Networks and Heterogeneous Media

https://www.aimspress.com/journal/nhm

NHM, 21(3): 848–864.
DOI: 10.3934/nhm.2026035
Received: 12 February 2026
Revised: 29 March 2026
Accepted: 03 April 2026
Published: 06 May 2026

Research article

Machine learning models and ensemble methods for stock index return forecasting

Aivaras Bielskis* **and Igoris Belovas**

Institute of Data Science and Digital Technologies, Vilnius University, Akademijos str. 4, Vilnius, LT-08412, Lithuania

* **Correspondence:** Email: aivaras.bielskis@mif.stud.vu.lt.

Abstract: Reliable prediction of financial market movements remains a challenging task due to high volatility, complex interdependencies, and sensitivity to external shocks. This study assessed the performance of advanced machine learning models, including long short-term memory (LSTM), gated recurrent unit (GRU), transformer networks, extreme gradient boosting (XGBoost), and deep multi-layer perceptron (DMLP), as well as proposes their ensemble combinations, in forecasting daily closing prices of five major stock indices (S&P 500, NASDAQ-100, Dow Jones Industrial Average, FTSE 100, and DAX). Results indicate that although all models achieved high predictive accuracy, profitability outcomes varied substantially across models and markets. Among single-model approaches, LSTM generally exhibited more stable positive returns in several indices, while other models showed pronounced variability depending on market conditions. Meanwhile ensemble strategies frequently ranked among the top-performing configurations, often matching or exceeding the performance of adaptive weighting schemes. Performance was strongly index-dependent, with S&P 500 and NASDAQ-100 exhibiting comparatively stronger profitability, whereas FTSE and Dow Jones showed weaker and less differentiated results. These findings emphasize that statistical accuracy (e.g., RMSE, R^2 metrics) alone is insufficient for profitable trading, underscoring the importance of financial performance metrics, such as total return, drawdown, and risk-adjusted measures, when evaluating predictive models.

Keywords: returns forecasting; machine learning; deep learning; ensemble methods; stock indices; trading

1. Introduction

The prediction of stock prices is a particularly challenging task, as prices are driven by numerous interrelated factors. The efficient market hypothesis (EMH), which posits that asset prices fully incorporate all available information, represents a fundamental obstacle to forecasting (cf. [1]).

Nonetheless, empirical evidence occasionally contradicts the EMH [2], suggesting that market anomalies can lead to deviations from theoretically expected price behavior. Moreover, recent research (see detailed survey in Section 2) indicate that promising forecasting performance is indeed achievable.

This study employs five models (long short term memory [LSTM], graded recurrent unit [GRU], extreme gradient boosting [XGBoost], deep multi-layer perceptron [DMLP], and transformer-based architectures) selected for their capacity to capture long-term dependencies, nonlinear dynamics, and high-dimensional interactions in time series data. The dataset comprises five major stock indices (S&P 500, NASDAQ-100, Dow Jones Industrial Average, FTSE 100, and DAX) spanning 2000–2025. To ensure unbiased performance evaluation, the data are partitioned into four segments: training, validation, extended retraining, and a final holdout test.

We introduce an ensemble approach that integrates model outputs using simple averaging and adaptive weighting schemes learned via linear regression (LR), ridge regression with polynomial features, and elastic net (EN), improving robustness under market fluctuations. Model evaluation incorporates both statistical metrics (MSE, MAE, R^2) and financial performance measures (maximum drawdown (MDD), total return (TR), annualized return (AR) and Calmar ratio (CR)), ensuring that results are assessed in terms of both predictive accuracy and risk-adjusted profitability. Finally, predictions are implemented in a long–short trading framework, where buy and sell signals are derived from predicted price movements relative to prior closing values. Trades are executed with risk controls in place, all trades incorporate a fixed transaction cost of 0.1% per buy and per sell; no additional slippage or bid-ask spread effects are explicitly considered, and performance is monitored through capital evolution and profit-and-loss (PnL) analysis.

The paper is organized as follows. The first part is the introduction. Section 2 provides an overview of recent studies. Section 3 details the methodological framework, including data preprocessing, model architectures, and evaluation metrics. Section 4 presents the experimental analysis and results, examining model performance and discussing the implications of the findings, with attention to strengths, limitations, and potential improvements. Finally, Section 5 concludes the study by summarizing key insights and outlining directions for future research.

2. Literature survey

While traditional econometric approaches (e.g., ARMA, ARIMA, and SARIMA) often struggle to capture the nonlinear dependencies and complex temporal dynamics of financial data, recent advances in machine learning address these limitations by offering models capable of handling long-term dependencies, high-dimensional feature interactions, and structural regime shifts. Within this context, ensemble learning has gained prominence as a strategy to improve predictive robustness by integrating diverse models. Instead of relying on a single architecture, ensembles combine complementary strengths, reduce variance, and mitigate biases, particularly in volatile markets. Moreover, hybrid methods that integrate signal decomposition, feature engineering, and econometric modeling with deep learning have demonstrated potential in enhancing both accuracy and profitability. The following section reviews recent contributions to financial time series forecasting, highlighting methodological innovations and empirical findings across diverse datasets and modeling strategies.

In a recent study, Olorunnimbe and Viktor [3] proposed an ensemble temporal transformer for financial time series forecasting. Rather than relying on ARIMA or a single deep model, they trained multiple transformers on sliding windows and combined outputs using averaging and a stacking meta-learner with a quantile estimator. Applied to 20 DJIA stocks, their method improved predictive accuracy by 40–60% over baseline transformers, particularly in volatile markets, underscoring the benefits of ensemble diversity and decomposition. In a complementary direction, Chen et al. (2024) [4] introduced a two-stage ensemble that integrates deep learning with portfolio optimization. Predictions from LSTM, GRU, MLP, and RF were combined through a time-varying OLS model and then incorporated into a diversified mean-variance with forecasting (DMVF) framework. Experiments on Shanghai Stock Exchange data demonstrated superior return–risk performance, highlighting the role of ensemble learning in financial decision making. Ma et al. [5] compared machine learning (ML) models (RF and SVR) and deep learning models (LSTM, DMLP and CNN) against ARIMA on China Securities 100 Index data. RF delivered the strongest performance among ML methods, while LSTM proved most effective among DL models in reducing errors. Their findings indicate that advanced models can enhance return prediction, but performance is model- and context-dependent, emphasizing the importance of careful model selection.

Signal decomposition techniques have also been used to improve forecasts. Dezhkam and Manzuri [6] applied the Hilbert–Huang transform (HHT) to extract frequency, amplitude, and phase features from stock time series, which were then fed into XGBoost. On *Standard and Poor's 500* stocks, this hybrid approach achieved higher cumulative returns and Sharpe ratios than benchmarks, illustrating the value of combining decomposition with machine learning. Feature engineering is another critical driver of forecasting gains. Daul et al. [7] investigated stock return prediction using financial, fundamental, and sentiment features with LASSO, LightGBM, and neural networks. Non-linear models, particularly neural networks, outperformed linear approaches by capturing complex feature interactions. The performance attribution method confirmed that interaction effects drive predictive gains, underscoring their importance in improving forecasting accuracy for investment strategies.

Transformer-based models are drawing more attention recently. Gezici and Sefer [8] introduced a framework leveraging vision transformers (ViT), Swin, and DeiT by converting OHLCV data and 65 technical indicators into 2D images. Tested on 20 years of ETF data, ViT outperformed LSTM, CNN-TA++, and buy & hold strategies, achieving superior F1 scores, annualized returns, and Sharpe ratios. Their results highlight ViT's profitability, robustness, and adaptability across market conditions. Hybrid methods that blend econometric and deep learning approaches also show promise. Mutinda and Langat [9] integrated GARCH with LSTM, GRU, and transformer models for Airtel stock forecasting (2019–2024), addressing missing values with GANs. GARCH–DL hybrids consistently outperformed standalone networks, with GARCH-LSTM yielding the best accuracy (RMSE = 0.2002, R^2 =0.9995). The results showed that combining volatility modeling with deep learning enhances predictive performance, especially in volatile emerging markets.

In 2024, to address the challenge of small-sample forecasting in agricultural commodity markets, Yue and Liu [10] used a multi-scale approach combining TimeGAN, XGBoost, and TCN-Attention. TimeGAN expanded sparse datasets, XGBoost handled feature selection, and TCN-Attention refined predictions. The approach reduced RMSE by 1.7% and improved R^2 accuracy by 4.3% compared to SVR and RF, while also outperforming GRU and LSTM with lower computational costs. This study illustrates the effectiveness of data augmentation in resource-constrained forecasting scenarios. Finally, Huang and Wang [11] compared ARIMA, ANN, CNN, transformer, GRU, and LSTM across

53 years of IBM, GE, and ExxonMobil stock price data. LSTM consistently achieved the best accuracy across datasets and horizons, with GRU as a strong alternative. Transformers performed well on larger datasets but struggled with smaller samples, while CNN and ANN risked overfitting. ARIMA remained competitive only in short-term forecasts. Their findings stress the need to align model choice with dataset size and forecasting horizon.

Recent studies also suggest that forecasting performance depends not only on model architecture, but also on how prediction targets and ensemble structures are defined. Yan et al. [12] proposed a flexible target prediction framework for quantitative trading in the American stock market, where multiple targets, such as raw price differences, moving-average differences, and exponential moving-average differences, were predicted using ensemble, fusion, and transfer-learning-based models. Their results showed that smoothed targets were generally easier to predict than raw price changes, implying that target design can materially influence both forecasting accuracy and downstream trading performance. In a related direction, Hayati et al. [13] developed an electricity price forecasting framework combining PCA, heterogeneous machine learning and deep learning predictors, and meta-ensemble learning, complemented by SHAP-based explainability. Their study showed that a LR meta-ensemble outperformed the individual models, while also providing interpretable evidence on feature and model contributions. Although this study was conducted outside direct stock index return forecasting, it is relevant to the present work because it reinforces the importance of ensemble construction and model interpretability in improving predictive systems.

Methodological advances have also emphasized that strong forecasting systems should address not only predictive accuracy but also reliability and evaluation rigor. Brusaferri et al. [14] proposed on-line conformalized neural network ensembles for day-ahead electricity price forecasting, extending ensemble methods toward probabilistic prediction through conformalized quantile regression and online recalibration. Their findings showed improved interval reliability, hourly coverage, and robustness under distribution shift, which is especially important in volatile and non stationary environments. Choi et al. [15], in turn, examined decomposition-based forecasting from a methodological perspective and proposed temporal consistency ensemble empirical mode decomposition (TC-EEMD) for practical metal price forecasting. Their study demonstrated that many decomposition-based methods may overstate performance because of future data leakage and unrealistic decomposition procedures, highlighting the importance of temporally consistent evaluation. Together, these studies broaden the scope of ensemble forecasting beyond predictive performance, drawing attention to uncertainty estimation, robustness, and the credibility of the experimental setup itself.

In summary, although the surveyed models and ensembles improve financial forecasting, the reviewed approaches still exhibit notable limitations. First, they concentrate on statistical accuracy (e.g., RMSE, R^2) rather than financial outcomes, such as returns, drawdowns and risk-adjusted performance. Second, the works rely on single models or limited ensemble setups, leaving gaps in systematic comparison. Building on these insights, we evaluate five selected models (LSTM, GRU, transformer, XGBoost and DMLP), and introduce new ensemble strategies, thus linking predictive accuracy with financial profitability.

3. Data and methods

We focus on five major stock indices (S&P 500, NASDAQ-100, Dow Jones Industrial Average (DJIA), FTSE 100, and DAX) to ensure broad market coverage and five models (LSTM, GRU, transformer, XGBoost, and DMLP), which are selected for their effectiveness in reducing forecasting errors. The models were trained and used to generate predictions on daily closing price returns to ensure stationarity, while evaluation metrics were computed on the reconstructed price series to capture both statistical accuracy and practical financial relevance.

Data and partitioning

The dataset indices are presented in Table 1. For model training, validation, and testing, the dataset is divided into four distinct periods, providing a structured framework for evaluating model performance: primary training window (**PT**) used for model development and parameter optimization; validation window (**VD**) for model selection and ensemble weight calibration; extended training window (**ET**) used to retrain models with optimal configurations for final forecasting; holdout period (**HT**) reserved exclusively for out-of-sample performance evaluation and visualization (see Figure 1).

This partitioning ensures that models are trained on historical data, validated on more recent periods to fine-tune their performance, and tested on unseen future data to evaluate generalization capabilities. The dataset under examination is hosted on GitHub [21].

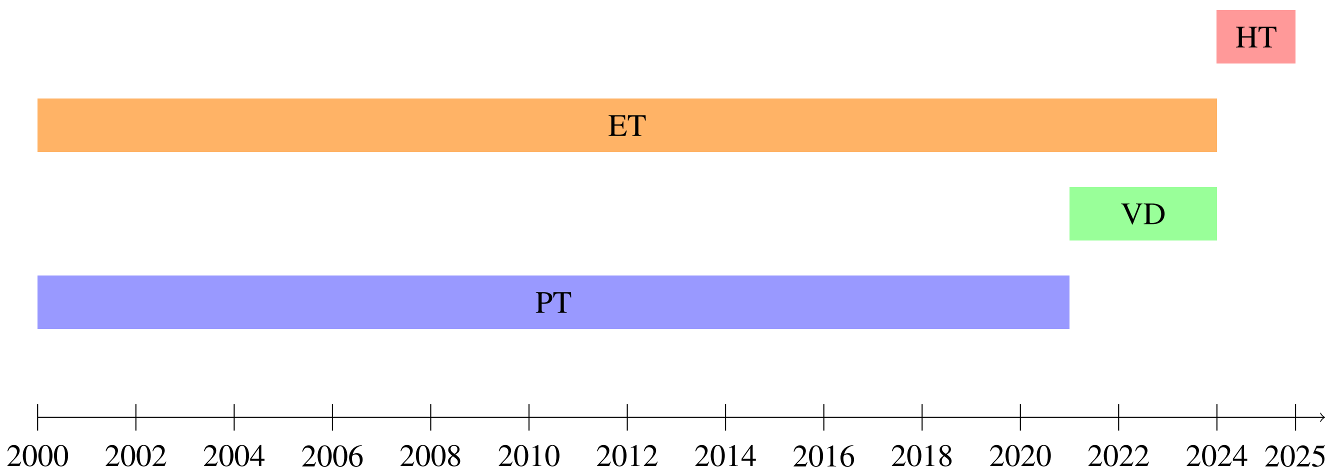

Figure 1. Dataset partitioning: Primary training (PT), validation (VD), extended training (ET), and holdout (HT).

Table 1. Overview of selected stock market indices.

Ticker	Description
GSPC	S&P 500 tracks 500 leading US companies, including Apple, Microsoft, and Amazon [16].
NDX	NASDAQ-100 represents 100 largest nonfinancial firms on NASDAQ, such as Apple, Microsoft, and Alphabet [17].
DJI	Dow Jones Industrial Average covers 30 large U.S. companies from diverse industries, including Goldman Sachs, IBM, and McDonald's [18].
FTSE	FTSE 100 comprises 100 largest firms on the London Stock Exchange, including HSBC, BP, and GlaxoSmithKline [19].
GDAXI	DAX tracks 30 German blue-chip companies listed in Frankfurt, including Volkswagen, Siemens, and Bayer [20]

Deep learning models

Long short-term memory (LSTM). Deep learning models, such as LSTMs [22], learn long-range dependencies in sequences via gated memory states. Heaton et al. [23] show that the LSTM network mitigates vanishing/exploding gradients and captures both short- and long-term dependencies. In stock prediction studies (cf. Lee and Yoo [24] or Bielskis and Belovas [25]), LSTMs were reported to achieve superior performance relative to several baseline models under their respective experimental settings. LSTMs are well-suited to time-series forecasting because they learn features directly from raw sequences and can model long-horizon temporal dependencies without handcrafted feature engineering.

In the presented implementation, we use a stacked LSTM architecture consisting of two LSTM layers, each followed by dropout regularization, and a final dense output layer. The model input is a sequence of length T with one feature per step, and the output is a single-step forecast.

Gated recurrent unit (GRU). The LSTM architecture includes multiple gates, leading to a relatively large number of parameters and higher training cost on large datasets. To address these issues, Cho et al. [26] introduced the GRU, a more parameter-efficient variant. For example, Wu et al. [27] applied GRUs to electricity price prediction and reported better performance than several traditional methods. GRUs simplify the gating structure by merging the forget and input mechanisms into a single update gate and omitting the output gate, thereby reducing complexity while maintaining the ability to model long-term dependencies.

We implement a stacked GRU architecture consisting of two GRU layers (the first returning sequences and the second not), each followed by dropout regularization, and a final dense output layer. The model input is a sequence of length T with one feature per step, and the output is a single-step forecast.

Transformer-based time series model. The proposed model is based on a simplified transformer encoder architecture with multi-head self-attention for time series forecasting. For a given input sequence $X \in \mathbb{R}^{T \times D}$, where T denotes the number of time steps and $D = 1$ corresponds to the scaled closing price, the sequence is first linearly projected into a d_{model}-dimensional representation. To incorporate temporal information, learnable positional embeddings are added to the projected sequence (cf. [3, 4, 28]). We employ a stacked transformer encoder consisting of L layers, learnable positional embeddings, and a final dense output layer. The model input is a univariate sequence of length T, and the output corresponds to a single-step forecast of the target variable.

XGBoost is a popular ensemble model for regression tasks, including price prediction. Chen et al. [29] applied XGBoost in their work on portfolio construction. The algorithm is based on gradient-boosted decision trees and optimizes the following objective function:

$$\mathcal{L}(\Theta) = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k), \tag{3.1}$$

where $l(y_i, \hat{y}_i)$ is the loss function (e.g., squared error loss, as used in our implementation), and $\Omega(f_k)$ is the regularization term that controls the complexity of each tree f_k. Here, K denotes the number of boosting rounds.

Deep multi-layer perceptron (DMLP) (see [30, 31]) is a feed-forward neural network architecture composed of multiple fully connected layers with non-linear activation functions. Formally, the output of layer l can be expressed as:

$$h^{(l)} = \sigma(W^{(l)} h^{(l-1)} + b^{(l)}), \tag{3.2}$$

where $h^{(l)}$ is the output of the l-th layer, $W^{(l)}$ is the weight matrix, $b^{(l)}$ is the bias vector, and σ is a non-linear activation function such as ReLU or sigmoid.

The adopted DMLP architecture first transforms the input sequence of length T into a single flattened vector, followed by two dense layers with ReLU activations and dropout regularization. A final dense output layer is used to produce a single-step forecast of the target variable.

Performance metrics

Model evaluation considers two categories: prediction accuracy (PA) and profitability (PR). Prediction accuracy is measured by standard error metrics: MSE (average squared difference between predicted and actual values), MAE (average absolute difference, directly interpretable in data units), and R^2 (proportion of variance in actual values explained by predictions). The error metrics (MSE and MAE) are computed on the reconstructed price series (after transforming predicted returns back to price levels), rather than directly on returns. Profitability (trading performance) is assessed by: MDD (largest peak-to-trough portfolio loss [32]), TR (percentage portfolio gain/loss over the period), AR (compounded yearly return), and CR (risk-adjusted return, defined as AR/MDD). Financial performance metrics are derived directly from algorithmic trading results, providing a combined evaluation of predictive accuracy and profitability in real market conditions.

Strategy implementation

The study implements a basic LS-strategy based on the predictions generated by ML models. The strategy adopts the following structured framework.

Signal generation. Several signal-generation rules were tested out-of-sample using threshold-based filters on the predicted return ρ. The selected strategy generates trading signals S_t by comparing the predicted price P_t^{pr} with the previous day's close P_{t-1}^{cl}. No trade is executed when the predicted return remains within a tolerance band, while trades are triggered only when the deviation exceeds the threshold. Specifically, a *buy* signal is issued when the predicted price exceeds the previous close and the deviation is larger than the threshold, whereas a *sell* signal is generated when the predicted price falls below the previous close under the same condition. For $\varepsilon = 10^{-4}$ and the predicted return ρ,

$$\rho = \frac{P_t^{pr}}{P_{t-1}^{cl}} - 1,$$

the rule is defined as

$$S_t = \begin{cases} \text{no signal}, & \text{if } |\rho| \leq \varepsilon, \\ \text{buy}, & \text{if } \rho > \varepsilon, \\ \text{sell}, & \text{if } \rho < -\varepsilon. \end{cases} \tag{3.3}$$

Trading execution. The algorithm executes trades based on the generated signals, taking long or short positions accordingly. Risk management is incorporated by closing positions when the price reaches predefined upper or lower thresholds. Capital allocation is based on discrete position sizing, where the number of units N_t is computed as

$$N_t = \left\lfloor \frac{C_t}{P_t^{op}} \right\rfloor, \tag{3.4}$$

ensuring that only whole units are traded by rounding down the fractional allocation. Here C_t denotes the available capital and P_t^{op} is the asset's opening price. A transaction cost of 0.1% is applied to every buy and sell order.

Trade calculation. The algorithm tracks each trade's performance; updates the position; and calculates profit and loss Θ_t, capital fluctuations, and unrealized gains or losses:

$$\Theta_t = \begin{cases} (P_t^{cl} - P_{t-1}^{buy}) \times N_t, & \text{if long position}, \\ (P_{t-1}^{sell} - P_t^{cl}) \times N_t, & \text{if short position}, \end{cases} \tag{3.5}$$

Unrealized capital C_t^{un} (considering ongoing trades) is computed as

$$C_t^{un} = C_t + \Theta_t. \tag{3.6}$$

This strategy ensures disciplined trading execution and capital allocation while maintaining effective risk management through pre-defined exit conditions. The trading performance is evaluated assuming an initial capital of 100,000 monetary units. Returns are computed using a compounding framework, where profits and losses are reinvested over time and capital is updated after each trade.

Model integration

Let $\mathcal{M} = \{\text{LSTM}, \text{GRU}, \text{transformer}, \text{XGBoost}, \text{DMLP}\}$ be the set of base forecasters. For each non-empty subset $S \subseteq \mathcal{M}$, an ensemble prediction is formed as

$$\hat{y}_t^{(m,S)} = \sum_{j \in S} w_j^{(m,S)} f_{j,t} + b^{(m,S)}, \tag{3.7}$$

where $f_{j,t}$ is model j's forecast, with weights w_j and intercept b estimated by method m. We study four weighting schemes: AVG (equal weights, no intercept), LR (least-squares regression on the validation data), EN (EN regression with L_1 and L_2 penalties) and PR (ridge regression applied to second-order polynomial expansions of the model forecasts). All model parameters were trained in-sample, while ensemble weights and trading thresholds were determined on the validation (VD) set. Specifically, the inputs to the weighting models (LR, EN and PR) were the predictions of the individual base models on the VD set, and the targets correspond to the observed values. The learned weights were fixed and applied to the entire holdout test (HT) period without further updating, ensuring a strictly out-of-sample assessment. These choices provide the foundation for the results and discussion presented next. The implementation is hosted on GitHub [21].

Model hyperparameters

The hyperparameters of all models were selected empirically based on validation performance using a combination of random search and grid search over the training and validation sets. The final configurations used in the experiments are presented in Table 2.

Table 2. Hyperparameter settings of the forecasting models.

Model	Hyperparameters
LSTM	2 layers (256, 128 units), dropout = 0.3, learning rate = 0.001, epochs = 500, batch size = 64, time steps = 30
GRU	2 layers (256, 128 units), dropout = 0.3, learning rate = 0.001, epochs = 500, batch size = 64, time steps = 30
DMLP	2 dense layers (256, 128 units), dropout = 0.3, learning rate = 0.001, epochs = 500, batch size = 64, time steps = 30
Transformer	d_model = 64, heads = 4, FF dim = 256, layers = 3, dropout = 0.3, learning rate = 0.0001, epochs = 500 (train) / 300 (test), batch size = 64, time steps = 30
XGBoost	n_estimators = 300, max_depth = 3, learning rate = 0.005, subsample = 0.7, colsample_bytree = 1, random state = 7, time steps = 30

4. Results and discussion

Models performance comparison

Overall model performance is summarized in Table 3. The reported values represent averages computed over the specified number of scenarios (N-SC). The relatively large values of MSE and MAE are explained by the fact that the errors are computed on price levels rather than returns, reflecting the scale of index values. Additional metrics include max drawdown (MAX-D), total return (Total-R), minimum return (Min-R) and maximum return (Max-R). Trading signals were generated by applying return and drawdown thresholds estimated over the full testing horizon, corresponding to a buy-and-hold investment from the start to the end of the test period. The reported values of return threshold $RT = 35.43\%$ and drawdown threshold $DT = -18.24\%$ represent average thresholds across all scenarios and are therefore not repeated as table columns.

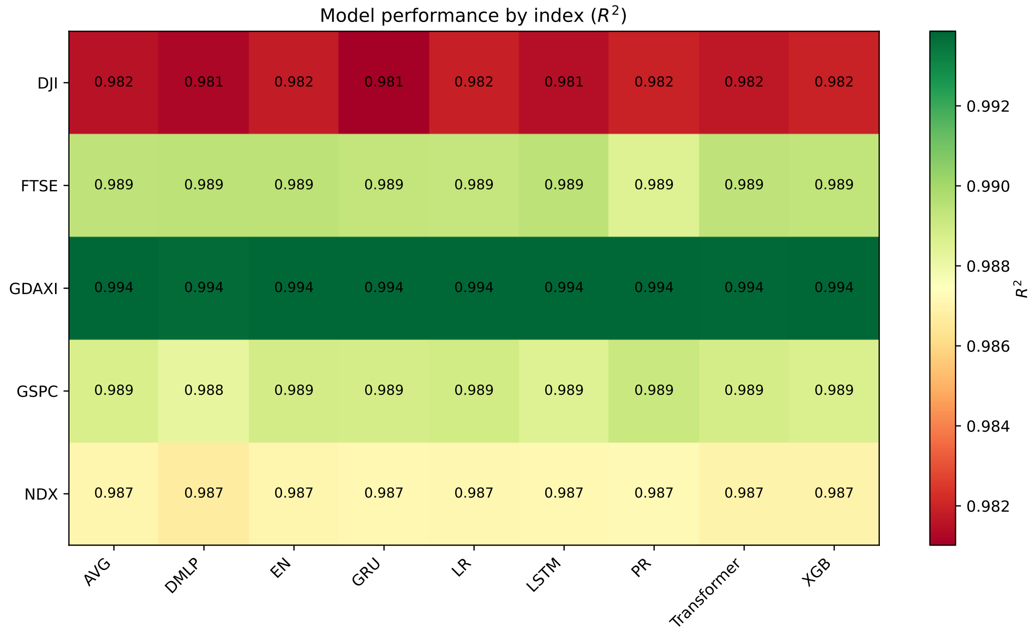

Figure 2. Heatmap of R^2 across models and indices; higher values indicate better fit.

Table 3. Comparative summary of overall performance across all evaluated models.

Algorithm	N-SC	MSE	MAE	R-SQ	Max-D	Trades	Total-R	Max-R	Min-R
LSTM	5	52,423	134.34	98.81%	-15.31%	45.00	32.18%	53.60%	13.40%
EN ensemble	130	51,966	134.31	98.82%	-15.17%	20.57	31.75%	56.78%	16.65%
GRU	5	53,091	135.06	98.80%	-15.14%	84.40	17.56%	51.02%	-2.34%
Transformer	5	52,398	134.32	98.81%	-17.97%	81.80	16.62%	50.98%	-25.03%
XGB	5	51,891	134.33	98.82%	-16.57%	36.40	15.01%	32.18%	1.52%
Average	130	52,437	134.41	98.81%	-17.82%	74.72	15.01%	66.90%	-31.52%
LR ensemble	130	51,831	134.45	98.82%	-21.30%	107.16	5.81%	46.15%	-44.63%
PR ensemble	130	51,699	134.38	98.81%	-23.84%	112.71	-0.46%	51.28%	-43.60%
DMLP	5	53,441	135.77	98.79%	-32.05%	184.80	-24.49%	-17.76%	-45.21%
Total	545	52,014	134.40	98.82%	-19.53%	79.14	12.95%	66.90%	-45.21%

The heatmap of R^2 (see Figure 2) demonstrates consistently high predictive accuracy across all models and indices, with values frequently exceeding 0.98.

In contrast, the heatmap of total returns (see Figure 3) gives a markedly different picture: while some algorithms delivered positive gains (e.g., LSTM on the S&P 500 and the transformer on the NDX), others incurred substantial losses despite exhibiting similarly strong R^2 values.

Next, we can see than the scatter plot in Figure 4 indicates only a weak relationship between R^2 and total return (Pearson's $r = 0.07$). This suggests that predictive accuracy alone does not guarantee practical usefulness, underscoring the importance of financial performance metrics.

The top-performing ensembles and single models for each index are summarized in Table 4. Although Table 3 reports averages, individual runs exhibit substantial dispersion, as illustrated in Table 4. Overall, ensemble methods (particularly those combining recurrent architectures with boosting-based models) tend to achieve the strongest performance across indices, although no single strategy dominates universally. For example, the GDAXI index attains its highest total return with the AVG_DMLP_GRU_XGB ensemble, yielding a 60% total return and a CR of 4.20, while alternative ensembles and single recurrent models remain competitive. Similarly, for the NASDAQ-100 and S&P 500, the best-performing strategies are predominantly ensemble-based, delivering total returns above 60% and 67%, respectively. In contrast, performance on the FTSE and DJIA indices is more modest and less differentiated, with multiple models exhibiting comparable outcomes. These results suggest that integrating structurally diverse predictors improves robustness and profitability in many cases, but does not guarantee superior performance across all markets.

Table 4. Comparison of the top three best-performing algorithms across indices.

Index	Algorithm	MSE	MAE	R-SQ	Max-D	Trades	Total-R	CR
GSPC	AVG_GRU_LSTM_XGB	3,289	38.22	98.87%	-16%	46	67%	1.99
GSPC	LSTM	3,341	38.25	98.85%	-19%	53	54%	1.30
GSPC	PR_DMLP_GRU_Transformer	3,092	37.86	98.94%	-9%	64	51%	2.82
NDX	AVG_Transformer_XGB	71,268	185.16	98.70%	-15%	53	61%	1.94
NDX	EN_DMLP_GRU_XGB	70,884	185.51	98.71%	-17%	33	57%	1.55
NDX	EN_GRU_LSTM_XGB	70,884	185.51	98.71%	-17%	33	57%	1.55
GDAXI	AVG_DMLP_GRU_XGB	42,094	149.07	99.38%	-7%	42	60%	4.20
GDAXI	AVG_DMLP_GRU_Transformer	42,197	149.13	99.38%	-7%	41	54%	3.83
GDAXI	GRU	41,921	148.73	99.39%	-10%	19	51%	2.49
FTSE	EN_GRU_LSTM	3,275	39.98	98.94%	-13%	11	23%	0.87
FTSE	EN_LSTM_Transformer_XGB	3,275	39.98	98.94%	-13%	11	23%	0.87
FTSE	EN_GRU_Transformer	3,275	39.98	98.94%	-13%	11	23%	0.87
DJIA	LR_GRU_Transformer_XGB	139,051	258.65	98.20%	-8%	23	32%	2.00
DJIA	LR_GRU_XGB	139,090	258.70	98.20%	-8%	23	32%	2.00
DJIA	LR_DMLP_LSTM_XGB	139,784	258.85	98.19%	-8%	25	31%	1.96

Figure 5 illustrates the capital trajectory of the best-performing model (Avg_GRU_LSTM_XGB on GSPC). The stepwise upward growth, characterized by limited drawdowns and sustained increases in both realized and unrealized capital, demonstrates the ensemble's practical viability in a trading context. This figure provides a clear illustration of how statistical accuracy can be transformed into stable and profitable trading outcomes when structurally diverse models are effectively combined.

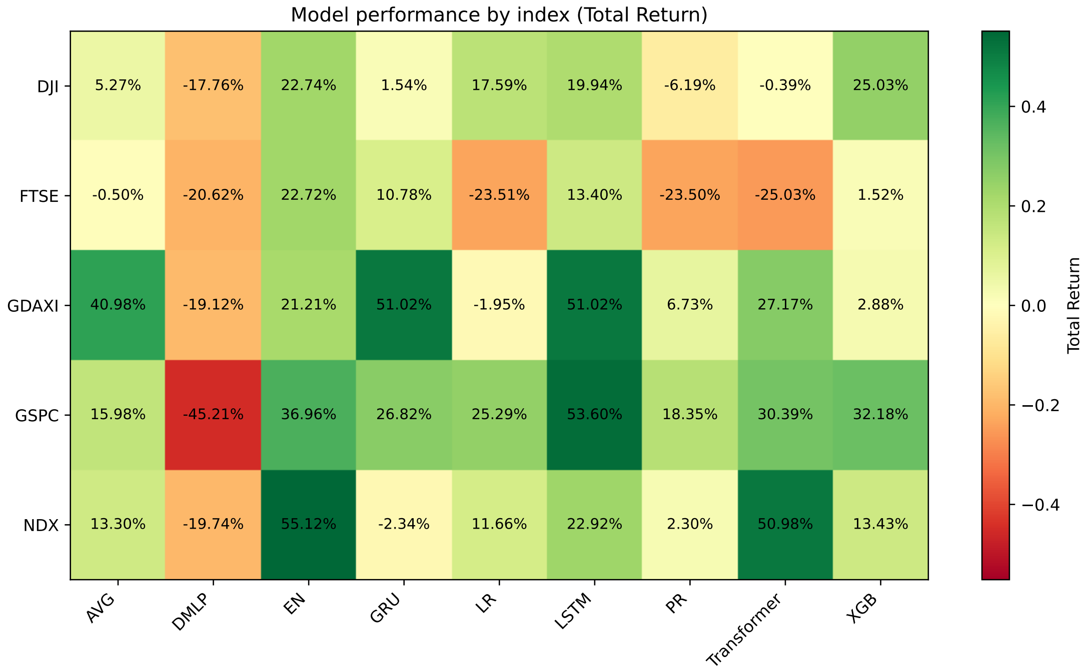

Figure 3. Heatmap of Total-R across models and indices; positive values (green) indicate profitable strategies, while negative values (red) show losses.

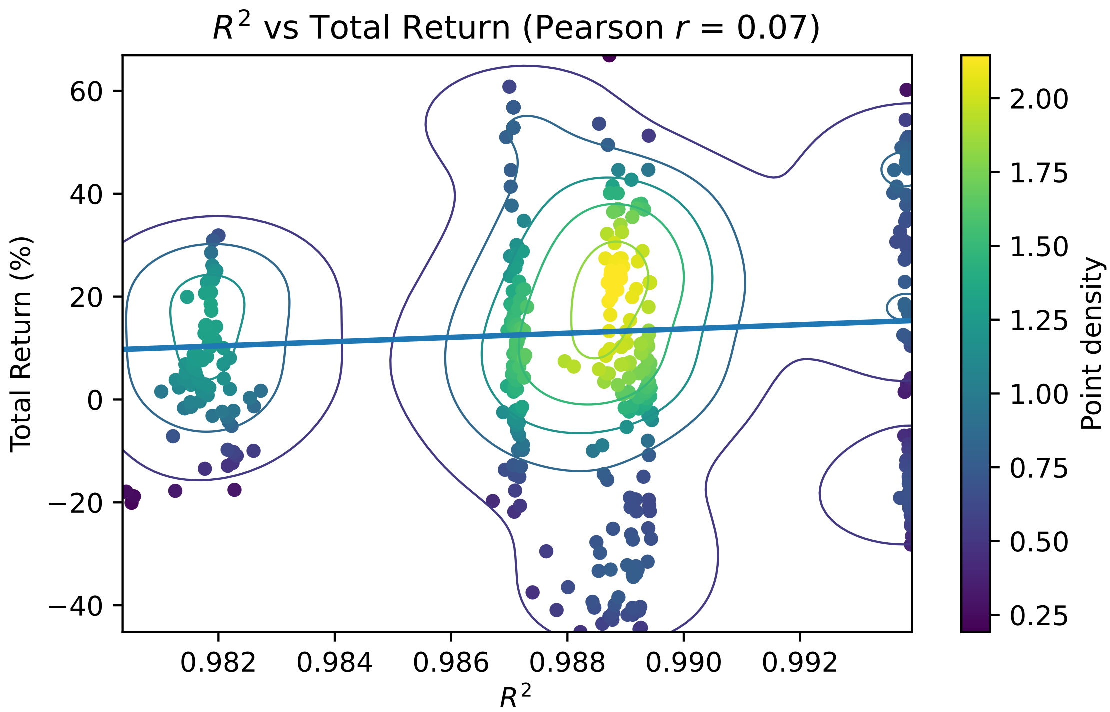

Figure 4. R^2 vs. Total-R across all runs.

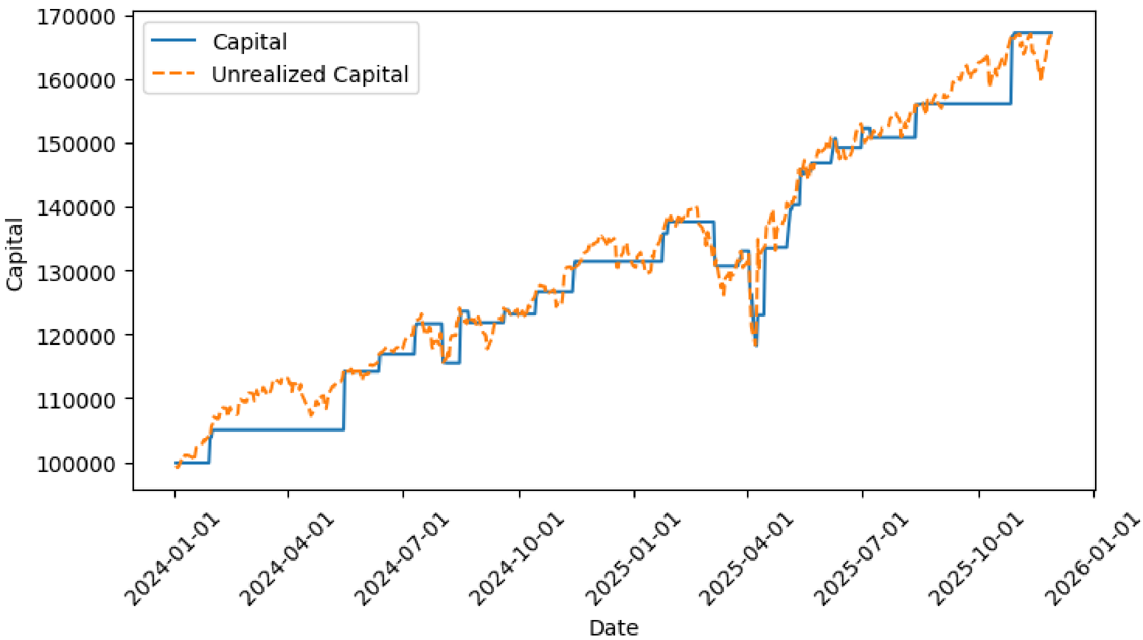

Figure 5. Capital trajectory of the best overall-performing model (Avg_GRU_LSTM_XGB on GSPC).

Discussion

The experimental results highlight a fundamental distinction between predictive accuracy and financial performance. While all models consistently achieved very high statistical accuracy ($R^2 \approx 0.98$), such values did not reliably translate into profitable trading outcomes. Several models exhibited strong predictive fit while producing weak or negative returns, indicating that accurately modeling historical price dynamics does not necessarily imply correct anticipation of return-relevant market movements.

Among the single-model approaches, LSTM generally delivered more stable profitability with moderate drawdowns across indices, whereas other models showed greater variability in trading performance. In particular, XGBoost and DMLP produced highly mixed outcomes across markets, yielding both positive and negative returns depending on the index considered. This variability highlights the sensitivity of these models to market conditions rather than consistent dominance.

Results from ensemble strategies further suggest that combining heterogeneous learners can improve robustness. Simple averaging ensembles (AVG), such as combinations of LSTM, GRU, DMLP, and XGBoost, frequently ranked among the top-performing configurations and often matched or exceeded the performance of adaptive weighting schemes based on LR, EN or polynomial ridge regression (PR). This observation indicates that straightforward aggregation of structurally diverse models may capture market dynamics more effectively than optimized but fixed weighting schemes. The underperformance of adaptive weighting schemes (LR, EN and PR) may be attributed to overfitting to the validation period, reducing robustness under changing market conditions, whereas simple averaging provides a more stable aggregation. This effect may be further amplified by regime shifts (e.g., COVID-19) and the absence of external features beyond closing prices.

Performance was strongly index-dependent. The German DAX (GDAXI) and NASDAQ-100 (NDX) exhibited comparatively higher profitability across multiple model configurations, whereas the FTSE 100 and DJIA showed weaker and less differentiated outcomes. These differences suggest that model effectiveness is influenced by structural characteristics of individual markets, including liquidity, sector composition, and volatility regimes.

Overall, the findings demonstrate that error-based metrics (MSE, MAE, R^2) are insufficient for evaluating financial forecasting models. Profitability measures (total return, drawdown, and risk-adjusted ratios) are essential for assessing the true economic relevance of predictive models.

5. Conclusions

We have investigated the effectiveness of ensemble learning techniques for enhancing financial time series forecasting. Rather than relying on individual predictive models, the proposed framework emphasizes the systematic optimization of model combinations, leading to improved predictive accuracy and robustness. In contrast to conventional approaches that primarily evaluate forecasting performance using statistical error metrics, we incorporate financial performance–based evaluation criteria. Specifically, measures, such as MDD, TR, and CR, are employed to assess real-world profitability and risk-adjusted performance. Furthermore, to refine ensemble predictions, we adopt an optimization-driven weight allocation strategy. LR, ridge regression, and EN models with polynomial features are used to determine optimal weighting schemes, enabling adaptability to changing market conditions while mitigating forecasting biases. The main conclusions can be summarized as follows:

- **Accuracy is not profitability.** Very high R^2 values did not translate into positive trading performance. Models may capture statistical variance but fail to predict price direction, making financial metrics indispensable for evaluation.
- **Need for financial validation.** Prior studies often focused on statistical accuracy (R^2, MSE, MAE), however our results show these metrics alone are insufficient. Profitability measures are essential to assess real-world applicability.
- **Ensemble benefits.** Hybrid ensembles of recurrent networks (LSTM, GRU), feedforward deep learning (DMLP), and boosting models (XGBoost) frequently ranked among the top-performing configurations, capturing complementary aspects of market dynamics.
- **Limitations of adaptive weighting.** Linear weighting methods (LR, EN and PR) did not consistently improve performance over simple averaging, indicating limited robustness of such schemes in volatile environments.
- **Index dependence.** Forecasting performance varied substantially by market. GSPC and NDX produced the most reliable profitability, while FTSE and DJIA lagged, reflecting differences in market structure and volatility.
- **Practical implications.** Machine learning ensembles show strong potential for financial forecasting, but effective deployment requires alignment of predictive accuracy with financial performance metrics, as well as careful risk management to limit downside exposure.

Future research directions. A key limitation of the present study is the reliance exclusively on historical closing prices as the predictive signal. While this setting provides a controlled environment for model comparison, it restricts the available information about market dynamics and may limit the models' ability to accurately predict price direction. This limitation may partly explain the observed discrepancy between high statistical accuracy and weaker trading profitability. Future research will therefore focus on two main directions. First, portfolio formation across multiple assets should be investigated, shifting the emphasis from single-index forecasting to multi-asset allocation. This will allow models to exploit diversification benefits, capture inter-market dependencies, and evaluate ensemble methods in a realistic investment context. Second, incorporation of additional predictive features, including intraday indicators, macroeconomic variables, sentiment measures, and volatility indices, will be explored to enrich the input space. By integrating heterogeneous signals, the models can potentially improve directional accuracy, reduce exposure to noise, and align predictions more closely with actionable market movements. Thus, future work will move beyond univariate closing-price prediction toward multi-dimensional, portfolio-oriented forecasting, with the aim of enhancing both predictive robustness and financial profitability. Moreover, it should involve a more rigorous statistical assessment of model performance, including significance testing (e.g., the Diebold–Mariano test for forecast accuracy, paired *t*-tests or bootstrap procedures for return differentials), alongside sensitivity analyses with respect to trading thresholds and transaction costs.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

The authors would like to thank the anonymous reviewers for carefully reading the manuscript and providing constructive comments and suggestions, which have helped to improve the quality of the paper.

Conflict of interest

The authors declare there is no conflict of interest.

Author contributions

Conceptualization: A.B. and I.B.; methodology, A.B. and I.B.; software, A.B.; validation, A.B. and I.B.; investigation, A.B.; writing-original draft preparation, A.B.; writing-review and editing, A.B. and I.B.; supervision, I.B.; All authors have read and agreed to the published version of the manuscript.

Data availability statement

The dataset analyzed in this work is publicly available on GitHub [21] and requests for additional information should be addressed to the corresponding author.

References

1. A. Timmermann, C. W. J. Granger, Efficient market hypothesis and forecasting, *Int. J. Forecast.*, **20** (2004), 15–27. https://doi.org/10.1016/S0169-2070(03)00012-8

2. M. C. Jensen, Some anomalous evidence regarding market efficiency, *J. Financ. Econ.*, **6** (1978), 95–101. https://doi.org/10.1016/0304-405X(78)90025-9

3. K. Olorunnimbe, H. L. Viktor, Ensemble of temporal transformers for financial time series, *J. Intell. Inf. Syst.*, **62** (2024), 1–25. https://doi.org/10.1007/s10844-024-00851-2

4. W. Chen, Z. Liu, L. Jia, A hybrid approach for portfolio construction: Combining two-stage ensemble forecasting model with portfolio optimization, *Comput. Intell.*, **40** (2024), e12617. https://doi.org/10.1111/coin.12617

5. Y. Ma, R. Han, W. Wang, Portfolio optimization with return prediction using deep learning and machine learning, *Expert Syst. Appl.*, **165** (2021), 113973. https://doi.org/10.1016/j.eswa.2020.113973

6. A. Dezhkam, M. T. Manzuri, Forecasting stock market for an efficient portfolio by combining XGBoost and Hilbert–Huang transform, *Eng. Appl. Artif. Intell.*, **118** (2023), 105626. https://doi.org/10.1016/j.engappai.2022.105626

7. S. Daul, T. Jaisson, A. Nagy, Performance attribution of machine learning methods for stock returns prediction, *J. Finance Data Sci.*, **8** (2022), 86–104. https://doi.org/10.1016/j.jfds.2022.04.002

8. B. Gezici, E. Sefer, Deep transformer-based asset price and direction prediction, *IEEE Access*, **12** (2024), 24164–24178. https://doi.org/10.1109/ACCESS.2024.3358452

9. J. K. Mutinda, A. K. Langat, Stock price prediction using combined GARCH-AI models, *Sci. Afr.*, **26** (2024), e02374. https://doi.org/10.1016/j.sciaf.2024.e02374

10. T. Yue, Y. Liu, Multi-scale price forecasting based on data augmentation, *Appl. Sci.*, **14** (2024), 8737. https://doi.org/10.3390/app14198737

11. J. Huang, Y. Wang, Comparative analysis of different machine learning techniques in forecasting stock price, in *Proceedings of the International Conference on Artificial Intelligence Innovation (ICAII)*, (2023), 50–64. https://doi.org/10.1109/ICAII59460.2023.10497212

12. K. Y. Yan, Z. H. Yue, C. C. Wu, Q. Q. He, J. M. Zhou, Z. H. Hao, et al., Flexible target prediction for quantitative trading in the American stock market: A hybrid framework integrating ensemble models, fusion models and transfer learning, *Entropy*, **28** (2026), 84. https://doi.org/10.3390/e28010084

13. A. Hayati, S. S. Gharehveran, K. Shirini, Electricity price forecasting with ensemble meta-models and SHAP explainers: A PCA-driven approach, *Sci. Rep.*, **16** (2026), 6466. https://doi.org/10.1038/s41598-026-35839-1

14. A. Brusaferri, A. Ballarino, L. Grossi, F. Laurini, On-line conformalized neural networks ensembles for probabilistic forecasting of day-ahead electricity prices, *Appl. Energy*, **398** (2025), 126412. https://doi.org/10.1016/j.apenergy.2025.126412

15. Y. Choi, D. Kim, J. Lee, Temporal consistency ensemble empirical mode decomposition for forecasting practical metal price, *Eng. Appl. Artif. Intell.*, **158** (2025), 111490. https://doi.org/10.1016/j.engappai.2025.111490

16. Yahoo Finance, S&P 500 index. Available from: https://finance.yahoo.com/quote/%5EGSPC/.

17. Yahoo Finance, NASDAQ-100 index. Available from: https://finance.yahoo.com/quote/%5ENDX/.

18. Yahoo Finance, Dow Jones industrial average. Available from: https://finance.yahoo.com/quote/%5EDJI/.

19. Yahoo Finance, FTSE 100 index. Available from: https://finance.yahoo.com/quote/%5EFTSE/.

20. Yahoo Finance, DAX performance index. Available from: https://finance.yahoo.com/quote/%5EGDAXI/.

21. A. Bielskis, Machine learning financial dataset repository. Available from: https://github.com/AivarasBi/ML.

22. S. Hochreiter, J. Schmidhuber, Long short-term memory, *Neural Comput.*, **9** (1997), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

23. J. B. Heaton, N. G. Polson, J. H. Witte, Deep learning for finance: Deep portfolios, *Appl. Stochastic Model Bus. Ind.*, **33** (2017) 3–12. https://doi.org/10.1002/asmb.2209

24. S. I. Lee, S. J. Yoo, Threshold-based portfolio: The role of the threshold and its applications, *J. Supercomput.*, **76** (2020), 8040–8057. https://doi.org/10.1007/s11227-018-2577-1

25. A. Bielskis, I. Belovas, Comparative analysis of stock price ARIMA and LSTM forecasting methods, *Proc. Lith. Math. Soc.*, **63** (2022). https://doi.org/10.15388/LMR.2022.29755

26. J. Chung, C. Gulcehre, K. Cho, Y. Bengio, Empirical evaluation of gated recurrent neural networks on sequence modeling, preprint, arXiv: 1412.3555, 2014. https://doi.org/10.48550/arXiv.1412.3555

27. W. Wu, W. Liao, J. Miao, G. Du, Using gated recurrent unit network to forecast short-term load considering impact of electricity price, *Energy Procedia*, **158** (2019), 3369–3374. https://doi.org/10.1016/j.egypro.2019.01.950

28. T. S. Mian, Evaluation of stock closing prices using transformer learning, *Eng. Technol. Appl. Sci. Res.*, **13** (2023), 11635–11642. https://doi.org/10.48084/etasr.6017

29. W. Chen, H. Zhang, M. K. Mehlawat, L. Jia, Mean–variance portfolio optimization using machine learning-based stock price prediction, *Appl. Soft Comput.*, **100** (2021), 106943. https://doi.org/10.1016/j.asoc.2020.106943

30. L. O. Orimoloye, M. C. Sung, T. Ma, J. E. V. Johnson, Comparing the effectiveness of deep feedforward neural networks and shallow architectures for predicting stock price indices, *Expert Syst. Appl.*, **139** (2020), 112828. https://doi.org/10.1016/j.eswa.2019.112828

31. R. Singh, S. Srivastava, Stock prediction using deep learning, *Multimed. Tools Appl.*, **76** (2016), 18569–18584. https://doi.org/10.1007/s11042-016-4159-7

32. M. Magdon-Ismail, A. F. Atiya, A. Pratap, Y. S. Abu Mostafa, On the maximum drawdown of a Brownian motion, *J. Appl. Probab.*, **41** (2004), 99–102. https://doi.org/10.1239/jap/1077134674