the actual data, which is the percentage change in price. The Dickey–Fuller test has been used to check the stationarity; the test showed that our returns fulfill the stationarity assumption. In addition, the normality of returns has been checked by conducting the Jarque–Bera test; the null hypothesis that the returns are not normally distributed is rejected. The ARCH effect has been investigated using the Ljung–Box test on square returns (Q test) and the Lagrange Multiplier test (ARCH-LM). The results reported in Table 2 reject the null hypothesis of independence in the given time series.

The following subsections discuss the results of each modeling approach and their impact on VaR and ES estimation.

3.1. GARCH family models

Table 3 shows the best candidates of the GARCH-family models with different distributional assumptions. The selection process depends on AIC, BIC, SIC, and Hannan criteria. Consequently, EGARCH(1,1) with Skewed Student-t distribution represents the best-fitting model for our training set. The estimated parameters, all of which were significant, are presented in Table 8. This suggests that incorporating asymmetry and heavy-tailed error distributions improves the ability of the models to capture volatility dynamics. Overall, the results highlight the importance of both model structure and distributional assumptions when selecting the most suitable GARCH specification.

In addition, various diagnostic plots were investigated to examine the validity of the selected model in fitting the volatility of Tadawul exchange prices. The diagnostic plots of Skwed-student-t-EGARCH(1,1) are illustrated as follows

- Fig. 3 depicts the empirical density of our returns' standardized residuals and QQ plots. The empirical density demonstrates that the standardized residuals are approximately normally distributed. In addition, the distribution of our returns compared with the theoretical distribution in the QQ plot shows that the standardized residuals are approximately normally distributed.

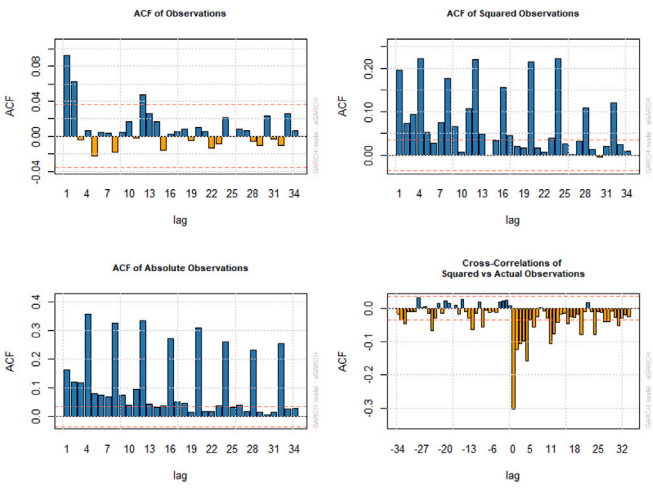

Fig. 6. Autocorrelation and cross-correlation functions of raw, squared, and absolute observations for dependence analysis.

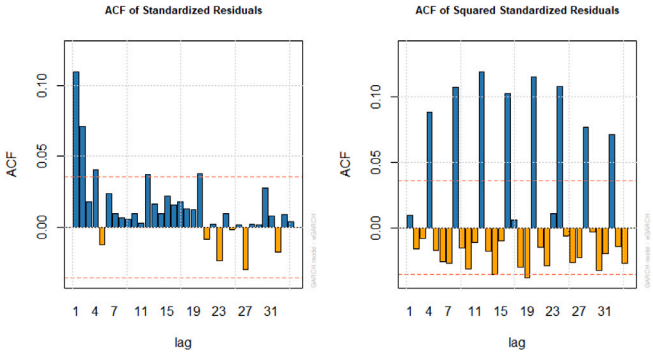

Fig. 7. Autocorrelation functions of standardized residuals and their squares for dependence diagnostics.

- Fig. 6 illustrates the autocorrelation function (ACF). It shows the ACF of our returns, which reveals weak autocorrelation and is a good indicator for our chosen GARCH model. It also shows the ACF of squared observations and the absolute returns with the interest of removing the sign from our returns, focusing on volatility. Both ACFs exhibit a weak autocorrelation, which is consistent with the results of our model. In addition, we highlight the positive cross-correlation between the squared returns and the squared lagged returns. This finding demonstrates that periods of high volatility are likely to be preceded by periods of high volatility.
- The ACFs of Standardized Residuals and Squared Standardized Residuals are shown in Fig. 7, both ACFs indicate the existence of weak autocorrelation in the standardized residuals of the squared standardized residuals, which is also a good sign for the validity of our selected model.

The diagnostic plots show the adequacy of our GARCH model for fitting the volatility of Tadawul exchange prices.

3.2. RNN

Python 3.8 was implemented on Windows 11 Education *N* system with an Intel Core i7-9750H processor at 2.60 GHz and 16 GB of RAM. The number of epochs and batch size were set to 16 and 32, respectively; these values are fixed across all models. The shape of the output and number of parameters (trainable and non-trainable) of

Table 4
Key parameters of the LSTM model used for forecasting.

Vector output technique		
Layer type	Output shape	Param #
LSTM	(None, 50)	10 400
Dense	(None, 1)	51

LSTM trainable params: 10,451.
Non-trainable params: 0.

Table 5
Key parameters of the GRU model used for forecasting.

Vector output technique		
Layer type	Output shape	Param #
GRU	(None, 50)	7950
Dense	(None, 1)	51

GRU trainable params: 8001.
Non-trainable params: 0.

Table 6
Key parameters of the BiLSTM model used for forecasting.

Vector output technique		
Layer type	Output shape	Param #
BiLSTM	(None, 100)	20 800
Dense	(None, 1)	101

BiLSTM trainable parameters: 20 901.
Non-trainable parameters: 0.

Table 7
Key parameters of the BiGRU model used for forecasting.

Vector output technique		
Layer type	Output shape	Param #
BiGRU	(None, 100)	15 900
Dense	(None, 1)	101

BiGRU trainable parameters: 16 001.
Non-trainable parameters: 0.

the trained RNN structures are listed in Tables 4–7. It is clear that as we move from LSTM to BiGRU, the structure becomes more intricate and the number of parameters rapidly increases. We then compare the actual prices in the last financial year to the forecasted prices for the same period using the GARCH model, LSTM, BiLSTM, GRU, and BiGRU, as shown in Figs. 8 and 9. Consequently, Table 9 presents a comparative evaluation of the GARCH family against recurrent structures (LSTM, GRU, BiLSTM, and BiGRU) using RMSE, AIC, and BIC metrics. The results clearly highlight the superior predictive performance of the BiGRU model, which achieves the lowest RMSE (0.01135), indicating a higher degree of accuracy in tracking volatility dynamics compared to both the GARCH model (0.0185) and the other recurrent architectures. GRU also performs competitively with an RMSE (0.01815) comparable to GARCH, suggesting its ability to effectively capture temporal dependencies with a relatively simple design.

In terms of information criteria, GRU stands out with the lowest AIC (16,002) and BIC (64,058.95), underscoring its efficiency in balancing model complexity with predictive capability. Although BiGRU yields the best accuracy based on RMSE, its information criteria values are higher, reflecting the trade-off between predictive precision and model parsimony. LSTM and BiLSTM, by contrast, exhibit relatively weaker performance, as indicated by higher RMSE values (0.04640 and 0.04417, respectively) and substantially larger AIC and BIC scores, suggesting over-parameterization relative to their predictive gains.

These findings reinforce the evidence from recent studies showing that GRU and BiGRU structures often outperform both traditional econometric approaches and other recurrent designs in volatility forecasting. For instance, SHEN et al. [48] and Umezuruike et al. [49] demonstrated the robustness of GRU in financial prediction tasks, attributing its effectiveness to computational efficiency and its ability to capture nonlinear dynamics. Furthermore, Shaikh and Ramadass [50] highlighted the advantages of bidirectional extensions, such as BiGRU, in handling abrupt market fluctuations by integrating forward- and backward-looking information. Consistent with BinMakhashen et al. [21], which found that simpler models generally provide better midterm forecasts for the Tadawul index, the present results indicate that while BiGRU achieves the highest accuracy, GRU offers a more parsimonious balance between accuracy and complexity, making it well-suited for practical forecasting.

3.3. VaR/ES and backtesting results

The numerical results of VaR and ES under different significance levels and different modeling methods are shown in Table 10. There is no doubt that VaR and ES are correlated negatively with the significance levels, indicating a higher risk as the confidence levels increase. Also, HS and Gaussian methods provide relatively moderate estimates across different thresholds, though the Gaussian method tends

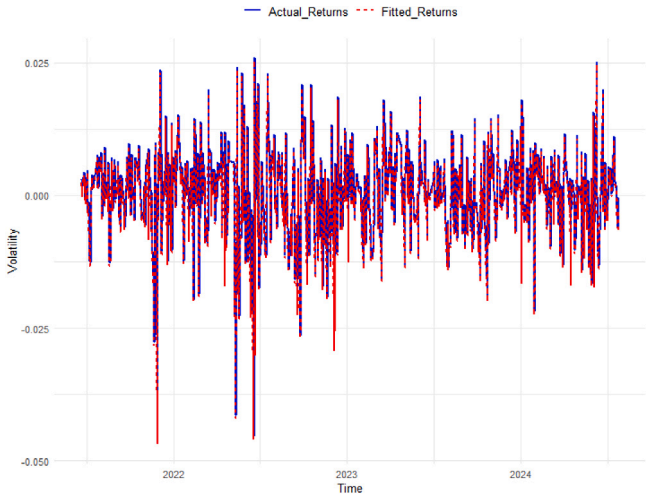

Fig. 8. Comparative analysis of Tadawul Exchange volatility forecasts from the GARCH(1,1) model with skewed Student-t distribution against validation data.

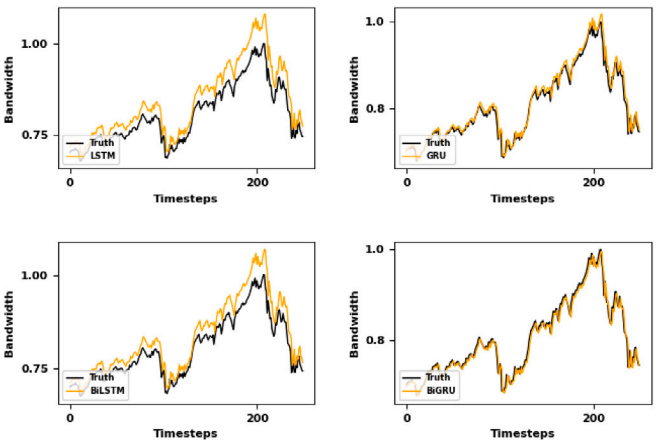

Fig. 9. Comparison of actual closing prices and forecasts from four RNN structures over the last 250 trading days.

to underestimate tail risk, particularly at more extreme quantiles. The Cornish–Fisher expansion shows instability, producing unrealistically large losses at the 1% and 0.5% levels, which reflects its sensitivity

to higher-order moments. Monte Carlo simulation offers more stable outcomes, though its estimates remain conservative compared with heavy-tailed GARCH specifications.

Moreover, within the GARCH family, models incorporating Student-t and skewed Student-t distributions (t-EGARCH and skewed-t-EGARCH) generate deeper quantile losses, highlighting their ability to accommodate fat tails and asymmetry. In contrast, the GED-EGARCH specification produces extreme estimates at lower quantiles, which suggests numerical instability or poor tail calibration. The recurrent structures (LSTM, BiLSTM, GRU, and BiGRU) deliver estimates that are close to HS benchmarks, with BiGRU slightly deeper in the tails, consistent with its stronger predictive performance shown in prior accuracy comparisons.

Backtesting results at the 90% and 95% confidence levels (Table 11) reveal important differences in model adequacy. The HS approach passes at the 90% level under the unconditional coverage test (LR_{uc}), but fails the conditional coverage (LR_{cc}), indicating clustering of violations. Gaussian and Cornish–Fisher methods fail decisively at both coverage levels, confirming their inability to model fat-tailed returns. Monte Carlo simulation shows some improvement but remains statistically rejected. Among GARCH variants, Gaussian-EGARCH demonstrates acceptable coverage at 95%, while skewed and heavy-tailed EGARCH specifications yield mixed results, with skewed-t-EGARCH performing best. Recurrent structures (particularly BiGRU) show low test statistics compared with LSTM, but the null of correct coverage is generally rejected, reflecting room for improvement in tail calibration.

At higher confidence levels (99% and 99.5%), presented in Table 12, the differences become sharper. Gaussian, Monte Carlo, and HS exhibit mixed adequacy, often underestimating tail risk. The skewed-t-EGARCH and t-EGARCH models perform markedly better, frequently passing both unconditional and conditional coverage tests, confirming their robustness in capturing extreme events. In contrast, LSTM-based models fail substantially, with excessively large test statistics that point to systematic underestimation of tail probabilities. BiGRU and GRU achieve relatively lower rejection statistics among the recurrent architectures, though they do not consistently pass all tests.

While the GRU models were more suitable for the Tadawul Exchange data, Fig. 9 shows that, in the case of RNN models (LSTM/BiLSTM), there is a noticeable gap between in-sample and backtesting performance, which suggests potential overfitting. This occurs because these neural models have high capacity, while our dataset is of moderate size. As a result, they may capture noise or idiosyncrasies in the training set rather than robust patterns. However, as an important direction for future work, we intend to investigate the applicability of the LSTM family to achieve better results on the Tadawul Exchange data. We also plan to apply overfitting mitigation techniques, such as dropout, weight regularization, early stopping, or synthetic data augmentation. For example, Agarwal et al. [51] demonstrates a CTGAN-based augmentation strategy that reduces overfitting in intrusion detection models for power systems, leading to improved adaptability against unseen attacks.

Taken together, these findings highlight that while recurrent structures provide stable and economically plausible VaR and ES estimates, particularly at moderate quantiles, their statistical adequacy in extreme tail regions lags behind well-specified heavy-tailed GARCH models. This is consistent with earlier studies that emphasize the superiority of fat-tailed conditional volatility models in risk management contexts (e.g., [52]). Nonetheless, the recurrent structures—especially BiGRU—show promise as complementary approaches, offering competitive performance in forecasting but requiring further refinement for extreme quantile calibration.

Subsequently, based on the given results, VaR, ES, and backtesting indicate that GARCH-family models effectively capture volatility dynamics in Tadawul exchange prices. Conversely, RNN variants demonstrate superior performance in modeling the nonlinear characteristics of clustered closing prices on the same exchange (see Fig. 10).

Table 8
Estimated parameters of the skewed-Student-t GARCH(1,1) model based on Tadawul exchange returns from the training dataset.

Parameter	Estimate	Std. error	t-value	$Pr(>\|r\|)$
mu	0.0006	0.0002	3.6418	0.0003
omega	−0.2251	0.0166	−13.581	0.000
alpha1	−0.1166	0.0254	−4.587	0.000
beta1	0.9723	0.0020	486.72	0.000
gamma1	0.2446	0.0487	5.0279	0.000
skew	0.9807	0.0187	52.515	0.000
shape	2.3487	0.1457	16.123	0.000

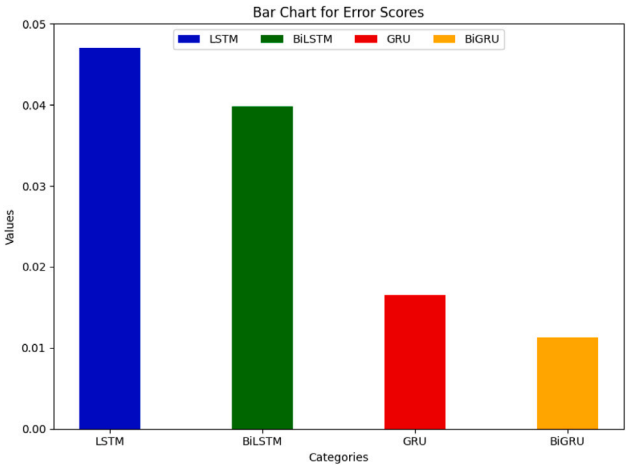

Fig. 10. Comparison of RNN model error scores, highlighting differences in forecasting accuracy.

Table 9
Comparison of forecasting accuracy between the proposed RNN networks and the GARCH model.

Metrics	GARCH	LSTM	GRU	BiLSTM	BiGRU
RMSE	0.0185	0.04640	0.01815	0.04417	0.01135
AIC	19 069.98	20 902	16 002	41 802	32 002
BIC	19 033.95	83 674.55	64 058.95	167 341.09	128 109.89

4. Conclusion

This study investigated the performance of statistical non-linear models, such as GARCH, and various deep learning architectures, including LSTM, BiLSTM, GRU, and BiGRU, in modeling volatility and forecasting market risk, with particular emphasis on VaR and ES. Overall, the results showed that several models improve forecasting accuracy and can serve as important references for future machine-learning applications in predicting stock market volatility.

The results demonstrated that while classical models such as GARCH remain competitive, especially when specified with heavy-tailed distributions, recurrent structures—most notably GRU and BiGRU—exhibit strong predictive accuracy with lower error measures. The GRU model provided the most favorable trade-off between accuracy and parsimony, whereas BiGRU delivered the most precise volatility forecasts, albeit with higher complexity.

In the context of risk forecasting, heavy-tailed GARCH specifications such as the skewed-t-EGARCH consistently outperformed Gaussian-based models, particularly in capturing extreme tail behavior. Backtesting further confirmed their adequacy at higher confidence levels, where recurrent models showed limitations. While RNN-based approaches yielded stable VaR and ES estimates aligned with historical simulation benchmarks, they struggled with extreme quantile calibration, often failing coverage tests at the 99% and 99.5% levels. Taken together,

Table 10
1-day ahead Value-at-Risk (VaR_α) expressed as a percentage under various significance levels.

Method	VaR_α				ES_α			
	0.1	0.05	0.01	0.005	0.1	0.05	0.01	0.005
HS	−1.151	−1.876	−4.476	−5.860	−2.532	−3.604	−6.723	−8.406
Gaussian	−1.708	−2.195	−3.107	−3.442	−2.342	−2.754	−3.561	−3.865
Cornish–Fisher	−0.394	−2.189	−8.815	−12.549	−1.369	−5.911	−8.815	−12.549
Monte Carlo simulations	−1.648	−2.146	−2.988	−3.341	−2.502	−2.968	−3.659	−3.816
Gaussian-EGARCH(1,1)	−1.780	−2.670	−5.510	−8.148	−2.326	−3.532	−7.598	−9.994
Skewed-Gaussian-EGARCH(1,1)	−1.806	−2.731	−5.677	−8.227	−2.338	−3.578	−7.741	−10.169
t-EGARCH(1,1)	−1.431	−2.398	−7.144	−11.849	−1.852	−3.150	−9.575	−14.320
Skewed-t-EGARCH(1,1)	−1.478	−2.480	−7.367	−12.220	−1.903	−3.265	−9.885	−14.790
GED-EGARCH(1,1)	−1.561	−3.462	−35.607	−77.694	−2.109	−5.122	−42.772	−85.980
LSTM	−1.168	−1.919	−4.328	−5.252	−2.450	−3.425	−6.220	−7.737
Bi-LSTM	−1.158	−1.883	−4.370	−5.233	−2.463	−3.460	−6.296	−7.848
GRU	−1.157	−1.855	−4.354	−5.420	−2.452	−3.459	−6.314	−7.889
Bi-GRU	−1.150	−1.862	−4.325	−5.567	−2.473	−3.498	−6.417	−8.015

Table 11
Backtesting results of the model's VaR predictions at 90% and 95% confidence levels, evaluating the accuracy of risk forecasts against observed returns.

Method	90% CL		95% CL	
	LR_{uc}	LR_{cc}	LR_{uc}	LR_{cc}
HS	3.1226(0.07)	9.9705(0.0068)[a]	11.6913(0.001)[a]	18.3602(0.00)[a]
Gaussian	51.2127(0.00)[a]	58.9097(0.000)[a]	18.2767(0.000)[a]	22.8784(0.00)[a]
Cornish–Fisher	181.0982(0.00)[a]	202.4017(0.000)[a]	18.2767(0.000)[a]	22.8784(0.00)[a]
Monte Carlo simulations	40.0867(0.00)[a]	45.1316(0.000)[a]	24.6259(0.000)[a]	26.4245(0.00)[a]
Gaussian-EGARCH(1,1)	Inf(0)[a]	Inf(0)[a]	0.097(0.755)	4.251(0.119)
Skewed-Gaussian-EGARCH(1,1)	24.236(0)[a]	32.758(0)[a]	3.397(0.065)	6.311(0.043)[a]
t-EGARCH(1,1)	NaN	NaN	7.52(0.01)[a]	15.468(0)[a]
Skewed-t-EGARCH(1,1)	NaN	NaN	1.344(0.246)	9.455(0.009)[a]
GED-EGARCH(1,1)	NaN	NaN	2.502(0.114)	18.765(0)[a]
LSTM	902.8770(0.00)[a]	923.3082(0.000)[a]	1427.0199(0.000)[a]	1447.451(0.00)[a]
BiLSTM	7.5774(0.00)[a]	7.5909(0.000)[a]	8.0479(0.000)[a]	8.0564(0.00)[a]
GRU	7.5704(0.00)[a]	7.5841(0.000)[a]	8.0421(0.000)[a]	8.0506(0.00)[a]
BiGRU	7.4690(0.00)[a]	7.4762(0.000)[a]	7.9583(0.000)[a]	7.9627(0.00)[a]

[a] Reject H_0.

Table 12
Backtesting results of the model's VaR forecasts at 99% and 99.5% confidence levels, assessing the accuracy of extreme risk predictions against observed returns.

Method	99% CL		99.5% CL	
	LR_{uc}	LR_{cc}	LR_{uc}	LR_{cc}
HS	5.7684(0.0163)	5.7791(0.0556)	3.1226(0.0772)	9.9705(0.0068)[a]
Gaussian	3.5416(0.0598)	3.5657(0.1682)	51.2127(0.000)[a]	58.9097(0.000)[a]
Cornish–Fisher	NaN	NaN	NaN	NaN
Monte Carlo simulations	3.5416(0.0599)	3.5657(0.1682)	0.1639(0.685585)	0.1880(0.9103)
Gaussian-EGARCH(1,1)	20.117(0)[a]	22.623(0)[a]	33.161(0)[a]	34.341(0)[a]
Skewed-Gaussian-EGARCH(1,1)	10.442(0.001)[a]	14.371(0.001)[a]	23.052(0)[a]	23.954(0)[a]
t-EGARCH(1,1)	1.439(0.23)	2.29(0.318)	0.235(0.628)	0.34(0.843)
Skewed-t-EGARCH(1,1)	0.43(0.512)	1.137(0.567)	1.139(0.286)	1.212(0.546)
GED-EGARCH(1,1)	0.085(0.77)	0.582(0.748)	0.11(0.74)	0.275(0.872)
LSTM	2702.2069(0.000)[a]	2722.6381(0.000)[a]	3260.1801(0.000)[a]	3280.6113(0.000)[a]
BiLSTM	8.6985(0.000)[a]	8.7029(0.000)[a]	8.8887(0.000)[a]	8.8924(0.000)[a]
GRU	8.6938(0.000)[a]	8.6982(0.000)[a]	8.8843(0.000)[a]	8.8879(0.000)[a]
BiGRU	8.6265(0.000)[a]	8.6287(0.000)[a]	8.8203(0.000)[a]	8.8221(0.000)[a]

[a] Reject H_0.

the findings underscore the complementary strengths of the two modeling paradigms. Traditional heavy-tailed GARCH models remain the benchmark for regulatory risk management due to their superior tail calibration, while recurrent structures offer valuable improvements in forecasting accuracy and flexibility for practical applications.

Nevertheless, several limitations should be acknowledged. Although the dataset covers 17 years, the effective sample size for training recurrent models is relatively modest, which may constrain generalizability, particularly for broader GCC markets. Moreover, signs of overfitting in LSTM and BiLSTM highlight the need for further methodological refinement. Future research should consider hybrid approaches that combine the statistical rigor of GARCH with the predictive flexibility of recurrent networks, integrate sentiment and macroeconomic indicators, validate models across different markets, and apply regularization strategies such as dropout, early stopping, or ensemble methods. These extensions would strengthen the robustness and practical relevance of risk forecasting in emerging financial markets.

CRediT authorship contribution statement

Abdaljbbar B.A. Dawod: Writing – original draft, Software, Methodology, Conceptualization. **Mohammed Nsaif:** Software, Methodology,

Conceptualization. **Dhafer G. Honi:** Review & editing, Methodology, Conceptualization. **Ismail H. Abdi:** Review & editing, Methodology, Conceptualization. **Husam A. Neamah:** Writing – review & editing, Supervision, Resources, Methodology, Formal analysis, Conceptualization.

Funding

The authors wish to acknowledge the financial support provided by the University of Debrecen Research Fund.

Declaration of competing interest

The authors have no relevant financial or non-financial interests to disclose. We confirm that this work is original and has not been published elsewhere, nor is it currently under consideration for publication elsewhere.

Data availability

The datasets used in this study are publicly available through the Tadawul stock exchange database.

References

[1] B.A.F. Bankenaufsicht, Revision to the basel ii: market risk framework, 2009, Internet: http://www.bis.org/publ/bcbs158. pdf, Stand 29, 2012.

[2] P. Boozary, S. Sheykhan, H. GhorbanTanhaei, Forecasting the bitcoin price using the various machine learning: A systematic review in data-driven marketing, Syst. Soft Comput. 7 (2025) 200209, http://dx.doi.org/10.1016/j.sasc.2025.200209, URL: https://www.sciencedirect.com/science/article/pii/S2772941925000274.

[3] C. Alexander, Market Risk Analysis, Value At Risk Models, vol. 4, John Wiley & Sons, 2008.

[4] A.B. Dawod, Z.M. Mohammed, Enhancing the accuracy of the historical simulation value-at-risk using exponentially weighted moving average (ewma) technique: A case of Khartoum stock exchange, Khartoum Univ. J. Manag. Stud. (2019) http://dx.doi.org/10.5897/KUJMS2018.0006, URL: https://www.semanticscholar.org/paper/15bd5d8d5501d7ad4a3f44fa51348ef41b87dcb9.

[5] T. Bollerslev, Generalized autoregressive conditional heteroskedasticity, J. Econometrics 31 (1986) 307–327.

[6] M. Vijh, D. Chandola, V.A. Tikkiwal, A. Kumar, Stock closing price prediction using machine learning techniques, Procedia Comput. Sci. 167 (2020) 599–606, http://dx.doi.org/10.1016/j.procs.2020.03.326, URL: https://www.sciencedirect.com/science/article/pii/S1877050920307924. international Conference on Computational Intelligence and Data Science.

[7] M.A.I. Sunny, M.M.S. Maswood, A.G. Alharbi, Deep learning-based stock price prediction using lstm and bi-directional lstm model, in: 2020 2nd Novel Intelligent and Leading Emerging Sciences Conference, NILES, Vol. 8, IEEE, 2020, pp. 7–92.

[8] J.J. Huang, K.J. Lee, H. Liang, W.F. Lin, Estimating value at risk of portfolio by conditional copula-garch method, Insurance Math. Econom. 45 (2009) 315–324, http://dx.doi.org/10.1016/j.insmatheco.2009.09.009, URL: https://www.sciencedirect.com/science/article/pii/S0167668709001267.

[9] S.N. Abdullah, Modeling the volatility of stock prices for the saudi stock market general index (TASI), Period. Eng. Nat. Sci. 8 (2020) 1981–1990.

[10] Y. Luan, S. Ye, Y. Li, L. Jia, X.G. Yue, Revisiting natural resources volatility via tgarch and egarch, Resour. Policy 78 (2022) 102896, http://dx.doi.org/10.1016/j.resourpol.2022.102896, URL: https://www.sciencedirect.com/science/article/pii/S0301420722003415.

[11] M. Al-Momani, A.B.A. Dawod, Model selection and post selection to improve the estimation of the arch model, J. Risk Financ. Manag. 15 (2022) http://dx.doi.org/10.3390/jrfm15040174, URL: https://www.mdpi.com/1911-8074/15/4/174.

[12] C. Kosapattarapim, Y.X. Lin, M. McCrae, Evaluating the volatility forecasting performance of best fitting garch models in emerging asian stock markets, Int. J. Math. Stat. 12 (2012) 1–15.

[13] P.R. Hansen, A. Lunde, A forecast comparison of volatility models: does anything beat a garch (1, 1)? J. Appl. Econometrics 20 (2005) 873–889.

[14] M. Usmani, S.H. Adil, K. Raza, S.S.A. Ali, Stock market prediction using machine learning techniques, in: 2016 3rd International Conference on Computer and Information Sciences, ICCOINS, IEEE, 2016, pp. 322–327.

[15] A. Shah, M. Gor, M. Sagar, M. Shah, A stock market trading framework based on deep learning architectures, Multimedia Tools Appl. 81 (2022) 14153–14171.

[16] D.L. Minh, A. Sadeghi-Niaraki, H.D. Huy, K. Min, H. Moon, Deep learning approach for short-term stock trends prediction based on two-stream gated recurrent unit network, IEEE Access 6 (2018) 55392–55404.

[17] P. Jaquart, D. Dann, C. Weinhardt, Short-term bitcoin market prediction via machine learning, J. Financ. Data Sci. 7 (2021) 45–66, http://dx.doi.org/10.1016/j.jfds.2021.03.001, URL: https://www.sciencedirect.com/science/article/pii/S2405918821000027.

[18] J. Li, J. Kang, J. Wu, H. Wang, X. Yang, Research on credit card default repayment prediction model, J. Financ. Data Sci. 10 (2024) 100136, http://dx.doi.org/10.1016/j.jfds.2024.100136, URL: https://www.sciencedirect.com/science/article/pii/S2405918824000217.

[19] E. Dave, A. Leonardo, M. Jeanice, N. Hanafiah, Forecasting indonesia exports using a hybrid model ARIMA-LSTM, Procedia Comput. Sci. 179 (2021) 480–487.

[20] S. Siami-Namini, N. Tavakoli, A.S. Namin, A comparative analysis of forecasting financial time series using ARIMA, LSTM, and BiLSTM, 2019, arXiv preprint arXiv:1911.09512.

[21] G.M. BinMakhashen, A.A. Bakather, A.A. Bin-Salem, An investigation of forecasting tadawul all share index (TASI) using machine learning, in: 2022 7th International Conference on Data Science and Machine Learning Applications, CDMA, IEEE, 2022, pp. 19–24.

[22] T.S. Alshammari, M.T. Ismail, A.W. Sadam, M.H. Saleh, J.J. Jaber, Modeling and forecasting saudi stock market volatility using wavelet methods, J. Asian Financ. Econ. Bus. 7 (2020) 83–93.

[23] Tsay, Analysis of Financial Time Series, vol. 48, John wiley & sons, 2006, http://dx.doi.org/10.1198/tech.2006.s405, URL: http://pubs.amstat.org/doi/abs/10.1198/tech.2006.s405.

[24] I. Goodfellow, Y. Bengio, A. Courville, Y. Bengio, Deep Learning, vol. 2, MIT press Cambridge, 2016.

[25] Organization of the petroleum exporting countries, Annu. Stat. Bull. (2020) 10, URL: https://asb.opec.org/, download Pdf, (Accessed 24 July 2024).

[26] S. Exchange, Annual reports - Saudi exchange, 2024, URL: https://www.saudiexchange.sa/wps/portal/saudiexchange/news{andr}eports/reports-publications/annual-reports?locale=en. (Accessed 24 July 2024).

[27] Saudi Arabian Monetary Authority, Historical preview, 2024, URL: http://www.sama.gov.sa/en-US/About/Pages/SAMAHistory.aspx. (Accessed 3 November 2025).

[28] Yahoo Finance, Tadawul all shares index (ٱtasi.sr) - yahoo finance, 2024, URL: https://finance.yahoo.com/quote/%5ETASI.SR/history/. (Accessed 24 July 2024).

[29] P. Jorion, Value At Risk: The New Benchmark for Managing Financial Risk, McGraw-Hill, 2007.

[30] C. Pérignon, D.R. Smith, Yield-factor volatility models, J. Bank. Financ. 31 (2007) 3125–3144, http://dx.doi.org/10.1016/j.jbankfin.2006.11.016, URL: https://www.sciencedirect.com/science/article/pii/S0378426607000453.

[31] R.F. Engle, Autoregressive conditional heteroscedasticity with estimates of the variance of United Kingdom inflation, Econ.: J. Econ. Soc. (1982) 987–1007.

[32] D.B. Nelson, Conditional heteroskedasticity in asset returns: A new approach, Econometrica 59 (1991) 347–370, URL: http://www.jstor.org/stable/2938260.

[33] L.R. Glosten, R. Jagannathan, R.D. E, On the relation between the expected value and the volatility of the nominal excess return on stocks, J. Financ. 48 (1993) 1779–1801, http://dx.doi.org/10.1111/j.1540-6261.1993.tb05128.x, URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1540-6261.1993.tb05128.x. https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1540-6261.1993.tb05128.xarXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1540-6261.1993.tb05128.x.

[34] J.M. Zakoian, Threshold heteroskedastic models, J. Econom. Dynam. Control 18 (1994) 931–955.

[35] N.P. Guglielmo Maria Caporale, N. Spagnolo, Igarch models and structural breaks, Appl. Econ. Lett. 10 (2003) 765–768, http://dx.doi.org/10.1080/1350485032000138403, https://doi.org/10.1080/1350485032000138403arXiv:https://doi.org/10.1080/1350485032000138403.

[36] R.H. Shumway, D.S. Stoffer, Time Series Analysis and Its Applications: With R Examples, Springer Texts in Statistics, 2011, http://dx.doi.org/10.1007/978-1-4419-7865-3.

[37] X. Zhang, X. Liang, A. Zhiyuli, S. Zhang, R. Xu, B. Wu, At-LSTM: An attention-based LSTM model for financial time series prediction, in: IOP Conference Series: Materials Science and Engineering, IOP Publishing, 2019, 052037.

[38] B. Yan, M. Aasma, et al., A novel deep learning framework: Prediction and analysis of financial time series using ceemd and LSTM, Expert Syst. Appl. 159 (2020) 113609.

[39] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput. 9 (1997) 1735–1780.

[40] M. Nsaif, G. Kovásznai, A. Malik, R. de Fréin, SM-FPLF: Link-state prediction for software-defined dcn power optimization, IEEE Access 12 (2024) 79496–79518, http://dx.doi.org/10.1109/ACCESS.2024.3408672.

[41] I. Sutskever, O. Vinyals, Q.V. Le, Sequence to sequence learning with neural networks, Adv. Neural Inf. Process. Syst. 27 (2014).

[42] M. Schuster, K.K. Paliwal, Bidirectional recurrent neural networks, IEEE Trans. Signal Process. 45 (1997) 2673–2681.

[43] Z. Hameed, B. Garcia-Zapirain, Sentiment classification using a single-layered bilstm model, IEEE Access 8 (2020) 73992–74001.
[44] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, Y. Bengio, Learning phrase representations using RNN encoder–decoder for statistical machine translation, 2014, arXiv preprint arXiv:1406.1078.
[45] G. Weiss, Y. Goldberg, E. Yahav, On the practical computational power of finite precision rnns for language recognition, 2018, arXiv preprint arXiv:1805.04908.
[46] P. Kupiec, Techniques for verifying the accuracy of risk measurement models, J. Deriv 2 (1995) 173–184.
[47] P.F. Christoffersen, Evaluating interval forecasts, Internat. Econom. Rev. 39 (1998) 841–862, URL: http://www.jstor.org/stable/2527341.
[48] G. Shen, Q. Tan, H. Zhang, P. Zeng, J. Xu, Deep learning with gated recurrent unit networks for financial sequence predictions, Procedia Comput. Sci. 131 (2018) 895–903, http://dx.doi.org/10.1016/j.procs.2018.04.298, URL: https://www.sciencedirect.com/science/article/pii/S1877050918306781. recent Advancement in Information and Communication Technology:.
[49] C. Umezuruike, D. Olaniyan, J. Olaniyan, A.E. Adeniyi, A. Oyebade, D. Abaneme, Comparative analysis of long short-term memory (lstm), gated recurrent unit (GRU) and transformer models in predicting stock prices, in: 2024 IEEE 5th International Conference on Electro-Computing Technologies for Humanity, NIGERCON, 2024, pp. 1–6, http://dx.doi.org/10.1109/NIGERCON62786.2024.10927198.
[50] Z.M. Shaikh, S. Ramadass, Unveiling deep learning powers: Lstm, BiLSTM, GRU, biGRU, RNN comparison, Indones. J. Electr. Eng. Comput. Sci. 35 (2024) 263–273.
[51] L. Agarwal, B. Jaint, A.K. Mandpura, Reducing overfitting in deep learning intrusion detection for power systems with CTGAN, Chaos Solitons Fractals 188 (2024) 115603.
[52] A.J. McNeil, R. Frey, P. Embrechts, Quantitative Risk Management: Concepts, Techniques and Tools-Revised Edition, Princeton University Press, 2015.