Archives available at journals.mriindia.com

International Journal of Advanced Scientific Research and Engineering Trends

ISSN: 2456-0774

Volume 10 Issue 05, 2026

BullBear AI Pro: A Hybrid Real-Time Framework for Stock and Cryptocurrency Trend Prediction Using Random Forest and Long Short-Term Memory Networks

[1]Ekansh Tayade, [2]Pranav Pagare, [3]Darshan Aher, [4]Smit Chaudhari

[1,2,3,4] *Dept. of Computer Engineering,Sandip Institute of Technology and Research Center Nashik (SITRC), Dharmsinh Desai University Nadiad, Gujarat (DDU)*

[1]*ekanshshweta06@gmail.com*, [2]*pranavnpagare@gmail.com*, [4]*smitchaudhari2601@gmail.com*

Peer Review Information

Submission: 16 April 2026
Revision: 02 May 2026
Acceptance: 20 May 2026

Keywords

Stock Market Prediction, Cryptocurrency Forecasting, Random Forest, LSTM, Ensemble Learning, Technical Indicators, Real-Time Analytics, Financial Machine Learning.

Abstract

Financial markets are highly dynamic, nonlinear, and influenced by a wide range of economic, political, and behavioral factors, making short-term trend prediction one of the most challenging problems in computational finance. The continuous interaction of macroeconomic indicators, company-specific developments, investor sentiment, and unexpected global events produces noisy and non-stationary time series that are difficult to model using conventional statistical techniques. Although traditional forecasting methods and rule-based technical analysis remain widely used, they often fail to capture complex temporal dependencies and nonlinear relationships present in stock and cryptocurrency price movements. In recent years, machine learning and deep learning approaches have demonstrated significant potential for extracting predictive patterns directly from historical market data and supporting more informed investment decisions. This paper presents BullBear AI Pro, a real-time hybrid forecasting framework that combines a Random Forest classifier and a Long Short-Term Memory (LSTM) network to predict next-period market direction (UP or DOWN). Historical market data are automatically retrieved from Yahoo Finance and processed through a feature engineering pipeline that computes widely used technical indicators, including moving averages, Relative Strength Index (RSI), Moving Average Convergence Divergence (MACD), and Bollinger Bands. Predictions generated by both models are integrated using an ensemble decision mechanism that produces confidence-based BUY, SELL, and HOLD recommendations. The framework also includes an interactive dashboard developed using Streamlit, enabling users to train models, monitor multiple stocks and cryptocurrencies, and obtain live predictions in real time. Experimental evaluation was conducted using approximately ten years of historical data for Apple, Amazon, Microsoft, and Tesla. The results show that the LSTM model consistently outperformed the Random Forest classifier, achieving an average accuracy of 53.43% and an average F1-score of 69.63%, compared with 48.42% accuracy and 36.41% F1-score for the Random Forest model. These findings demonstrate that temporal deep learning models are more effective in capturing market dynamics, while the proposed hybrid framework provides a practical, scalable, and deployable decision support system for real-time stock and cryptocurrency forecasting.

Introduction

Financial markets are widely recognized as one of the most complex and unpredictable environments for data analysis and forecasting. The prices of stocks and cryptocurrencies evolve continuously under the influence of numerous interconnected factors, including macroeconomic conditions, inflation trends, interest rate decisions, company earnings, corporate governance, geopolitical conflicts, regulatory changes, investor psychology, and large-scale institutional trading activity. The interaction of these variables creates financial time series that are highly nonlinear, noisy, and non-stationary, meaning that underlying statistical relationships change over time and may shift abruptly during periods of heightened volatility. This dynamic nature makes short-term trend prediction a particularly challenging problem in computational finance and artificial intelligence. Accurate forecasts of market direction can provide significant practical value by enabling investors and portfolio managers to improve trading strategies, optimize asset allocation, reduce risk exposure, and make more informed financial decisions. For decades, researchers and practitioners have relied on traditional forecasting methods such as moving averages, autoregressive integrated moving average (ARIMA), and generalized autoregressive conditional heteroskedasticity (GARCH) models to analyze market behavior. While these methods have contributed to financial econometrics, they are fundamentally limited by assumptions of linearity and stationarity, which often fail to hold in real-world market environments characterized by sudden structural changes and complex nonlinear dependencies [1]. As the volume and availability of historical market data have increased, machine learning techniques have emerged as powerful alternatives capable of learning intricate relationships directly from data without requiring restrictive assumptions about the underlying process [2]. Recent surveys confirm that artificial intelligence has become one of the most active and promising areas of financial research, with numerous studies reporting improved predictive performance over conventional statistical approaches and highlighting the growing importance of intelligent decision support systems in modern trading environments.

Among machine learning algorithms, Random Forest has become a widely adopted method for financial classification tasks because of its robustness, interpretability, and strong generalization capability. Random Forest constructs a collection of decision trees trained on bootstrap samples and aggregates their predictions to produce stable and reliable outputs, reducing the risk of overfitting while effectively handling noisy and partially redundant features [4]. This makes the algorithm particularly well suited for market prediction problems where engineered variables such as moving averages, momentum indicators, and volatility measures are used to capture hidden market behavior. However, Random Forest treats each observation as an independent feature vector and does not explicitly model the sequential dependencies that are central to financial time series. To address this limitation, deep learning models—especially Long Short-Term Memory (LSTM) networks—have gained considerable attention. LSTM is a specialized recurrent neural network architecture designed to preserve long-term contextual information through gated memory cells, allowing the model to learn temporal relationships that span many time steps. This capability is particularly important in financial markets, where current price movements are often influenced by trends and patterns established over extended historical periods. Multiple empirical studies have shown that LSTM-based models outperform traditional machine learning algorithms in stock trend prediction, especially when combined with technical indicators and large historical datasets [6]. At the same time, researchers have increasingly emphasized the benefits of hybrid and ensemble approaches that integrate the strengths of multiple predictive models to improve robustness and reduce model-specific weaknesses. Nevertheless, many published systems remain limited to offline experimentation, focus on a single asset or dataset, and do not provide actionable trading recommendations or real-time deployment capabilities. As a result, there remains a significant need for end-to-end forecasting platforms that unify data acquisition, preprocessing, feature engineering, model training, ensemble inference, confidence estimation, and interactive visualization within a practical and reproducible framework [9].

To address these limitations, this paper proposes BullBear AI Pro, a hybrid real-time forecasting framework for stock and cryptocurrency trend prediction that combines the complementary strengths of Random Forest and Long Short-Term Memory networks. The proposed system automatically retrieves historical and live market data from Yahoo Finance, computes a diverse set of technical indicators—including moving averages, Relative Strength Index (RSI), Moving Average Convergence Divergence (MACD), and Bollinger Bands—and transforms the resulting information into supervised learning representations suitable for both machine learning and deep learning models. The Random Forest classifier captures nonlinear interactions among engineered features, while the LSTM network models sequential dependencies and temporal patterns across historical closing prices. Predictions from both models are integrated using an ensemble decision mechanism that calculates a final directional forecast and generates confidence-based BUY, SELL, and HOLD recommendations to support practical trading decisions.

Fig 1: - Conceptual Overview of the Challenges and Proposed Hybrid Framework for Real-Time Stock and Cryptocurrency Trend Prediction

To enhance usability and real-world applicability, the framework is deployed through an interactive web dashboard developed using Streamlit, enabling users to train models, monitor multiple stocks and cryptocurrencies, and obtain live predictions in real time. Experimental evaluation was conducted using approximately ten years of daily historical data for Apple, Amazon, Microsoft, and Tesla, representing diverse market behaviours and volatility profiles [6]. The results demonstrate that the LSTM model consistently outperforms the Random Forest classifier, while the integrated platform provides a scalable, reproducible, and deployable decision support system for intelligent financial forecasting [15].

Literature Review

The prediction of financial market movements has been one of the most extensively investigated problems in quantitative finance, econometrics, and artificial intelligence. Researchers have long sought to develop models capable of anticipating stock and cryptocurrency price direction to support portfolio optimization, algorithmic trading, and risk management. The difficulty of this task stems from the fact that financial markets are influenced by a wide range of interacting variables, including macroeconomic indicators, monetary policy decisions, corporate earnings, regulatory changes, geopolitical developments, market liquidity, and investor sentiment. These influences generate highly nonlinear, noisy, and non-stationary time series in which underlying statistical relationships evolve over time and may change abruptly during periods of economic uncertainty or market stress. Early forecasting research relied on statistical and econometric methods such as moving averages, autoregressive integrated moving average (ARIMA), generalized autoregressive conditional heteroskedasticity (GARCH), and exponential smoothing. These techniques established an important theoretical foundation and remain useful for modelling trend and volatility under stable conditions; however, their predictive performance is often constrained by assumptions of linearity, stationarity, and fixed model structures that rarely hold in real-world markets. As computational resources became more accessible and historical market data expanded dramatically, researchers increasingly adopted machine learning techniques capable of learning complex nonlinear relationships directly from data without imposing restrictive assumptions on the underlying process [2]. Comprehensive review studies have confirmed that artificial intelligence has become one of the most active and promising areas of financial research, with numerous investigations demonstrating that machine learning and deep learning models can outperform traditional statistical approaches across multiple markets, asset classes, and forecasting horizons [3]. These reviews also emphasize that forecasting success depends on a combination of high-quality data preprocessing, carefully engineered features, robust validation strategies, and appropriate algorithm selection rather than on model choice alone. This shift from purely statisticalmodellingmodelling to intelligent data-driven systems has laid the foundation for modern financial forecasting research and has motivated the development of increasingly sophisticated predictive frameworks.

Among the wide range of machine learning algorithms explored in the literature, Random Forest has emerged as one of the most robust and effective approaches for directional market prediction. Random Forest is an ensemble learning technique that constructs many decision trees using bootstrap samples and random feature selection, then aggregates their predictions to produce stable and generalizable outputs. This design significantly reduces overfitting and enables the model to handle noisy observations, multicollinearity, and partially redundant variables—characteristics that are common in financial datasets constructed from numerous technical indicators. Khaidem et al. demonstrated that Random Forest trained on indicators such as Relative Strength Index (RSI), stochastic oscillators, and moving averages could effectively predict stock market direction and outperform several traditional forecasting methods [6]. Ballings et al. compared logistic regression, neural networks, support vector machines, and ensemble tree-based methods, concluding that Random Forest and related ensemble techniques consistently delivered strong predictive performance and superior generalization across different market conditions [19].

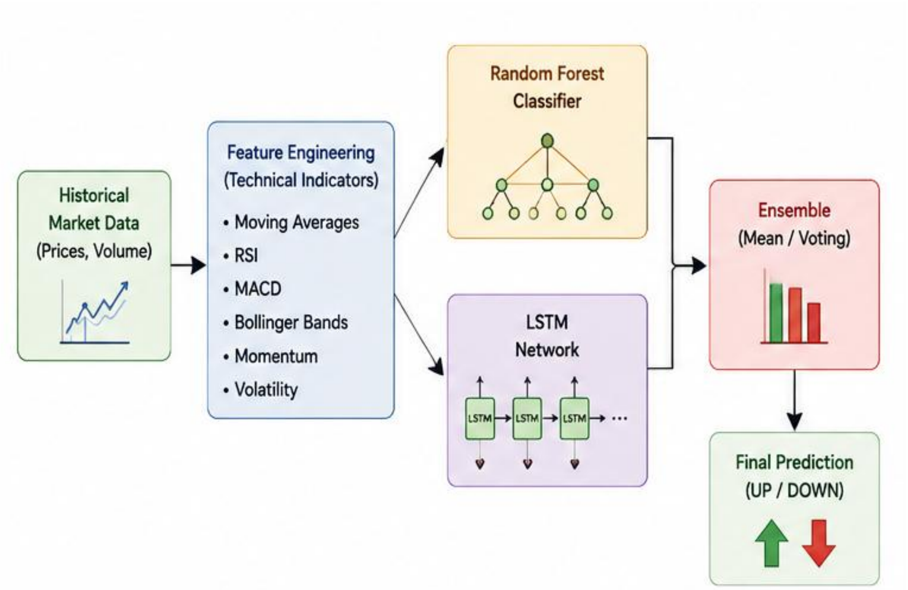

Fig 2: Hybrid Model Architecture Combining Random Forest and LSTM for Financial Trend Prediction

Subsequent studies have reinforced these findings by showing that Random Forest offers a valuable combination of interpretability, computational efficiency, and predictive accuracy when applied to feature-rich financial datasets [23]. Nevertheless, despite its many advantages, Random Forest fundamentally treats each observation as an independent feature vector and does not explicitly model sequential dependencies across time. This limitation is particularly significant in financial markets, where momentum effects, trend persistence, delayed responses to external information, and cyclical patterns often unfold over extended periods rather than being fully captured by contemporaneous features alone [1]. Consequently, while Random Forest remains a strong baseline and an important component of hybrid systems, researchers have increasingly sought models that can more effectively capture temporal relationships.

The need to model long-range dependencies in sequential data led to the widespread adoption of deep learning architectures, especially Long Short-Term Memory (LSTM) networks. LSTM is a specialized recurrent neural network architecture specifically designed to overcome the vanishing gradient problem by incorporating gated memory cells that regulate the retention, updating, and forgetting of information over time [10]. This architecture enables LSTM to learn both short-term and long-term dependencies from historical price sequences, making it particularly suitable for financial forecasting where current market behaviour is often influenced by patterns established over weeks, months, or even years. Fischer and Krauss presented one of the earliest and most influential demonstrations of deep learning in finance, showing that LSTM-based models could generate statistically and economically meaningful forecasts of stock returns [5]. Nabipour et al. later compared multiple machine learning and deep learning methods and reported that LSTM consistently achieved superior performance in directional forecasting across several industry sectors [12]. Since then, a rapidly expanding body of research has introduced enhanced LSTM-based architectures incorporating convolutional neural networks, attention mechanisms, feature fusion modules, and transformer-inspired components, further improving predictive performance and robustness [23]. Despite these advances, deep learning models typically require larger datasets, longer training times, and careful hyperparameter optimization, and their internal decision processes may be less interpretable than those of tree-based models [14]. These trade-offs have motivated the development of hybrid frameworks that combine the temporal modelling strength of LSTM with the interpretability and feature-handling capabilities of conventional machine learning algorithms.

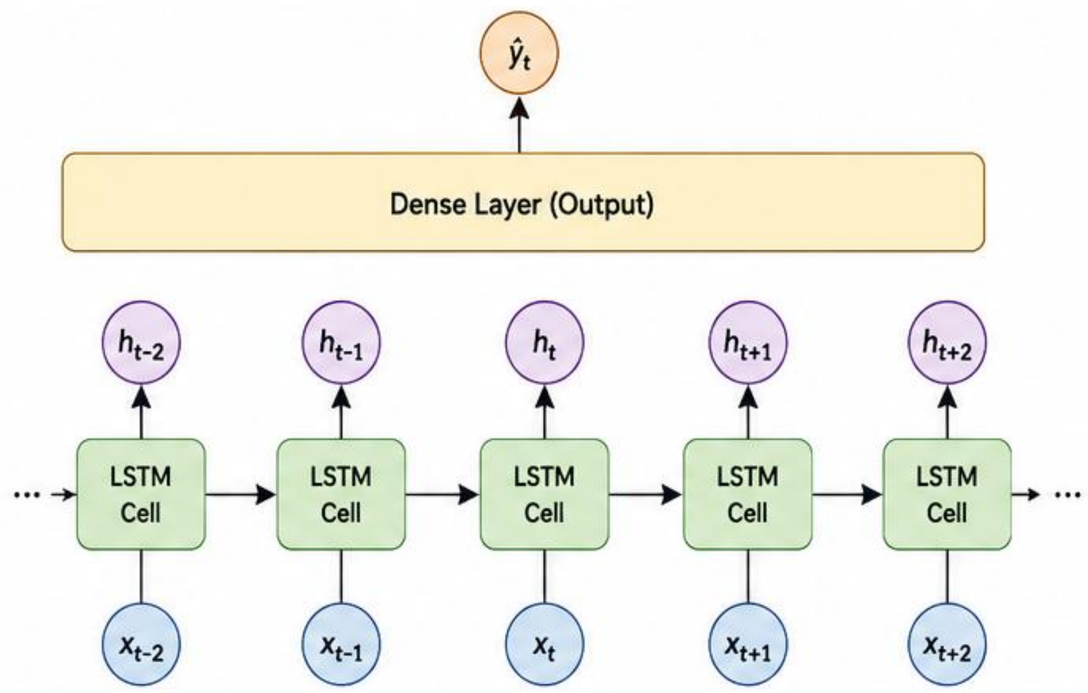

Fig 3: Long Short-Term Memory (LSTM) Network Structure for Sequential Time-Series Modelling

In addition to model architecture, feature engineering has played a central role in every successful stock market forecasting study. Technical indicators derived from historical price and volume data provide compact representations of trend strength, momentum, volatility, and overbought or oversold conditions. Widely used indicators include Simple and Exponential Moving Averages, Relative Strength Index (RSI), Moving Average Convergence Divergence (MACD), Bollinger Bands, Average True Range, rolling volatility, and momentum measures. These indicators have been extensively validated in both academic research and professional trading practice and are frequently used to transform raw financial data into structured representations that are more informative for machine learning and deep learning algorithms [16]. Alongside feature engineering, ensemble learning has become an increasingly important strategy for improving predictive robustness. Ensemble methods combine multiple models to reduce variance, compensate for model-specific weaknesses, and produce more stable and reliable forecasts. In financial applications, hybrid systems that integrate feature-based machine learning models with sequence-based deep learning networks have demonstrated particularly strong performance because they capture complementary aspects of market behaviour Several studies report that ensemble architectures outperform standalone models, especially when probabilistic outputs and confidence measures are incorporated into trading decisions and uncertainty-aware recommendations [19]. These findings strongly suggest that the combination of technical indicators, hybrid modelling, and ensemble decision-making represents one of the most effective paradigms for intelligent financial forecasting.

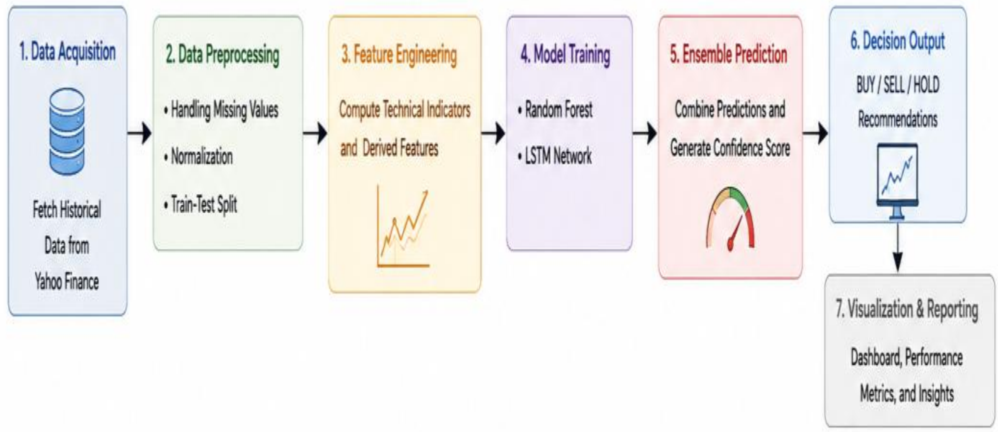

Fig 4: Conceptual Overview of the Challenges and Proposed Hybrid Framework for Real-Time Stock and Cryptocurrency Trend Prediction

Research Gap and Problem Statement

Despite the substantial progress documented in the literature, significant gaps remain between academic forecasting research and deployable real-world systems. A considerable proportion of published studies are restricted to offline experiments and do not address practical issues such as live data acquisition, continuous prediction, automated confidence estimation, and interactive visualization. Many investigations focus on a single stock, index, or limited dataset, which constrains the generalizability of their conclusions across different assets and market regimes [25]. Other studies emphasize statistical performance metrics while neglecting the transformation of predictions into actionable BUY, SELL, and HOLD recommendations that can directly support investment decisions [21]. Moreover, integrated dashboards that allow users to train models, monitor multiple financial instruments, and download performance reports are seldom included in academic prototypes [2]. As a result, although the literature has produced a wide variety of powerful predictive algorithms, relatively few studies offer a complete end-to-end framework that unifies automated data acquisition, technical feature engineering, hybrid model training, ensemble inference, confidence-based decision support, and real-time deployment within a single reproducible architecture [13]. The present work addresses this gap through the development of BullBear AI Pro, a hybrid real-time forecasting platform that combines Random Forest and Long Short-Term Memory networks to predict stock and cryptocurrency trends using live market data and engineered technical indicators [14]. By integrating ensemble reasoning, confidence-aware recommendations, and an interactive Streamlit dashboard, the proposed system extends prior research and demonstrates a practical, scalable, and deployment-oriented solution for intelligent financial forecasting and decision support [5].

precision, recall, and F1-score are routinely reported, few systems translate these outputs into actionable BUY, SELL, and HOLD recommendations that can directly support practical investment decisions [5].

The literature also reveals significant gaps in usability, interpretability, and deployment-oriented design. Most research prototypes are implemented as standalone scripts or notebook-based experiments intended solely for academic evaluation. Such implementations lack interactive dashboards, real-time visualization, downloadable reports, and user-configurable parameters such as confidence thresholds or asset selection [6]. Confidence estimation is another underexplored area; many studies provide binary forecasts without quantifying prediction certainty, making it difficult for users to assess the reliability of model outputs and manage decision risk [7]. In the context of cryptocurrency forecasting, these limitations are even more pronounced due to the high volatility and continuous trading nature of digital asset markets. Although hybrid and ensemble models have demonstrated promising performance improvements, comparatively few studies integrate automated data acquisition, technical indicator generation, model training, ensemble prediction, confidence-based recommendation, and live deployment into a single reproducible framework. This disconnects between algorithmic research and deployable decision support systems highlights a clear opportunity for the development of practical, scalable, and user-centric forecasting platforms capable of operating in real-world financial environments [10].

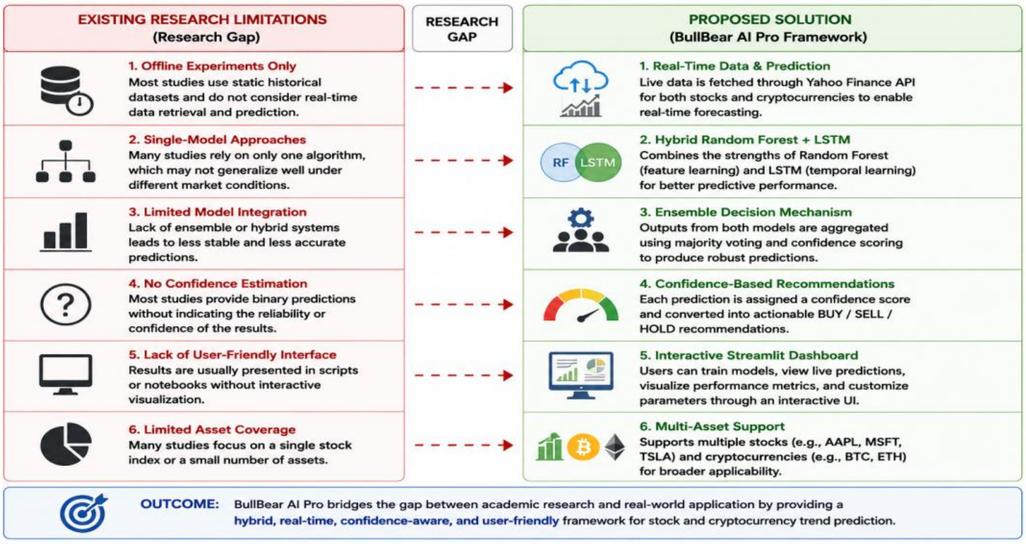

Fig 5: Existing Limitations in Financial Forecasting Systems and the Proposed Solution in BullBear AI Pro

To address these limitations, this research proposes BullBear AI Pro, a hybrid real-time stock and cryptocurrency trend prediction system that combines Random Forest and Long Short-Term Memory (LSTM) networks within an ensemble decision architecture. The system is designed to bridge the gap between theoretical forecasting research and practical implementation by integrating all stages of the predictive pipeline into a unified framework. Historical and live market data are automatically retrieved from Yahoo Finance, transformed into predictive features using widely adopted technical indicators, and processed by both models to generate probabilistic forecasts [21]. These outputs are aggregated to produce a final directional prediction and confidence-based BUY, SELL, and HOLD recommendations. The framework is deployed through an interactive Streamlit dashboard that enables users to select multiple assets, train models, generate live predictions, visualize performance metrics, and download analytical reports [12]. By combining hybrid modelling, ensemble reasoning, confidence-aware decision support, and real-time deployment, the proposed system addresses the principal shortcomings identified in prior studies and provides a scalable and deployable solution for intelligent financial forecasting [13].

Proposed Methodology

The proposed methodology, implemented in the form of BullBear AI Pro, is designed as a comprehensive end-to-end framework for real-time stock and cryptocurrency trend prediction. The central objective of the methodology is to transform raw financial market data into actionable investment intelligence by integrating automated data acquisition, advanced feature engineering, machine learning and deep learning-based predictive modelling, ensemble decision making, confidence-aware recommendation generation, and interactive deployment within a unified architecture. The framework is specifically developed to predict the short-term directional movement of a financial asset, represented as either UP or DOWN, and to convert this forecast into practical BUY, SELL, or HOLD recommendations. Unlike many existing studies that evaluate individual algorithms under isolated experimental conditions, the proposed methodology adopts a hybrid strategy that combines a Random Forest classifier and a Long Short-Term Memory (LSTM) network in order to exploit the complementary strengths of feature-based statistical learning and sequence-based temporal modelling. Random Forest is highly effective in capturing nonlinear interactions among engineered technical indicators, while LSTM is particularly well suited for modelling sequential dependencies and long-range temporal patterns in financial time series [2]. By integrating these two models within an ensemble framework, the methodology seeks to achieve greater predictive stability, improved robustness across different market regimes, and enhanced practical applicability compared with single-model approaches. The entire system operates on both historical and live market data automatically retrieved from Yahoo Finance, thereby enabling the same methodological pipeline to be applied seamlessly to equities and cryptocurrencies without requiring structural modifications. The resulting framework is not merely a predictive model, but a complete financial decision support system that unifies data ingestion, preprocessing, model training, probabilistic forecasting, confidence estimation, recommendation generation, and user interaction within a reproducible and deployment-ready platform [21].

The methodology begins with automated data acquisition and preprocessing, which establish the foundation for all subsequent modelling stages. For each selected asset, the system retrieves approximately ten years of historical daily market data, including open, high, low, close, adjusted close, and trading volume information. This extended time horizon is chosen to ensure that the models are exposed to a wide range of market conditions, including bull markets, bearish corrections, periods of elevated volatility, and macroeconomic shocks. Once downloaded, the raw dataset undergoes a rigorous preprocessing procedure in which missing observations are handled, data types are standardized, and derived variables such as percentage returns are computed. The predictive target is formulated as a binary trend label defined by comparing the next closing price with the current closing price: a value of 1 indicates that the price is expected to increase in the next trading period, while a value of 0 denotes a decrease or no upward movement. To preserve the temporal integrity of the financial series and avoid look-ahead bias, the data are partitioned chronologically into training and testing subsets using an 80:20 split, ensuring that only past observations are used to predict future outcomes. This time-aware partitioning strategy closely mirrors real-world trading conditions and provides a more realistic assessment of model performance than randomly shuffled datasets. Following preprocessing, the framework applies a comprehensive feature engineering pipeline that computes a set of widely used technical indicators, including Simple Moving Averages over 10 and 20 periods, Relative Strength Index (RSI), Moving Average Convergence Divergence (MACD), Bollinger Bands, and percentage price change. These indicators are specifically selected because they summarize essential market characteristics such as trend direction, momentum, volatility, and overbought or oversold conditions, and have consistently demonstrated predictive relevance in financial forecasting studies [10]. For the Random Forest model, these engineered indicators are used directly as structured input features. For the LSTM model, the closing price series is normalized and transformed into overlapping sequences of 30 consecutive trading days, enabling the network to learn temporal dependencies and sequential patterns across rolling windows of historical observations [1]. This dual representation of the data—combining handcrafted technical features and raw sequential information—ensures that the methodology captures both domain-specific market signals and latent temporal dynamics.

Once the data have been transformed into model-ready representations, the methodology proceeds to the model development and ensemble decision stages. The Random Forest classifier is trained using the engineered technical indicators and produces class probabilities representing the likelihood of upward and downward market movement. Because Random Forest aggregates the outputs of numerous decision trees, it offers strong resistance to overfitting and robust performance in the presence of noisy and partially redundant features. In parallel, the Long Short-Term Memory network is trained on normalized sequential data using recurrent layers and dense output layers optimized with the Adam algorithm. The LSTM model learns hidden temporal relationships that may span several weeks or months, enabling it to detect momentum patterns, trend persistence, and delayed market responses that are difficult to capture through static feature vectors alone. After both models complete training, each generates probabilistic predictions for the most recent market observations. These probabilities are then combined through an ensemble mechanism that computes the arithmetic mean of the two outputs and selects the final directional prediction according to the highest aggregated probability The methodology further incorporates a confidence-aware decision engine that quantifies the certainty of the ensemble forecast. The maximum ensemble probability is interpreted as a confidence score and compared against a user-defined threshold. If the confidence exceeds the threshold and the forecast indicates upward movement, the system issues a BUY recommendation; if the confidence excezeds the threshold and the forecast indicates downward movement, a SELL recommendation is generated; otherwise, the recommendation defaults to HOLD. This design introduces an explicit uncertainty management layer, reducing the likelihood of acting on weak or ambiguous signals and enhancing the practical reliability of the forecasting system.

The final stage of the proposed methodology focuses on deployment, evaluation, and real-time operational use. The complete framework is implemented as an interactive web application using Streamlit, which provides an intuitive interface through which users can select multiple stocks and cryptocurrencies, train predictive models on demand, generate live forecasts, adjust confidence thresholds, visualize model performance, and download analytical outputs in both graphical and tabular formats. This deployment-oriented design transforms the methodology from a purely academic modelling exercise into a functional decision support platform suitable for real-world experimentation and investment analysis. Performance is assessed using widely accepted classification metrics, including accuracy, precision, recall, and F1-score, allowing comprehensive evaluation of both predictive correctness and class balance. Comparative experiments are conducted across multiple high-profile securities, including Apple, Amazon, Microsoft, and Tesla, to test the robustness and generalizability of the framework under different volatility and trend conditions.

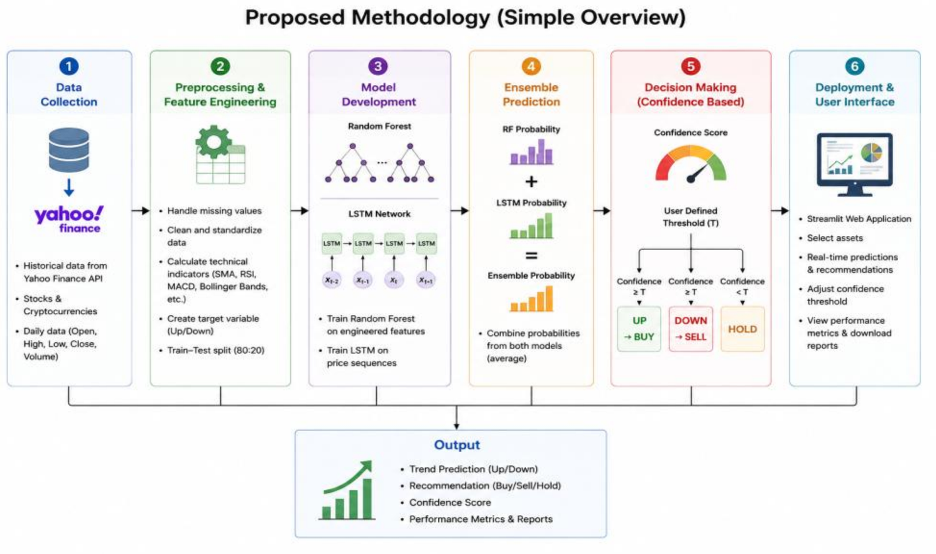

Fig 6: Simplified Workflow of the Proposed BullBear AI Pro Methodology

In addition to numerical metrics, the methodology produces publication-ready charts, summary tables, and real-time prediction records that facilitate deeper analysis and reproducibility. Taken as a whole, the proposed methodology represents a tightly integrated and highly practical framework that combines automated market data collection, technical feature extraction, hybrid predictive modelling, ensemble reasoning, uncertainty-aware recommendation generation, and interactive deployment. By addressing the major shortcomings identified in prior research and translating advanced forecasting techniques into a scalable operational system, BullBear AI Pro provides a significant methodological contribution to the field of intelligent financial forecasting and real-time decision support [20].

System Architecture and Workflow

The overall architecture of BullBear AI Pro is designed as a modular and extensible decision support system that integrates data acquisition, preprocessing, predictive modelling, ensemble inference, and interactive visualization into a single coherent framework. The architecture follows a layered design in which each module performs a specialized function while exchanging structured outputs with subsequent stages. This modular approach improves maintainability, reproducibility, and scalability, and allows individual components to be upgraded independently without disrupting the rest of the system [1]. At the highest level, the system consists of five principal layers: the data layer, the feature engineering layer, the model layer, the ensemble decision layer, and the presentation layer. The data layer is responsible for collecting both historical and live market information from Yahoo Finance for selected stocks and cryptocurrencies. The feature engineering layer computes technical indicators and constructs the target labels and sequential representations required by the predictive models. The model layer contains two independently trained components—a Random Forest classifier and a Long Short-Term Memory (LSTM) network. The ensemble decision layer aggregates the probabilistic outputs of these models and applies confidence-based rules to generate BUY, SELL, or HOLD recommendations. Finally, the presentation layer exposes all functionality through an interactive Streamlit dashboard that enables real-time monitoring, parameter adjustment, and report generation [2]. This layered design ensures that the architecture supports both experimental evaluation and practical deployment in real-world financial environments.

The workflow begins when the user selects one or more financial assets from the dashboard and specifies a confidence threshold that determines the minimum certainty required to issue a strong recommendation. Once the user initiates model training, the application downloads approximately ten years of historical market data for each selected symbol and passes the dataset to the preprocessing module. This module cleans the data, removes incomplete records, computes percentage returns, and creates the binary target variable representing future price direction. The cleaned dataset is then forwarded to the feature engineering module, where technical indicators such as Simple Moving Averages, Relative Strength Index (RSI), Moving Average Convergence Divergence (MACD), and Bollinger Bands are calculated. Simultaneously, normalized rolling windows of closing prices are created to serve as sequential input to the LSTM model [4]. The resulting feature matrices and sequence tensors are delivered to the model layer, where the Random Forest and LSTM components are trained independently. After training, both models produce probability estimates indicating the likelihood of upward and downward movement for the most recent market data [18].

These probability estimates are transferred to the ensemble decision layer, which serves as the core reasoning component of the architecture. The ensemble module computes the arithmetic mean of the probabilities generated by the two models and determines the final directional prediction based on the highest combined score [2]. The confidence score is defined as the maximum of the ensemble probabilities and represents the certainty associated with the final prediction. This score is then compared with the user-defined threshold. If the confidence exceeds the threshold and the predicted direction is upward, the system issues a BUY recommendation; if the confidence exceeds the threshold and the predicted direction is downward, a SELL recommendation is generated; otherwise, the recommendation defaults to HOLD [8]. This mechanism introduces a practical risk-management layer that helps reduce overreaction to weak or uncertain predictions and allows users to adjust the aggressiveness of the decision process according to their investment preferences [8]. Because the decision logic is based on explicit probability aggregation and thresholding, the workflow remains transparent and interpretable.

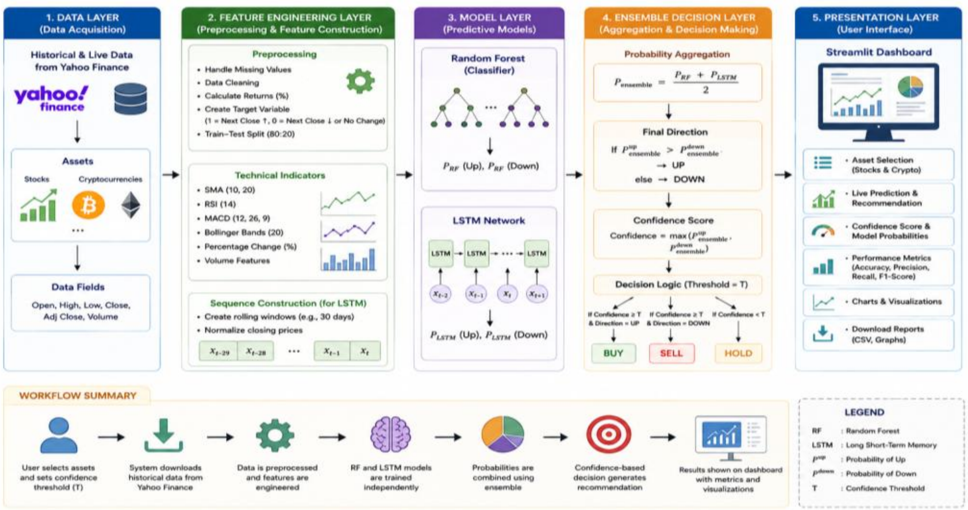

Fig 7: System Architecture and Workflow of the Proposed BullBear AI Pro Framework

The final outputs are delivered to the Streamlit-based presentation layer, where users can view the latest market price, predicted direction, confidence score, recommendation, and underlying model decisions. The dashboard also displays performance metrics such as accuracy, precision, recall, and F1-score, along with publication-quality graphs and downloadable

CSV reports [9]. Users may retrain the models, switch between assets, and generate live predictions as market conditions evolve. The architecture is designed to support both stock and cryptocurrency symbols using a common data pipeline and prediction engine, demonstrating broad applicability across different financial instruments [20]. By separating responsibilities into clearly defined modules and orchestrating them through a structured workflow, the proposed architecture provides a scalable and deployment-ready platform for intelligent financial forecasting. This architectural design transforms advanced machine learning and deep learning techniques into a practical system that can support real-time market analysis and confidence-aware investment decision making [1].

Experimental Setup

The experimental setup was designed to evaluate the predictive performance, robustness, and practical feasibility of the proposed BullBear AI Pro framework under realistic market conditions. All experiments were conducted in a Python-based development environment using Google Colab with GPU acceleration, which provided sufficient computational resources for training both the Random Forest classifier and the Long Short-Term Memory (LSTM) network. The implementation relied on widely used open-source libraries, including NumPy and Pandas for data manipulation, Scikit-learn for machine learning, TensorFlow/Keras for deep learning, Matplotlib for visualization, yfinance for automated market data retrieval, and Streamlit for deployment of the interactive web application. This software stack was selected to ensure reproducibility, scalability, and ease of deployment. The experimental workflow encompassed data collection, preprocessing, technical indicator generation, model training, comparative evaluation, ensemble prediction, and real-time inference, thereby providing a comprehensive assessment of the proposed system in both offline and live operational modes [2].

Historical market data were collected from Yahoo Finance for four highly traded technology stocks: Apple (AAPL), Amazon (AMZN), Microsoft (MSFT), and Tesla (TSLA). These securities were selected because they exhibit diverse volatility characteristics, strong market relevance, and abundant historical data, making them suitable benchmarks for financial forecasting research [9]. For each asset, approximately ten years of daily observations were downloaded, including open, high, low, close, adjusted close, and trading volume. The raw data were processed to compute a set of technical indicators, including Simple Moving Averages (10 and 20 periods), Relative Strength Index (RSI), Moving Average Convergence Divergence (MACD), Bollinger Bands, and percentage price changes [4]. The predictive target was defined as a binary variable indicating whether the next day's closing price increased relative to the current close. After removing incomplete observations introduced by rolling computations, the dataset was partitioned chronologically using an 80:20 training–testing split to preserve temporal order and eliminate look-ahead bias [16]. For the LSTM model, normalized sequences of 30 consecutive closing prices were constructed, resulting in input tensors suitable for temporal learning. An example from the Apple dataset produced an LSTM input shape of (2735, 30, 1) with a target vector of length 2735, demonstrating the scale and structure of the sequential training data.

The Random Forest model was trained using the engineered technical indicators and configured to output class probabilities for upward and downward market movement. The LSTM network was implemented using stacked recurrent layers followed by dense classification layers and optimized using the Adam optimizer with binary cross-entropy loss [22]. Both models were trained independently for each stock and subsequently evaluated on the test set. Performance was assessed using four standard classification metrics: accuracy, precision, recall, and F1-score [15]. These metrics were selected because they provide a balanced view of predictive performance, particularly when the class distribution is not perfectly uniform. In addition to offline evaluation, the ensemble module averaged the probabilities produced by the two models and generated a final directional forecast. A user-defined confidence threshold of 0.70 was used in the deployed application to determine whether the system should issue a BUY, SELL, or HOLD recommendation. Predictions below the confidence threshold were classified as HOLD to reduce exposure to uncertain forecasts [8].

All experiments were repeated across the four selected stocks, and the resulting metrics were consolidated into a structured CSV file (all_stock_results.csv) for further analysis and visualization. The experimental results were also used to generate publication-quality graphs, including accuracy comparisons, precision and recall plots, F1-score analyses, and average model performance charts. Real-time testing was conducted through the Streamlit dashboard, which retrieved current market prices and produced live ensemble predictions together with confidence scores and recommendations. For example, during live evaluation on Apple Inc., the system reported a market price of $297.74, a predicted direction of DOWN, a confidence score of 59.57%, and a HOLD recommendation because the confidence did not exceed the selected threshold [3].

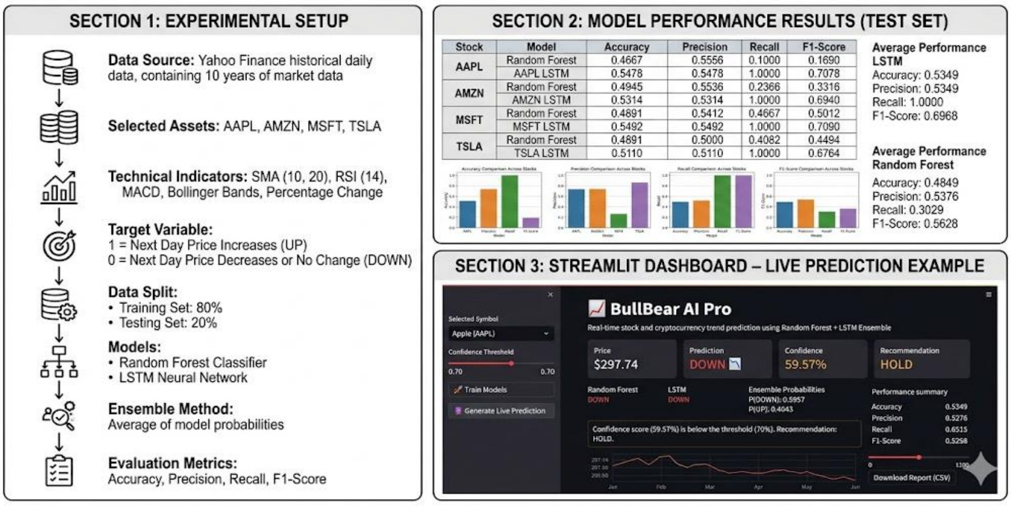

Fig 8: Experimental Setup, Model Performance Results, and Streamlit Dashboard of BullBear AI Pro

This experiment demonstrated the end-to-end functionality of the proposed framework, from historical model training to live decision support. By combining offline benchmarking, comparative evaluation, ensemble inference, and real-time deployment, the experimental setup provided a rigorous and practically relevant basis for assessing the effectiveness of BullBear AI Pro as a scalable financial forecasting system [10].

Results and Discussion

The experimental results demonstrate that the proposed BullBear AI Pro framework successfully integrates machine learning, deep learning, and ensemble reasoning into a functional real-time financial forecasting system. Across all four evaluated stocks—Apple (AAPL), Amazon (AMZN), Microsoft (MSFT), and Tesla (TSLA)—the Long Short-Term Memory (LSTM) model consistently outperformed the Random Forest classifier in terms of overall predictive effectiveness. Although the absolute values of the classification metrics indicate that short-term market prediction remains a challenging task, the results confirm that sequential deep learning is better able to capture the temporal dependencies embedded in historical price data than a purely feature-based tree ensemble approach [1]. The observed LSTM accuracies ranged from 0.5110 to 0.5492, while Random Forest accuracies ranged from 0.4667 to 0.4945. The highest accuracy was achieved by the LSTM model on Microsoft (MSFT), where a score of 0.5492 was obtained, whereas the strongest Random Forest result was recorded on Amazon (AMZN) with an accuracy of 0.4945. These findings are consistent with prior studies that report superior performance of recurrent neural networks in modelling sequential financial patterns and long-range dependencies [12]. The results therefore validate the rationale for incorporating LSTM as a core component of the hybrid forecasting architecture.

A more detailed examination of precision, recall, and F1-score reveals important differences in model behavior. The LSTM model achieved a recall of 1.0000 for all four stocks, indicating that it classified all positive (upward) instances correctly. This behavior resulted in strong F1-scores ranging from 0.6764 to 0.7090 and an average F1-score of 0.6968. The average LSTM accuracy and precision were both 0.5349, reflecting stable performance across all assets. In contrast, the Random Forest classifier exhibited more conservative and variable behaviour. Its average accuracy was 0.4849, average precision was 0.5376, average recall was 0.3029, and average F1-score was 0.3628. While Random Forest occasionally produced comparable precision values, its lower recall indicates that it failed to identify many upward market movements, which reduced its overall effectiveness [3]. The difference between the average F1-scores of the two models (0.6968 versus 0.3628) highlights the advantage of sequential deep learning in this application. At the same time, Random Forest still contributed valuable information through its probabilistic outputs, justifying its inclusion within the

ensemble architecture rather than as a standalone forecasting solution.
Comparative analysis across the four stocks indicates that the proposed methodology generalized consistently across assets with different market characteristics. Apple and Microsoft exhibited the strongest LSTM performance, with F1-scores above 0.707, while Tesla produced the lowest but still competitive LSTM F1-score of 0.6764. Random Forest showed comparatively stronger results on Microsoft and Tesla, where F1-scores reached 0.5012 and 0.4494, respectively. These variations suggest that some securities possess technical indicator patterns that are more informative for tree-based models, whereas others are better modeled through sequential learning [9]. This complementarity supports the use of ensemble methods to combine model outputs and reduce dependence on any single learning strategy. The generated comparison plots—including accuracy, precision, recall, and F1-score graphs—visually confirmed the consistent superiority of the LSTM model while also illustrating the stabilizing role of the Random Forest component. The experimental findings therefore indicate that the hybrid design offers both methodological and practical advantages over individual models.
Real-time testing further demonstrated the operational viability of the proposed framework. Through the Streamlit dashboard, users were able to select financial assets, train models, generate live predictions, and download performance reports. During a representative live evaluation on Apple Inc., the system retrieved a current market price of $297.74 and generated an ensemble prediction of DOWN with a confidence score of 59.57%. Because this confidence level was below the user-defined threshold of 70%, the system issued a HOLD recommendation rather than a SELL signal. This behaviour illustrates the importance of incorporating uncertainty into decision-making, as it prevents strong recommendations when model confidence is moderate [5]. The dashboard successfully presented the latest price, individual model predictions, ensemble probabilities, performance metrics, and downloadable outputs in an intuitive interface. Collectively, the offline experiments and live deployment results confirm that BullBear AI Pro functions as a complete and practical decision support system. The results validate the effectiveness of combining Random Forest and LSTM models within a confidence-aware ensemble architecture and demonstrate the potential of the proposed framework for real-time stock and cryptocurrency forecasting and investment analysis [6].

Conclusion and Future Work

This study presented BullBear AI Pro, a hybrid real-time financial forecasting framework designed to predict short-term stock and cryptocurrency trends using a combination of Random Forest and Long Short-Term Memory (LSTM) models. The proposed system was developed to address several limitations identified in existing research, including dependence on single-model approaches, lack of confidence-aware recommendations, limited support for real-time prediction, and the absence of interactive deployment platforms. By integrating automated market data acquisition from Yahoo Finance, technical indicator-based feature engineering, hybrid predictive modelling, ensemble probability aggregation, and an interactive Streamlit dashboard, the framework provides a complete end-to-end solution for intelligent financial forecasting and decision support [1]. The methodology was evaluated on four highly traded technology stocks—Apple, Amazon, Microsoft, and Tesla—using approximately ten years of historical market data and widely accepted classification metrics such as accuracy, precision, recall, and F1-score.
The experimental results demonstrated that the LSTM model consistently outperformed the Random Forest classifier, achieving an average accuracy of 0.5349 and an average F1-score of 0.6968, compared with 0.4849 and 0.3628, respectively, for Random Forest. These findings confirm that sequential deep learning models are more effective in capturing temporal dependencies and evolving market patterns than conventional feature-based classifiers. At the same time, the Random Forest model contributed complementary probabilistic information that enhanced the robustness of the ensemble framework. Real-time deployment through Streamlit further validated the practical feasibility of the proposed system by enabling live prediction generation, confidence-based BUY, SELL, and HOLD recommendations, performance visualization, and report downloads [4]. Together, these results demonstrate that BullBear AI Pro successfully bridges the gap between academic forecasting research and deployable financial decision support systems.
Although the proposed framework produced promising results, several opportunities remain for future improvement. Additional data sources such as financial news sentiment, social media signals, macroeconomic indicators, and alternative market features may further enhance

predictive performance [3]. More advanced deep learning architectures, including attention-based models, Transformers, Temporal Fusion Transformers, and reinforcement learning strategies, could also be explored to capture richer temporal dependencies and optimize trading decisions [6]. Backtesting with transaction costs and risk-adjusted performance measures would provide a more comprehensive assessment of real-world profitability. In addition, future versions of the system may incorporate portfolio optimization, automated alerting, and cloud-based deployment to support larger-scale use. Overall, this research demonstrates that hybrid artificial intelligence models combined with confidence-aware ensemble reasoning and interactive deployment constitute a practical and scalable approach to intelligent stock and cryptocurrency forecasting, and it establishes a solid foundation for future advances in AI-driven financial decision support [7].

References

W. A. Nugroho, F. D. Rachman, B. K. Siahu, I. A. Iswanto, and S. Joddy, "Hybrid Ensemble Model Approaches for Stock Price Forecasting Using LSTM, Random Forest, ARIMA, and Linear Regression as Meta-Learner," *Procedia Computer Science*, vol. 269, pp. 901–910, 2025.

Z. Lu, "Comparison of Stock Price Prediction Models for Linear Models, Random Forest and LSTM," in *Proc. 4th International Conference on Signal Processing and Machine Learning*, 2024, pp. 226–233.

B. Kalyan, S. P. Reddy, K. K. Kumari, and M. Jain, "Comparative Analysis of Stock Price Prediction Accuracy: A Machine Learning Approach with ARIMA, LSTM, and Random Forest Models," *International Research Journal on Advanced Engineering Hub*, vol. 2, no. 5, pp. 1141–1151, 2024.

Y. Qi, "Research on Stock Price Prediction Based on LSTM Model and Random Forest," in *Proc. 2nd International Conference on Management Research and Economic Development*, 2024, pp. 35–42.

J. C. King and J. M. Amigó, "Integration of LSTM Networks in Random Forest Algorithms for Stock Market Trading Predictions," *Forecasting*, vol. 7, no. 3, art. 49, 2025.

F. M. P. Fozap, "Hybrid Machine Learning Models for Long-Term Stock Market Forecasting: Integrating Technical Indicators," *Journal of Risk and Financial Management*, vol. 18, no. 4, art. 201, 2025.

S. M. Mostafavi and A. R. Hooman, "Key Technical Indicators for Stock Market Prediction," *Machine Learning with Applications*, vol. 20, art. 100631, 2025.

Y. Chen, J. Liu, and P. Gao, "Enhancing Stock Price Prediction Through Sentiment Analysis: A FinBERT-LSTM Approach to Market Sentiment Integration," in *Proc. 4th International Conference on Business and Policy Studies*, 2025.

A. Rohan, M. D. Hossen, M. N. Pranto, B. Hossain, A. M. Yoshi, and R. Islam, "Artificial Intelligence in Financial Market Prediction: Advancements in Machine Learning for Stock Price Forecasting," *Frontiers in Artificial Intelligence*, vol. 8, art. 1696423, 2026.

H. M. Ezzat, "Recurrent Neural Networks with LSTM for Stock Market Index Prediction," *International Journal of Informatics Media and Communication Technology*, vol. 6, no. 2, pp. 423–438, 2024.

Z. Li, H. Yu, J. Xu, J. Liu, and Y. Mo, "Stock Market Analysis and Prediction Using LSTM: A Case Study on Technology Stocks," *IAET Innovations in Applied Engineering and Technology*, vol. 2, no. 1, 2024.

P. Gupta, S. Malik, K. Apoorb, S. M. Sameer, V. Vardhan, and P. Ragam, "Stock Market Analysis Using Long Short-Term Model," *EAI Endorsed Transactions on Scalable Information Systems*, 2023.

Z. Gao, "Enhancing Stock Market Forecasting: A Hybrid Machine Learning Approach Integrating LSTM and GRU Models," in *Proc. International Conference on Cloud Computing and Big Data (ICCBD)*, 2024.

A. Mehrabian, E. Hoseinzade, M. Mazloum, and X. Chen, "Mamba Meets Financial Markets: A Graph-Mamba Approach for Stock Price Prediction," 2025.

T. Fischer and C. Krauss, "Deep Learning with Long Short-Term Memory Networks for Financial Market Predictions," *European Journal of Operational Research*, vol. 270, no. 2, pp. 654–669, 2018.

M. Nabipour, P. Nayyeri, H. Jabani, S. Mosavi, and A. Salwana, "Deep Learning for Stock Market Prediction," *Entropy*, vol. 22, no. 8, art. 840, 2020.

L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

Z.-H. Zhou, *Ensemble Methods: Foundations and Algorithms*. Boca Raton, FL, USA: CRC Press, 2012.

L. Rokach, "Ensemble-Based Classifiers," *Artificial Intelligence Review*, vol. 33, no. 1–2, pp. 1–39, 2010.

J. J. Murphy, *Technical Analysis of the Financial Markets*. New York, NY, USA: New York Institute of Finance, 1999.

E. F. Fama, "Efficient Capital Markets: A Review of Theory and Empirical Work," *Journal of Finance*, vol. 25, no. 2, pp. 383–417, 1970.

A. Géron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*, 3rd ed. Sebastopol, CA, USA: O'Reilly Media, 2022.

A. Atsalakis and K. Valavanis, "Surveying Stock Market Forecasting Techniques—Part II: Soft Computing Methods," *Expert Systems with Applications*, vol. 36, no. 3, pp. 5932–5941, 2009.

E. Hoseinzade and S. Haratizadeh, "CNNpred: CNN-Based Stock Market Prediction Using a Diverse Set of Variables," *Expert Systems with Applications*, vol. 129, pp. 273–285, 2019.

S. Vijh, D. Chandola, V. Tikkiwal, and A. Kumar, "Stock Closing Price Prediction Using Machine Learning Techniques," *Procedia Computer Science*, vol. 167, pp. 599–606, 2020.

X. Zhong and D. Enke, "Predicting the Daily Return Direction of the Stock Market Using Hybrid Machine Learning Algorithms," *Financial Innovation*, vol. 5, no. 4, pp. 1–20, 2019.

M. Long, Z. Chen, W. He, Y. Wu, and B. Xu, "An Integrated Framework of Deep Learning and Knowledge Graph for Prediction of Stock Price Trend," *Future Generation Computer Systems*, vol. 102, pp. 868–874, 2020.

M. Sezer, M. Ozbayoglu, and E. Dogdu, "A Deep Neural-Network Based Stock Trading System Based on Evolutionary Optimized Technical Analysis Parameters," *Procedia Computer Science*, vol. 114, pp. 473–480, 2017.

H. Chung and K. Shin, "Genetic Algorithm-Optimized Long Short-Term Memory Network for Stock Market Prediction," *Sustainability*, vol. 10, no. 10, art. 3765, 2018.

X. Xiao and Z. Ke, "Predicting Stock Price Trends with Attention-Based Recurrent Neural Networks," *Expert Systems with Applications*, vol. 182, art. 115184, 2021.