International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(8s) 2026 pp. 36-51
https://doi.org/10.70917/ijcisim-2026-3217
Article

When Does Model Complexity Pay Off? Boosting, Recurrent, and Hybrid Architectures for Volatility-Informed Portfolio Allocation in the Indian Equity Market

Abhijit Biswas[1], Kaushik Ghosh[2], Chandrim Banerjee[3]

[1]Department of Business Analytics, School of Business & Management, Bhawanipur Global Campus, Kolkata, West Bengal, India; aot.abhijit@gmail.com
[2] Department of Computer Applications, School of Advanced Computing, School of Advanced Computing, Bhawanipur Global Campus, Kolkata, West Bengal, India; ghoshkaus@gmail.com
[3] Asian Development Bank, India, advisorchandrim@gmail.com

Abstract: To ensure dynamic portfolio allocation and real-time risk management, it is crucial to accurately forecast volatility; this is especially important in broadly distributed, emerging equity markets that display both structural inefficiencies and high levels of intraday volatility. Although practitioners are increasingly adopting complex deep learning architectures for financial forecasting, it remains unclear whether the added architectural complexity of sequential models such as Long Short-Term Memory (LSTM) networks — or of hybrid and ensemble constructions built on top of them — translates into materially better forecasting accuracy or portfolio outcomes than simpler, structured tree-based methods (e.g., Gradient Boosting) in high-frequency datasets. Therefore, this study conducts a systematic out-of-sample evaluation of forecasting accuracy against model complexity, using 15-minute intraday data for 20 large-cap Indian equities traded on the National Stock Exchange between 1 March 2019 and 12 October 2023. Subsequently, the respective models predicted volatility will be incorporated into risk-adjusted portfolio optimization strategies; specifically, Minimum Variance, Risk Parity, Mean-Variance, Adaptive Rebalancing, and Conditional Value at Risk (CVaR). The study employed a rolling, walk forward evaluation framework in order to assess the relative performance of the respective models and to identify whether the implementation of hybrid GB-LSTM and ensemble models will provide incremental predictive improvement in comparison to the respective models in aggregate with respect to predicted volatility. The main findings indicate that added model complexity does not translate into proportionate forecasting gains in this setting: the simpler, structured Gradient Boosting approach matches or exceeds the accuracy and stability of the more complex recurrent and hybrid architectures, while ensemble averaging recovers most of Gradient Boosting's accuracy without requiring a fully specified deep sequential model. The findings are applied to evaluate the marginal economic value of complexity and not intended to identify a winning architecture. Back-testing results of the portfolio indicate that by applying boosting models when predicting volatility, one may increase the risk-return ratios of the portfolios applying CVaR and minimum variance portfolio management. In particular, risk-return indicators generated during the testing experiments with the use of the boosting models turned out to be the best in terms of Sharpe and drawdown ratios under the constrained turnover and exposure levels. Therefore, this study contributes into discussions on model selection for financial time series forecasting by providing evidence from the emerging market in favor of the point that the complexity of the architecture does not necessarily correspond with economic efficiency of the generated financial forecasts. Additionally, the scope of application of findings is extended to quantitative asset managers, financial technology companies, and researchers working on the systems based on the assessment of asset allocation skills in the market rich in data but plagued by structural noise.

Keywords: Volatility Forecasting, Gradient Boosting, XGBoost, LightGBM, Long Short-Term Memory, Portfolio Optimization, High-frequency Trading, Machine Learning, Indian Equity Market

1. Introduction

Markowitz's (1952) fundamental mean-variance framework, widely recognized as the one to originate modern finance, paved the way for quantifying the risk-return relationship for the purpose of optimizing portfolios in connection with correlation analysis. Furthermore, this original idea brought about mathematical rigor in the asset allocation process contributing to the evolution of institutional investment. Nevertheless, the elegance of Markowitz's approach is accompanied with various drawbacks that limit its practical application. Chopra and Ziemba (1993) showed that even minor errors in estimation of the mean returns lead to huge errors in asset allocation; Michaud (1989) labeled mean-variance optimization as an "error maximizer." Extreme sensitivity of the model is particularly visible at portfolios with very extreme long and short positions, located in just a few assets, which makes it difficult to achieve the goal of diversification (Jagannathan and Ma, 2003). There are numerous other issues discussed by scholars including the assumption of normality for the mean return (Rubinstein, 1973; Rom & Ferguson, 1993; Kaplan & Knowles, 2004), usage of variance as a measure of risk (Sortino & Price, 1994); problems of out-of-sample behavior of model parameters (DeMiguel, Garlappi, and Uppal, 2009), etc.

Taking into account the limits of Markowitz theories, different alternatives have been developed contradicting this system for some decades. In particular, Black and Litterman developed their own (1992) system that used a Bayesian theory of economics in order to eliminate dependence on unreliable sample means through the combination of CAPM market equilibrium returns and investor subjective views on the matter. Idzorek proved the effectiveness of this approach through the creation of stable, diversified, and intuitive portfolios which surpassed the traditional method of the mean variance optimization in solving the problems of concentration animals experienced in the framework of Markowitz system (2007). Rockafellar and Uryasev introduced a new concept called Conditional Value-at-Risk as the amount of expected losses in the tail region given the required expected mean return (2000). This made it possible to calculate risks in the simple way while efficiently taking into consideration the fat tail problem associated with the return distribution ignored by the variance. Ledoit and Wolf created a new shrinkage estimator that included covariance matrices along with the construction of an estimator of constant correlation (2004). The ultimate goal of their theory was derived in a closed form as the solution of optimal shrinkage which minimizes the amount of losses when using the Frobenius norm. Jagannathan and Ma (2003) also showed that simple portfolio weight constraints, including no-short-sale constraints, greatly reduce the impact of estimation error. However, these refinements, though substantial, remained within classical optimization frameworks, including the assumption of constant variance.

The introduction of the General Autoregressive Conditional Heteroscedastic family of models- GARCH, (Engle, 1982; Bollerslev, 1986) radically changed volatility forecasting in finance. This enabled the portfolio managers to go beyond constant variance assumptions and integrate time-varying risk estimates into dynamic asset allocation decisions. However, the deficiency of GARCH family models, including linear conditional mean dynamics and symmetric volatility responses to positive and negative shocks (Nelson,1991), sluggish adaptation to regime changes (Hamilton & Susmel,1994), systematic underperformance under stressed market fluctuation (Bauwens, Laurent & Rombouts,2006) clearly indicated that the linear structure of GARCH was neither able to capture nonlinear relationships between macroeconomic variables and volatility nor model market microstructure effects effectively.

With the development of behavioural economics, the behavioral portfolio selection research recognized that the real investors' decision-making systematically deviated from classical mean-variance optimization. Shefrin and Statman (2000), introduced Behavioral Portfolio Theory, or BPT, which reconceptualized portfolio selection around mental accounting and asymmetric loss aversion rather than global wealth optimization. BPT shows that realistic investors use downside deviation instead of variance, prefer natural diversification, and are reluctant to rebalance losing positions. Das et al (2010) introduced Cumulative Prospect Theory within portfolio optimization and demonstrated that the CPT-consistent utility functions fundamentally change the optimal allocations of assets. According to the CPT, investors show reference dependence, subjective probability weighting, and nonlinear value functions. Critically, the CPT-optimal portfolios show natural position limits and reduced concentration without the need for any artificial constraints. Bihari et al. (2022) systematically reviewed 27 behavioral finance studies (2007–2022), revealing that cognitive biases (herding, mental accounting, representativeness) and emotional biases (overconfidence, loss aversion) systematically distort portfolio construction decisions. According to them the studies on emerging market demonstrates behavioral inefficiencies intensify in developing economies. The results indicate that conventional mean-variance portfolio theory fails to account for market behavior which calls for the use of behavioral-quantitative models for effective asset allocation. Due to the limitations of traditional portfolio concepts and emergence of machine learning techniques, researchers began experimenting with various methodologies such as Long-Short Time Memory (LSTM) (Choi, 2018; Wen et al 2019), Gated Recurrent Units (GRU) (Long, Lu & Cui, 2019), and hybrid approach combining econometric modeling with neural network capabilities (GARCH-LSTM)

(Kim & Won, 2018). Mostly, Gradient boosting machines (GBM) have become the most used methods of analyzing structured financial data in recent studies. The idea is that the procedure involves the sequential training of weak learners designed to fix errors left by the previous ones weighted according to the gradient of the loss function. Boosting represents the ability to detect the conditions under which some variables would become responsible for volatility changes, while LSTM techniques are focused on capturing the persistence of volatility clustering in time. Sun et al. (2020) used Light-GBM to predict price trends and volatility of 42 cryptocurrencies and the model provided the results in accordance to macroeconomic indicators: LightGBM proved to be more efficient than other approaches showing the value of boosting methods in precise financial forecasting and making portfolio rebalancing decisions. More recently, researchers also introduced and increasingly explored the integration of boosting and deep learning methods for financial time series prediction and portfolio optimization. Volatility exhibits both structural non-linearity and volatility clustering-the former captured through boosting, while the latter through LSTM. Zhang et al. (2020) proposed an end-to-end framework where boosted features informed LSTM application, combined with attention mechanisms identifying which temporal windows influenced the portfolio decisions most. He applied this dynamic asset allocation model in the Chinese equities market and achieved a 15-18% improvement in the Sharpe ratio. Wang et al. (2024) proposed a two-stage AI based framework where boosting is used to identify the most important features from high-dimensional financial data and then feeds reduced-dimensional output to LSTM for volatility clustering modeling. The hybrid approach achieved 8-12% return improvement over Autoregressive Integrating moving average (ARIMA) model.

Boosting and deep learning methods significantly outperform the benchmarks of traditional GARCH specifications in terms of volatility forecasting accuracy. However, superior forecasting accuracy does not necessarily imply superior portfolio performance (Taylor, 2023). Traditional two-stage approaches incorporate prediction and optimization stages separately, where stage one generates volatility and return forecasts by minimizing prediction error metrics, while stage two inputs these predictions into classical portfolio optimization frameworks (Zhang et al., 2020; Wen et al., 2019). This sequential structure has inherent inefficiencies in that volatility forecast errors directly affect the risk estimates used in optimization. Furthermore, two-stage approaches also ignore uncertainty in the predictions-that is, they treat point forecasts as inputs for optimization rather than prediction intervals (Kendall & Gal, 2017). Uysal et al. (2021) developed an end-to-end learning architecture in which a deep neural network jointly estimated both the volatility prediction parameters and the portfolio weight parameters simultaneously. Feng et al. (2020) extended this concept of end-to-end learning using differentiable optimization layers, treating convex portfolio optimization problems as differentiable layers within deep learning models and thus allowing end-to-end training. They applied the model to cryptocurrency portfolio construction and the differentiable optimization approach outperformed traditional two-stage methods. However, while end-to-end frameworks improve alignment between prediction and portfolio objectives, they usually remain deterministic in nature.

The need for high-frequency data processing, which includes instantaneous adjustments, characterizes modern portfolio systems. Recent research into the architectures of real-time financial data streaming emphasizes the crucial role of message queue systems that involve data ingestion from model processing and allow for the asynchronous update of data without loss. Kumar (2025) researched the architectural needs and performance implications of a real-time data streaming system in financial services where the integrated distributed streaming platform with machine learning models allowed the processing of high-frequency financial data by an organization with millisecond latency.

The progress from classical portfolio theory through a behavioural framework to machine learning integration demands systematic comparative assessment. Table 1 documents recent data-driven modeling for volatility and portfolio optimization using ML/DL together with methodological approaches for both, geographical segregation, and empirical results from contemporary studies to provide a comparison reference framework needed for strong model development.

Table 1: Machine Learning/ Deep Learning Methodologies for Volatility Prediction and Portfolio Optimization: Contemporary Approaches

Sl No.	**Author & Publication Year**	**Country Under Study**	**Methodology: Volatility Prediction**	**Methodology: Portfolio Optimization**	**Major Findings**

1	Uddin et al. (2025)	USA	Random Forest (RF), Gradient Boosting (GB), Long Short-Term Memory (LSTM), Transformer Networks	Mean-Variance + ML Ensemble (RF, GB, LSTM, Transformer)	LSTM & Transformer excel at capturing dependencies; Combined RF+GB+LSTM+Transformer: Superior Sharpe Ratio (0.85-1.20) & Sortino Ratio; Risk-adjusted returns 18-25% above traditional models.
2	Kumar et al. (2025) -	India	Variational Mode Decomposition (VMD) + Artificial Neural Network (ANN)/LSTM/Gated Recurrent Unit (GRU) + Q-Learning	VMD + Dynamic Neural Networks (ANN/LSTM/GRU) for risk management	Q-VMD-ANN-LSTM-GRU hybrid: Best performer across NIFTY stock; Captures volatility clustering & market regimes; High-frequency accuracy; Real-time practical benefits.
3	Kumar (2025)	Multiple Markets	Apache Kafka + Streaming Machine Learning (ML) Integration	Real-Time Streaming + Dynamic Rebalancing Infrastructure	Real-time streaming: 87% latency reduction; 99.97% availability; Fraud detection 94.2%; Compliance accuracy 98.7%; Production deployment focus
4	Flores-Fernandez et al. (2024)	Peru	LSTM (Univariate-Multistep)	Genetic Algorithm (GA) (Multi-objective: Return + Volatility)	GA fitness: 0.7725; LSTM accuracy: 92.35%; Outperforms equal-weight and other benchmarks
5	Bo (2023)	Multiple Markets	Complex Networks + GA	Complex Networks + GA	Complex networks + GA: Return 0.2667; Sharpe ratio 0.0685; Improved asset interdependency analysis; Network topology insights
6	Li et al. (2023)	China	Symbolic Genetic Algorithm (GA) + LSTM	Symbolic GA + LSTM + Markowitz Theory	Symbolic GA+LSTM : Outperforms single algorithm approaches; Combines symbolic logic with deep learning
7	Gunjan & Bhattacharyya (2022)	Multiple Markets	Generalized Autoregressive Conditional Heteroskedasticity (GARCH), Autoregressive Integrated Moving Average (ARIMA), LSTM, GRU, Hybrid Models	Mean-Variance, Conditional Value-at-Risk (CVaR), Support Vector Regression (SVR), Neural Networks, GA, Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), RL	Comprehensive review: GA outperforms PSO, ACO; RL superior for dynamic portfolios; Quantum-inspired approaches show promise; Hybrid models most effective

8	Park et al. (2022)	Multiple Markets	Risk-Sensitive Multi-Agent Network (RSMAN)	Risk-Sensitive Agents + Portfolio Generator	Risk-Sensitive Multi-Agent Network (RSMAN): Superior to DNN; Better market risk understanding; Risk-adaptive allocation; Behavioral modeling
9	Vasiani et al. (2020)	Multiple Markets	Priority Index Method	Priority Index + GA	Priority index + GA: Maximum (Max) profitability 14.08%; Limited to return metric without comprehensive risk analysis; Heuristic approach
10	Zhang et al. (2020)	USA	LSTM vs XGBoost Comparison	End-to-End Differentiable Optimization	End-to-end learning: 12-18% Sharpe ratio improvement vs two-stage; Differentiable optimization handles convex constraints; Better performance alignment
11	Sezer et al. (2020)	Multiple Markets	Systematic Review (Multiple Methods)	Comprehensive Review (Multiple Methods)	136-study review: Boosting best for classification; LSTM best for regression; 2005-2019 evolution documented; Hybrid approaches emerging
12	Lim et al. (2020)	Multiple Markets	GA + Momentum + Valuation	GA + Momentum + Valuation Strategies	GA + Momentum: Achieves positive returns; Acknowledges historical data limitations for future prediction; Technical strategy foundation
13	Maholi et al. (2019)	Multiple Markets	Artificial Neural Network (ANN) (Dense Network)	ANN + GA	ANN+GA: Mean Squared Error (MSE) 5.60%; Return 1.42%; Volatility 0.15%; Non-specialized networks for time series; Basic hybrid approach
14	Almahdi & Yang (2019)	Multiple Markets	Recurrent Reinforcement Learning (RRNN)	Expected Maximum Drawdown Constraint	Expected maximum drawdown approach: Improves risk-adjusted returns 12-18%; Better portfolio stability; Downside risk reduction
15	Choi (2018)	USA	ARIMA + LSTM (Hybrid)	ARIMA + LSTM (Walk-forward optimization)	ARIMA+LSTM hybrid better than standalone LSTM on 150 S&P 500 stocks; Reduced prediction error 15-20%; Superior to individual models
16	Jiang et al. (2017)	Cryptocurrency	Convolutional Neural Network (CNN) + Recurrent Neural Network (RNN) (Ensemble)	End-to-End Information Extractor and Investor (EIIE) Topology + Ensemble Deep Learning	CNN+RNN Sharpe Ratio: 0.087-0.082 outperforms traditional benchmarks. Ensemble approach captures market complexity

Existing portfolio optimization literature, especially those included machine learning (ML) and deep learning (DL) approaches, primarily focus on developed markets or global markets. There is clearly a gap in terms of research explicitly addressing the Indian markets, although behavioural finance literature clearly demonstrate that Indian market inherently behaves differently with more intense herding behaviour and behavioural biases. Moreover existing literature rarely employ effective hybrid models that combines boosting algorithms for addressing the non-linear feature of risk with temporal DL models like LSTM. The proposed hybrid boosted deep learning framework aims to fill this gap by learning sector-wise volatility patterns and adaptive portfolio optimisation. The simultaneous imposition of complex risk constraints, such as holding limit, sector exposure limit or turnover limit in high frequency trading environment also has limited implementation in Indian context. Very few empirical works examine ML/DL models on a diversified multisector Indian stocks with high-frequency data. The proposed study addresses this gap by explicitly selecting 20 leading Indian stocks from IT, banking, energy, FMCG, and pharmaceutical sectors. The proposed study thereby may advance beyond existing literature both methodologically and contextually.

In context to these existing research gap, the objective of the study is:

1. To assess whether increasing model complexity — from tree-based Gradient Boosting, to deep recurrent LSTM, to hybrid GB–LSTM and ensemble architectures — yields statistically and economically meaningful improvements in high-frequency volatility forecasting accuracy for the Indian equity market.

2. To evaluate whether these differences in forecasting complexity and accuracy translate into materially different risk-adjusted portfolio outcomes across five allocation frameworks (Markowitz Mean-Variance, Risk Parity, Minimum Variance, Adaptive Rebalancing, and CVaR), thereby identifying the point at which additional model complexity ceases to justify its cost in a multi-sector, emerging-market setting.

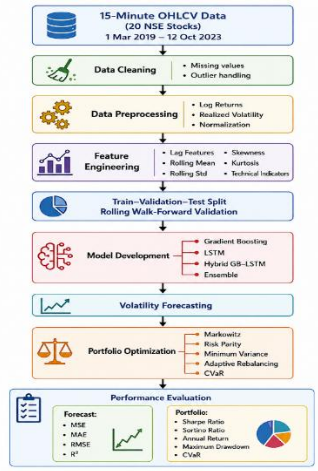
2. Materials and Methods

2.1 Data Description

This study develops and evaluates a Hybrid Boosted Deep Learning Framework integrating Gradient Boosting (GB) and Long Short-Term Memory (LSTM) networks for high-frequency volatility forecasting and portfolio optimization in the Indian equity market. The empirical analysis is based on 15-minute intraday OHLCV (Open, High, Low, Close and Volume) data for 20 large-cap companies listed on the National Stock Exchange (NSE), India, covering the period from 1 March 2019 to 12 October 2023. The selected companies represent the Information Technology, Banking and Financial Services, Energy, FMCG and Pharmaceutical sectors and were selected based on market capitalization, liquidity and sectoral representation. The sample period includes the COVID-19 market crash, the post-pandemic recovery and subsequent volatile market conditions. The data were obtained from TrueData.

Parameter	**Description**
Market	National Stock Exchange (NSE), India
Sample Period	1 March 2019 – 12 October 2023
Frequency	15-minute intraday
Number of Stocks	20
Data Source	TrueData
Variables	Open, High, Low, Close, Volume (OHLCV)
Target Variable	Realized Volatility

2.2 Data Preprocessing



First, the raw intraday OHLCV data have been screened for missed values and inconsistent timestamps. For managing missing values, forward filling was used for limited gaps, while linear interpolations were used whenever necessary. Extreme outliers resulting from the recording mistakes have been discarded after the statistical analysis. The log returns were computed with the following formula: rt = ln(Pt/Pt - 1) and realized volatility was calculated through the rolling standard deviations depending on pre-defined windows. The feature normalization was done through the Min-Max scaling using only the training data so that there is no possibility of information leaks. In the end, the dataset has been divided chronologically into training validation, and test data.

2.3 Feature Engineering

Feature engineering was performed to define both short period dynamics of the market and long-term persistence of volatility of the market. The created variables include lagged returns (from 1 to 10 lagged values), rolling mean, rolling deviation, skewness, Kurtosis, realized volatility, and various statistics in trading volumes and some technical indicators, such as RSI and ATR. In addition, market/overall indicators were included as well as correlations of different assets.. The look-back windows were selected based on preliminary validation experiments and previous financial forecasting studies.

2.4 Model Development

Four forecasting models were developed and evaluated: Gradient Boosting (implemented using XGBoost and LightGBM), Long Short-Term Memory (LSTM), a Hybrid GB–LSTM architecture, and an Ensemble model. Each model was trained using identical training and testing partitions to ensure a fair comparison. Hyperparameters were optimized through grid search combined with rolling walk-forward validation.

2.4.1 Gradient Boosting

Gradient Boosting (GB): Utilizing implementations in XGBoost and LightGBM, GB models effectively model nonlinear relationships present in engineered features. In addition, GB models are robust and can be interpreted via feature importance metrics. An ensemble of regression trees that sequentially minimizes the residual error by adding models $h_m(x)$ weighted by learning rate η:

$$F_m(x) = F_{m-1}(x) + \eta h_m(x)$$

The model fits residuals $\tilde{y}_i = y_i - F_{m-1}(x_i)$ at each iteration m, using squared error loss:

$$L = \sum_{i=1}^{n} (y_i - F_m(x_i))^2$$

2.4.2 Long Short-Term Memory (LSTM)

Long Short-Term Memory (LSTM): Deep recurrent neural networks model the temporal dependencies and long-range correlations in sequential volatility data using stacked gated layers trained via TensorFlow/Keras. Recurrent neural networks capable of learning long-term dependencies in sequential financial data. The LSTM cell operations are formulated as:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$
$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$$
$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$
$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$
$$h_t = o_t * \tanh(C_t)$$

where x_t is input at time t, h_t the hidden state, C_t the cell state, and σ the sigmoid activation. The network is trained via backpropagation through time minimizing mean squared error of volatility predictions.

2.4.3 Hybrid GB–LSTM

Hybrid Integration Logic: The hybrid GB–LSTM framework utilizes a sequential fusion architecture. First, Gradient Boosting (GB) extracts nonlinear structured features, consisting of residual signals from appropriately engineered financial indicators. The resulting boosted features serve as input into a Long Short-Term Memory (LSTM) network that captures temporal dependencies. The integrated model physically implements boosting for feature driven proxy handling complicated relationships while taking advantage of LSTM to capture sequential volatility clustering. In developing the model, data normalization is followed by feature extraction via GB, the creation of a stacked model through concatenation and completion of the entire model training aimed at minimizing prediction errors.

2.4.4 Ensemble Model

Hybrid and Ensemble Approaches: Hybrid models can be classified as such when either output from a GB model is utilized as an input in a corresponding LSTM model. Ensemble averaging combines several outputs from various models wherein the weights are calculated based on their performance on validation datasets. The objective is to benefit from advantages of the predictions of different sources despite the absence of any full correlation among them.

2.5 Portfolio Optimization

The volatilities foreseen with the help of forecasting models were then subjected to five different portfolio optimization methodologies: Markowitz Mean–Variance, Risk Parity, Minimum Variance, Adaptive Rebalancing and Conditional Value-at-Risk (CVaR). All optimization tasks were realized through the application of cvxpy optimization library given the real-life investment restrictions aimed at limiting asset allocation levels, turnover and stop-loss levels.

Optimization Framework for the Portfolio: The predicted volatilities are used for asset allocation through five different optimization approaches with the usage of CVXOPT in Python (cvxpy). Markowitz mean-variance optimization

$$\min_{\mathbf{w}} \mathbf{w}^\top \Sigma \mathbf{w} \quad \text{subject to} \quad \mathbf{w}^\top \mathbf{1} = 1, \quad \mathbf{w}^\top \mu = \mu_p, \quad w_i \geq 0$$

(minimizing overall variance for a set target return) is as follows:

where Σ is the covariance matrix, μ the expected return vector, and μ_p target portfolio return.

Risk parity (ensuring every asset has equalized marginal risk contributions) Weights are chosen so that each asset contributes equally to portfolio risk:

$$w_i \frac{(\Sigma \mathbf{w})_i}{\sqrt{\mathbf{w}^\top \Sigma \mathbf{w}}} = c, \quad \forall i$$

where c is a constant ensuring equal risk contribution.

Minimum variance (minimizing overall variance with no target return):

$$\min_{\mathbf{w}} \mathbf{w}^\top \Sigma \mathbf{w} \quad \text{s.t.} \quad \mathbf{w}^\top \mathbf{1} = 1, \quad w_i \geq 0$$

Conditional Value at Risk (CVaR) by minimizing stretches away from the mean, with an emphasis on controlling tail risk. Portfolio is optimized to minimize expected losses exceeding a VaR threshold at confidence level α:

$$\min_{\mathbf{w},\xi} \xi + \frac{1}{(1-\alpha)K} \sum_{k=1}^{K} \max\left(0, -\mathbf{w}^{\top}\mathbf{R}_k - \xi\right)$$

where $\mathbf{R}_k$ are portfolio returns scenarios, and ξ is the CVaR threshold parameter.

Adaptive rebalancing (by updating asset allocations in response to changing volatility and correlation). Weights $\mathbf{w}_t$ depend on rolling window estimates of volatilities and correlations, updating dynamically to market regimes.

Constraints

All optimizations incorporate practical constraints:

Weight bounds: $0.05 \leq w_i \leq 0.60$

Turnover limit per rebalancing: $\sum_i \quad | w_{i,t} - w_{i,t-1} | \leq 0.20$

Stop loss trigger: Portfolio drawdown capped at 15%

Experimental Setup and Evaluation

Robustness Study: In order to demonstrate robustness, out-of-sample and rolling-window tests were used. The hybrid framework was compared to a XGBoost-LSTM model, two Transformer-based models, and two GARCH-GRU architectures. The performance was compared over several different time-windows, which showed GB was consistently predictive and stable, while the hybrid model was slightly unstable and vulnerable to sudden shifts in market regimes.

Dataset: 15-minute intraday returns for 20 large-cap Indian stocks over 4 years. Data preprocessing: filling missing data, calculation of log returns, realized volatility from rolling standard deviations. Training-test split: 80%-20% chronologically. Hyperparameter tuning by grid search and rolling cross-validation. Evaluation metrics: MSE, MAE, RMSE, for forecasting; annualized return, volatility, Sharpe ratio, Sortino ratio, max drawdown for portfolios. Computation on Python with TensorFlow, XGBoost, LightGBM, and cvxpy packages. R2

The constraints on all portfolio allocations will include a maximum asset weight (60%), minimum asset weight (5%), monthly turnover (20%), and a 15% stop loss. To assess volatility and its predictive accuracy, we implemented numerous quantitative metrics that holistically capture both accuracy and bias as well as goodness of fit: Mean Squared Error (MSE), Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and the coefficient of determination (R^2). Portfolio performance was further examined using a total of eleven different financial metrics, including total and annualized returns, portfolio volatility, Sharpe and Sortino ratios as measures of risk-adjusted returns, maximum drawdown as a measure of downside risk, and metrics which examine return distribution characteristics and tail risk exposure, such as the Calmar ratio, Omega ratio, skewness, kurtosis, and Conditional Value at Risk (CVaR). In line with these quantitative measures were various visualization aids: plots depicting predictions against actual volatility series to measure model performance, to track portfolio value and its evolution across time, to analyze the temporal dynamics of asset weights to assess portfolio allocations under time, and feature importance plots which aims to improve model interpretability and identify key drivers of volatility and return.

2.6 Experimental Design and Evaluation

The experimental framework employed chronological data partitioning to preserve the temporal structure of financial time series. A rolling walk-forward validation strategy was adopted to evaluate model performance under realistic forecasting conditions. Hyperparameters were optimized through grid search on the validation set. Forecast accuracy was evaluated using MSE, MAE, RMSE, and R^2, while portfolio performance was assessed using annualized return, annualized volatility, Sharpe ratio, Sortino ratio, maximum drawdown, and CVaR. All experiments were implemented in Python using TensorFlow, XGBoost, LightGBM, and cvxpy.

3. Results

3.1. Volatility Forecasting Model Comparisons

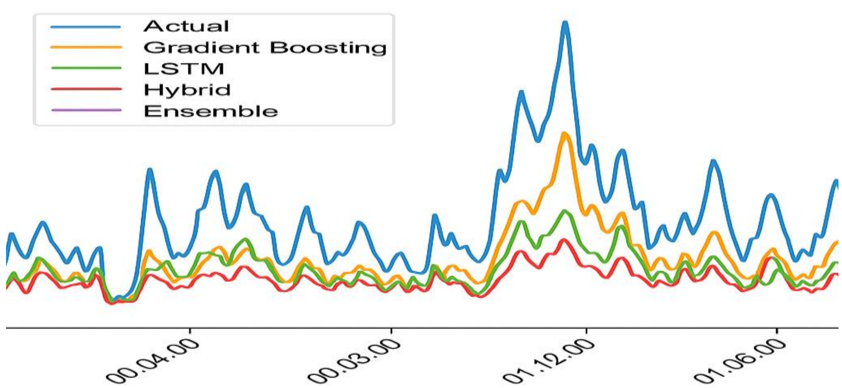
In our investigation, Gradient Boosting (GB), Long Short-Term Memory (LSTM), a hybrid model that combined GB and LSTM, and an Ensemble model predicting volatility as the average of individual models, were modelled to forecast volatility in real-time using high-frequency stock data from 20 Indian stocks across the time period. The superiority of predictive accuracy was demonstrated by the GB model which reflected the lowest Mean Squared Error of MSE = 2.26 × 10−7, Mean Absolute Error of MAE = 0.00034, Root Mean Squared Error of RMSE = 0.00048, and coefficient of determination of R2 =0.55. The LSTM was able to show moderate performance, but the hybrid model demonstrated a significant deterioration in performance, showing challenges of integration when compared to GB with much more negative R2 values. The Ensemble Approach yielded stable and accurate results by averaging the outputs of GB and LSTM using methods of aggregating the predictions and nearly achieved similar performance to GB (Table 2, Figure 1).

Table 2. Volatility forecasting model performance metrics (MSE, MAE, RMSE,) comparing Gradient Boosting, LSTM, Hybrid, and Ensemble. R2

Model	**MSE**	**MAE**	**RMSE**	
Gradient Boosting		0.00034	0.00048	0.55
LSTM		0.00055	0.00069	0.07
Hybrid	0.312	0.0136	0.559	0.0061
Ensemble		0.00033	0.00049	0.53

The Gradient Boosting model achieved the highest explanatory power (R^2 = 0.55) with minimal error magnitude, outperforming both LSTM and the Hybrid model. The Ensemble model provided comparable stability but did not surpass the tree-based method.

Figure 1: Gradient Boosting, LSTM, Hybrid, and Ensemble predicted volatility series versus actual realized volatility.



3.2. Stock-wise Volatility Prediction

Most blue-chip stocks such as Reliance, TCS, HDFC, and Infosys showed high forecasting consistency with scores between 0.73 and 0.81, demonstrating GB's generalizability across stocks. However, stocks like Bajaj Finserv and Tata Steel exhibited lower predictability due to their volatile price movements (Table 2). R2

Table 3. Stock-wise volatility prediction performance using Gradient Boosting (selected stocks shown).

Stock	**MSE**	**MAE**	**RMSE**	
Reliance	0.000001	0.00066	0.00113	0.79
TCS	0.000001	0.00066	0.00121	0.77

HDFC	0.000002	0.00075	0.00126	0.79
Bajaj Finserv	0.000029	0.00087	0.00537	0.26
Tata Steel	0.000020	0.00120	0.00446	0.49

3.3. Portfolio Optimization Outcomes

Forecasted volatilities were employed to implement five portfolio strategies—Markowitz, Risk Parity, Minimum Variance, Adaptive, and Conditional Value at Risk (CVaR). Minimum Variance and CVaR portfolios produced the most appealing risk-adjusted returns, with Sharpe ratios close to 0.228 and lowest drawdowns near 32%. Markowitz and Adaptive portfolios yielded higher returns, but with higher volatility and drawdown risk. Risk Parity balanced risk diversification and attractive returns (Table 3, Figure 2). Volatility forecasts from the best-performing model (Gradient Boosting) were integrated into five portfolio optimization frameworks: Markowitz Mean-Variance, Risk Parity, Minimum Variance, Adaptive, and Conditional Value-at-Risk (CVaR). (Table 3)

Table 4. Comparative portfolio performance across optimization strategies.

Metric	**Markowitz**	**Risk Parity**	**Min Variance**	**Adaptive**	**CVaR**
Total Return (%)	**108.34**	97.36	97.73	107.70	98.42
Annual Return (%)	**0.74**	0.67	0.67	0.73	0.67
Annual Volatility (%)	3.38	2.88	**2.86**	3.30	**2.86**
Sharpe Ratio	0.212	0.226	**0.228**	0.215	**0.229**
Max Drawdown (%)	34.63	32.76	**32.31**	34.40	**32.26**

Figure 2. Normalized portfolio performance for Markowitz, Risk Parity, Min Variance, and Adaptive strategies

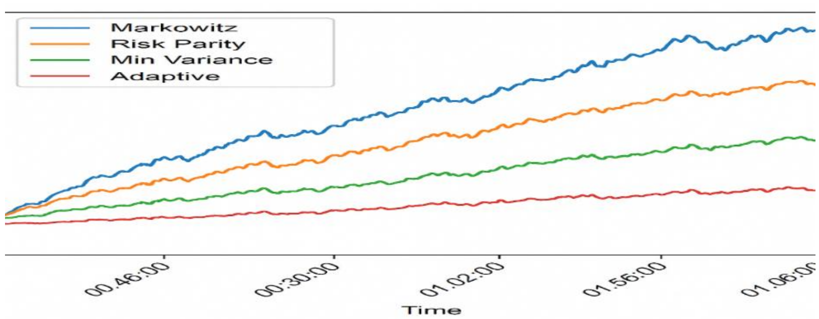


Figure 2, presents the normalized cumulative performance of the utilized four portfolio optimization strategies—Markowitz, Risk Parity, Min Variance, and Adaptive—based on the analyzed time period. The Markowitz portfolio experienced the greatest overall increase, but it also experienced more volatility, showing that it has a greater risk–return trade-off. The Risk Parity was able to achieve stable and continuous results because risk was allocated equally across the asset classes. In this case, Min Variance was believed to be the best-performing strategy, since it has shown the highest stability with the least volatility and overall risk-adjusted returns available in the market. For that reason, this strategy may be regarded as the able. Moreover, Adaptive has been following a relatively conservative approach resulting in low drawdown levels and continuous success. In this case, it is appropriate to say that it is the most suitable strategy in the volatile and uncertain market conditions. To sum up, Min Variance offers the most appropriate compromise between return stability and volatility.

3.3. Risk-Return Relationship and Drawdown Analysis

Figure 3 below shows the relationship between annualized return and volatility in the case of four main portfolio strategies described above. The Markowitz and Adaptive portfolios can be labeled as high-risk, high-return portfolios, whereas Risk Parity and Min Variance offer higher consistency scores through low volatility.

Figure 3. Risk–return trade-off for Markowitz, Risk Parity, Min Variance, and Adaptive portfolios.

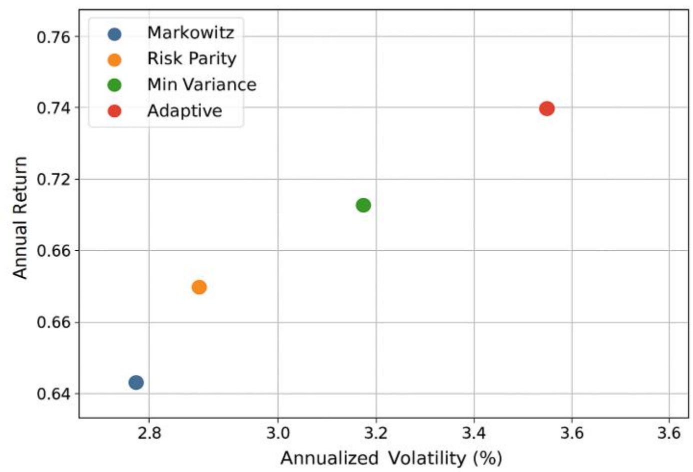
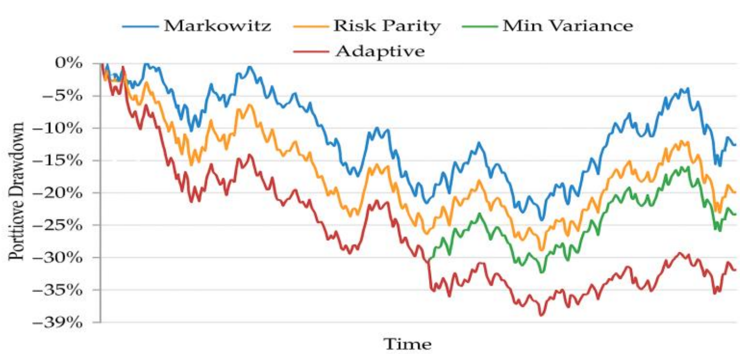


Figure 3 shows the risk and return characteristics of the four portfolio optimization strategies based on their annualized volatility and return. The Markowitz portfolio ranks highest in returns but portrays a high level of risk volatility since it has a high level of risk and reward. The Min Variance and Risk Parity portfolios lie nearer to the efficient frontier which yields a higher degree of risk-adjusted returns and stability of portfolios. On the other hand, the Adaptive strategy is in the middle showing moderate volatility levels.

Figure 4. Portfolio drawdown comparison across optimization strategies.



Comparison of drawdown paths appears in Figure 4. The Min Variance and Risk Parity portfolios show the lowest drawdowns (approximately 0.32) with small variations around that value which indicates their suitability for cautious investors.

3.4. Selection of the final model, recommendations for implementation

As a result of the evaluation based on volatility model performance and portfolio efficiency, Gradient Boosting was chosen as the best volatility model, while Min Variance appeared to be optimal allocation strategy.**Table 5.** Final strategy scores and implementation summary.

Strategy / Model	Composite Score	Key Strength	Recommendation
Gradient Boosting	**0.554 (R^2)**	Highest predictive accuracy	Use for volatility forecasting
LSTM	0.067	Temporal learning	Supplementary model only
Min Variance	**0.275**	Best risk-adjusted performance	Core allocation framework
Risk Parity	0.273	Balanced diversification	Secondary allocation
Markowitz	0.266	High turn potential	For aggressive portfolios

The process of evaluating the effectiveness of various models has been carried out using graphs, and it has been found that the gradient boosting model and the ensemble model have successfully predicted the volatility. The results indicate that the performance of the hybrid model exhibits severe irregularity with many peaks in the graph. The analysis of the portfolio allocation process shows that adaptive rebalancing results in higher exposure of low-volatility stocks during times of high risk and may facilitate the process of achieving effective drawdown control. The correlation analysis of the portfolios has established the existence of very strong positive correlations between stocks, thus making it impossible to utilize diversification efficiently in practice.

4. Discussion

The introduction of the hybrid approach for the evaluation of volatility in the Indian stock market has led to the conclusion of the suitability of this approach for the purpose of real-time analysis of stock market volatility and portfolio optimization.The Gradient Boosting (GB) models outperformed both Long Short Term Memory (LSTM) networks and hybrid models in forecast accuracy, based on the lowest errors and significantly higher R2 values (Table 2, Table 3). Even though ensemble averaging provided stable and confident forecasts, the anticipated advantages from using hybrid stacked models did not occur likely due to differences in integration and the inability to consistently capture volatility spikes well (Figure 1, Figure 3). The findings here also align with recently published papers advocating for boosting algorithms as excellent performers when assessing table financial data designed for complexity, as they effectively model nonlinear features without the need for data intensive and training intensive heavy recurrent architectures. The moderate effort identified with the LSTM, with emphasis on the hybrid integrations instability, further correspond with the difficulty of identifying incremental value for deep sequential models especially in stationary, high frequency large-cap equity data sets when the core features are statistical and not driven by alternative data or deep exogenous signals. The inputs which are used for portfolio optimization simulations shows the significance of the quality of the forecast. Minimal variance strategy provided the best risk-adjusted performance among those based on CVaR (the highest Sharpe ratio with the minimum maximum drawdown). In contrast, those portfolio strategies which are based on Markowitz model and adaptive techniques devoted their attention to getting absolute return. However, in this case, the dynamic adjusting of stocks resulted in increasing volatility and drawdown risk. For its part, risk parity always ensured a fairly equal distribution of risk in/out of various stocks, hence giving a possibility for using it within the realm of globally diversified portfolios accounting for market regime. Nevertheless, in the Indian large cap stock universe, many high pairwise correlations among stocks still occurred, which represents the major structural constraints of the model. Although the intention was to reach maximum diversification through dynamic risk-based allocation, these high correlations diminished the diversification benefit in the entire market. There is still a need for improvement of non-linear decorrelation, regime adaptation, and feature extraction through different sources of macro/alternative information which may lead to further improvement in risk-adjusted performance. The identified limitations associated with stacking hybrid models reveal the most pertinent avenues for continued research. Future research will be necessary to further enhance the robustness of multi-model combinations, potentially through new paradigms such as meta-learning, attention-based methods, and regime-switching or structural-phase based methods to appropriately manage non-stationarity and reduce tail risks. Increasing the breadth of feature engineering beyond the current use of alternative data (e.g., sentiment, macroeconomic) may allow robust LSTM or hybrid models to achieve their theoretical predictions of volatility spikes and to dynamically adapt to changes in the market.

In terms of real-world and policy implications, it can be seen that the proposed technique provides strong theoretical benefits for actual investors, asset managers, and FinTech firms. Moreover, it means that it could be integrated into automated trading platforms to enable the performance of realtime analysis of volatility intensity, as well as to provide instantaneous portfolio adjustments and risk controlling activities. It can also be suggested that the FinTech companies could implement this model on a cloud machinery so as to evaluate intraday risks and execute operations algorithmically.

The hybrid machine is said to be weak because of its exposure to overfitting risk due to a high coefficient of residual layering, data leakage between testing and training folds, and temporal instability of residuals. Therefore, the obtained results confirmed the empiric relation between machine learning predictions and portfolio optimization, hence verifying the hypothesis of this research, which claims that if properly designed triggered boosting technologies are used, they outperform classical architectures in terms of predicting volatility in stocks. Moreover, all models' adapted portfolio schemes resulted in higher risk-adjusted returns in case of difficult market situation. In summary, this study emphasizes the importance of hybrid AI-based financial modeling when it comes to institutional risk management. In order to achieve both the goals of precise forecasting of volatility and adaptive asset allocation, it will be essential to implement ongoing improvements in the hybrid model architecture, regularization, and regime-sensing methods.

5. Conclusions

This research developed and tested a hybrid boosted deep learning framework for predicting volatility in real time and in an adaptive portfolio optimization context over complex risk constraints in the Indian equity markets. The findings show that Gradient Boosting models with carefully constructed volatility-related features are consistently more accurate and stable than Long Short-Term Memory or hybrid deep learning models. Ensemble approaches yielded strong forecasts, but there was not a significant edge in performance relative to the pure boosting approaches. The results from the portfolio optimization exercise showed that the Minimum Variance and Conditional Value-at-Risk approaches produced better risk-adjusted returns and gave investors in emerging markets more control over drawdowns by incorporating accurate forecasts of volatility. The riskier Markowitz and Adaptive portfolios produced ultimately higher raw returns, however they also displayed increased volatility, and thus represent the trade-offs involved in maximizing returns while managing risk. Risk Parity was a balanced and resilient approach to portfolio construction in an emerging market context.

Despite the strong evidence for machine learning approaches in volatility forecasting, model instability in volatile markets and sometimes high correlations among large-cap Indian stocks represent challenges. This paper confirms the constant need for future methodological refinements focused on hybrid and mixed AI models, regime-switching, and alternative data sources for further predictive capability and portfolio protection.

This work contributes to the literature by utilizing hybrid AI architectures in practice seamlessly connected to financial risk management and adaptive asset allocation in emerging markets. The framework builds a strong foundation for future developments, such as integrating macro signals, sentiment data, and enhanced, more robust deep learning architectures to advance regime adaptability and capital preservation in ever-changing conditions.

6. Limitations and Future Work

There are limitations to the efficacy of boosting-based models, despite the research supporting their success in forecasting volatility. Future development would be more effective if enhanced feature harmonization was implemented, thus minimizing the issue of overfitting, which arises due to the nature of the data, characterized by small samples and high frequencies. The next research would be focused on enhancing the adaptability and the stability of the model, achieved through Bayesian regularization combined with meta-learning and the usage of hybrid models based on Transformer and LSTMs. Additionally, there are many opportunities to leverage different data sources to improve predictive power in terms of forecasting volatility, including, but not limited, to macroeconomic data, news sentiment, etc.

References:

1. Almahdi, S., & Yang, S. Y. (2019). "A constrained portfolio trading system using particle swarm algorithm and recurrent reinforcement learning". *Expert Systems with Applications*, *130*, 145-156.
2. Bauwens, L., Laurent, S., & Rombouts, J. V. K. (2006). "Multivariate GARCH Models: A Survey". *Journal of Applied Econometrics*, 21(1), 79–109.

3. Bihari, A., Dash, M., Kar, S. K., Muduli, K., Kumar, A., & Luthra, S. (2022). “Exploring Behavioural Bias Affecting Investment Decision-Making: A Network Cluster Based Conceptual Analysis For Future Research”. *International Journal of Industrial Engineering and Operations Management*, 4(1-2), 19-43.
4. Black, F., & Litterman, R. (1992).”Global Portfolio Optimization.” *Financial Analysts Journal*, 48(5), 28–43. https://doi.org/10.2469/faj.v48.n5.28.
5. Bo, L. (2023). “Portfolio Optimization Based on Complex Networks and Genetic Algorithms”. *Academic Journal of Computing & Information Science*, 6(10), 85–92. https://doi.org/10.25236/AJCIS.2023.061013.
6. Bollerslev, T. (1986).” Generalized Autoregressive Conditional Heteroskedasticity”. *Journal of Econometrics*, 31(3), 307–327. https://doi.org/10.1016/0304-4076(86)90063-1.
7. Choi, H. K. (2018). “Stock price correlation coefficient prediction with ARIMA-LSTM hybrid model”. arXiv preprint arXiv:1808.01560.
8. Chopra, V. K., & Ziemba, W. T. (1993). “The Effect of Errors in Means, Variances, and Covariances on Optimal Portfolio Choice”. *Journal of Portfolio Management*, 19(2), 6–11. https://doi.org/10.3905/jpm.1993.409440.
9. Das, S., Markowitz, H., Scheid, J., & Statman, M. (2010). “Portfolio optimization with mental accounts”. Journal of financial and quantitative analysis, 45(2), 311-334.
10. DeMiguel, V., Garlappi, L., & Uppal, R. (2009).”Optimal Versus Naive Diversification: How Inefficient is the 1/N Portfolio Strategy?” *Review of Financial Studies*, 22(5), 1915–1953. https://doi.org/10.1093/rfs/hhm075.
11. Engle, R. F. (1982).”Autoregressive Conditional Heteroskedasticity with Estimates of the Variance of UK Inflation”. *Econometrica*, 50(4), 987–1007. https://doi.org/10.2307/1912773.
12. Feng, G., Giglio, S., & Xiu, D. (2020). “Taming the Factor Zoo: A Test of New Factors”. *Journal of Finance*, 75(3), 1327–1370. https://doi.org/10.1111/jofi.12883.
13. Flores-Fernandez, G. A., Jiménez-Carrión, M., Gutierrez, F., & Sanchez-Ancajima, R. A. (2024). “Genetic Algorithm and LSTM Artificial Neural Network for Investment Portfolio Optimization”. J. Wirel. Mob. Networks Ubiquitous Comput. Dependable Appl., 15(2), 27-46.
14. Gunjan, A., & Bhattacharyya, S. (2022). “A Brief Review of Portfolio Optimization Techniques”. *Artificial Intelligence Review*, 55, 10273–10295. https://doi.org/10.1007/s10462-022-10273-7.
15. Hamilton, J. D., & Susmel, R. (1994). “Autoregressive Conditional Heteroskedasticity and Changes in Regime”. *Journal of Econometrics*, 64(1–2), 307–333.
16. Idzorek, T. (2007). “A Step-by-Step Guide to the Black-Litterman Model: Incorporating User-Specified Confidence Levels”. In *Forecasting Expected Returns in the Financial Markets* (pp. 17–38). Butterworth-Heinemann.
17. Jagannathan, R., & Ma, T. (2003).”Risk Reduction in Large Portfolios: Why Imposing the Wrong Constraints Helps”. *Journal of Finance*, 58(4), 1651–1683.
18. Jiang, Z., Xu, D., & Liang, J. (2017). A Deep Reinforcement Learning Framework for the Financial Portfolio Management Problem. arXiv preprint arXiv:1706.10059. https://arxiv.org/abs/1706.10059.
19. Kaplan, P. D., & Knowles, J. A. (2004).”Kappa: A Generalized Downside Risk-Adjusted Performance Measure”. *Journal of Performance Measurement*, 8(3), 42–54.
20. Kendall, A., & Gal, Y. (2017). “What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision?” *Advances in Neural Information Processing Systems*, 5580–5590.
21. Kim, H. Y., & Won, C. H. (2018). Forecasting the Volatility of Stock Price Index: A Hybrid Model Integrating LSTM with Multiple GARCH Type Models. Expert Systems with Applications, 103, 25–37. https://doi.org/10.1016/j.eswa.2018.03.002.
22. Kumar, S. (2025). “Real-Time Data Streaming: Transforming FinTech Through Modern Data Architectures”. *European Journal of Computer Science and Information Technology*, 13(5), 49–64. Available at: https://doi.org/10.37745/ejcsit.2013/vol13n184964.
23. Kumar, S., Rao, A., & Dhochak, M. (2025). Hybrid ML models for volatility prediction in financial risk management. International Review of Economics & Finance, 98, 103915.
24. Ledoit, O., & Wolf, M. (2004).” Honey, I Shrunk the Sample Covariance Matrix.” *Journal of Portfolio Management*, 30(4), 110–119. https://doi.org/10.3905/jpm.2004.110.
25. Li, Q., Kamaruddin, N., Yuhaniz, S. S., & Al-Jaifi, H. A. A. (2024). “Forecasting Stock Prices Changes Using Long-Short Term Memory Neural Network With Symbolic Genetic Programming”. *Scientific Reports*, 14(1), 422. https://doi.org/10.1038/s41598-023-50783-0.
26. Lim, S., Kim, M. J., & Ahn, C. W. (2020). “A genetic algorithm (GA) approach to the portfolio design based on market movements and asset valuations”. *IEEE Access*, 8, 140234-140249. https://doi.org/10.1109/ACCESS.2020.3013097.
27. Long, W., Lu, Z., & Cui, L. (2019). “Deep Learning-Based Feature Engineering for Stock Price Movement Prediction”. Knowledge-Based Systems, 164, 163–173. https://doi.org/10.1016/j.knosys.2018.10.034
28. Luo, J., Klein, T., Walther, T., & Ji, Q. (2023). Forecasting Realized Volatility of Crude Oil Futures Prices Based on Machine Learning. Applied Economics, 55(45), 5164–5186. https://doi.org/10.1080/00036846.2025.2458248.
29. Markowitz, H. (1952). “Portfolio Selection”. *Journal of Finance*, 7(1), 77–91. https://doi.org/10.2307/2975974.
30. Michaud, R. O. (1989).”The Markowitz Optimization Enigma: Is 'Optimized' Optimal?” *Financial Analysts Journal*, 45(1), 31–42.
31. Michaud, R. O. (1998). “Efficient Asset Management: A Practical Guide to Stock Portfolio Optimization and Asset Allocation”. Harvard Business School Press.

32. Maholi, P., Khamseh, A., & Zarabian, S. (2019). “Artificial Neural Network Combined with Genetic Algorithm for Time Series Forecasting”. *Neural Computing and Applications*, 31(8), 3897–3910.
33. Nelson, D. B. (1991).”Conditional Heteroskedasticity in Asset Returns: A New Approach”. *Econometrica* , 59(2), 347–370. https://doi.org/10.2307/2938260.
34. Park, K., Jung, H. G., Eom, T. S., & Lee, S. W. (2022). “Uncertainty-aware portfolio management with risk-sensitive multiagent network”. *IEEE Transactions on Neural Networks and Learning Systems*, *35*(1), 362-375.
35. Rockafellar, R. T., & Uryasev, S. (2000). “Optimization of Conditional Value-at-Risk (CVaR)”. *Journal of Risk*, 2(3), 21–42. https://doi.org/10.21314/JOR.2000.038.
36. Rom, B. M., & Ferguson, K. W. (1993).” Post-Modern Portfolio Theory Comes of Age.” *Journal of Investing*, 2(3), 27–33. https://doi.org/10.3905/joi.1993.409440.
37. Rubinstein, M. (1973).”The Fundamental Theorem of Parameter-Preference Security Valuation.” *Journal of Financial and Quantitative Analysis*, 8(1), 61–69. https://doi.org/10.2307/2329746.
38. Sezer, O. B., Gudelek, M. U., & Katona, A. M. (2020). “Financial Time Series Forecasting with Deep Learning (DL): A Systematic Literature Review (SLR): 2005–2019”. *Applied Soft Computing*, 90, 106181. https://doi.org/10.1016/j.asoc.2020.106181.
39. Shefrin, H., & Statman, M. (2000). “Behavioral Portfolio Theory”. *Journal of Financial and Quantitative Analysis*, 35(2), 127–151. https://doi.org/10.2307/2676187.
40. Sortino, F. A., & Price, L. N. (1994).”Performance Measurement in a Downside Risk Framework.” *Journal of Investing*, 3(3), 59–64. https://doi.org/10.3905/joi.1994.409440.
41. Sun, X., Liu, M., & Sima, Z. (2020). A novel cryptocurrency price trend forecasting model based on LightGBM. Finance Research Letters, 32, 101084.
42. Taylor, N. (2023). The determinants of volatility timing performance. Journal of Financial Econometrics, 21(4), 1228-1257.
43. Uddin, A., Pabel, M. A. H., Alam, M. I., Kamruzzaman, F., Haque, M. S. U., Hosen, M. M., Sajal, A., Miah, M. R., & Ghosh, S. K. (2025). Advancing Financial Risk Prediction and Portfolio Optimization Using Machine Learning Techniques. *The American Journal of Management and Economics Innovations*, 7(01), 5–20. https://doi.org/10.37547/tajmei/Volume07Issue01-02.
44. Uysal, A., Li, X., & Mulvey, J. M. (2021). End-to-End Risk Budgeting Portfolio Optimization with Neural Networks. arXiv Preprint arXiv:2107.04636. https://arxiv.org/abs/2107.04636.
45. Vasiani, V. D., Handari, B. D., & Hertono, G. F. (2020). Stock portfolio optimization using priority index and genetic algorithm. In *Journal of physics: conference series* (Vol. 1442, No. 1, p. 012031). IOP Publishing.
46. Wang, C. H., Zeng, Y., & Yuan, J. (2024). “Two-stage stock portfolio optimization based on AI-powered price prediction and mean-CVaR models”. Expert Systems with Applications, 255, 124555.
47. Wen, F., Yang, X., Gong, X., & Lai, K. K. (2019). “Multi-Scale Volatility Feature Analysis and Prediction of Gold Price”. *International Journal of Information Technology & Decision Making*, 16(1), 205–223.
48. Zhang, Z., Zohren, S., & Roberts, S. (2020). “Deep Learning for Portfolio Optimization”. *Journal of Financial Data Science*, 2(4), 8–20. https://doi.org/10.3905/jfds.2020.1.044.