Journal of Artificial Intelligence and Data Mining (JAIDM), Vol. 14, No. 1, 2026, 95-114.

Shahrood University of Technology

Journal of Artificial Intelligence and Data Mining (JAIDM)
Journal homepage: http://jad.shahroodut.ac.ir

Research paper

A Hybrid Approach to Stock Market Forecasting with LSTM, Modified Complex Variational Mode Decomposition, and Secretary Bird Optimization Algorithm

Homa Mehtarizadeh[1], Najme Mansouri[2*], Behnam Mohammad Hasani Zade[3] and Mohammad Mehdi Hosseini[4]

[1,4] *Department of Applied Mathematics, Shahid Bahonar University of Kerman.*
[2,3] *Department of Computer Science, Shahid Bahonar University of Kerman.*

Article Info

Article History:
Received 06 March 2025
Revised 15 August 2025
Accepted 23 September 2025

DOI:10.22044/jadm.2025.15878.2703

Keywords:
Stock market, Secretary Bird Optimization Algorithm, Modified Complex Variational Mode Decomposition, Long Short-Term Memory, Prediction.

**Corresponding author: najme.mansouri@gmail.com (N. Mansouri).*

Abstract

Accurate and reliable stock price prediction is a formidable and essential task in financial markets, requiring the use of advanced techniques. This paper presents an innovative approach that integrates Long Short-Term Memory (LSTM) networks with Modified Complex Variational Mode Decomposition (MCVMD) for preprocessing and the Secretary Bird Optimization Algorithm (SBOA) for hyperparameter optimization. In the preprocessing phase, MCVMD decomposes stock price time series into intrinsic mode functions, effectively capturing complex patterns and reducing noise. To enhance predictive performance, SBOA optimizes both the hyperparameters of the LSTM network and the decomposition parameters of MCVMD. The proposed methodology is evaluated on datasets from six companies: Ferrari NV (RACE) and Intesa Sanpaolo (ISP) from Italy, Amadeus IT (AMA) and Repsol (REP) from Spain, and Hitachi Ltd (6501) and Chugai Pharmaceutical Co., Ltd. (4519) from Japan. Results show that the LSTM-MCVMD-SBOA model achieves lower error values compared with conventional benchmarks, including ARIMA-GARCH, vanilla LSTM, Long Short-Term Memory-Particle Swarm Optimization (LSTM-PSO), and Long Short-Term Memory-Sine Cosine Algorithm (LSTM-SCA). SBOA was selected because of its superior balance between exploration and exploitation, inspired by secretary bird hunting and evasion behavior, which enables efficient search in complex optimization landscapes. Overall, the proposed model demonstrates significantly improved predictive accuracy over conventional methods, highlighting the efficacy of combining advanced decomposition with nature-inspired optimization for stock market forecasting.

1. Introduction

In the world of financial markets, stock price prediction is a fundamental yet highly complex task, attracting considerable interest from both researchers and practitioners. Accurate predictions, particularly in volatile markets, facilitate informed decision-making, enhance risk management, and support strategic planning. Time series models and linear regressions often fail to capture the inherent nonlinearities and dynamic nature of financial time series data, resulting in suboptimal performance. Thus, advanced techniques have been explored in an effort to address these challenges more effectively.

This includes LSTM networks, which are specialized forms of Recurrent Neural Networks (RNN). The use of LSTM networks for forecasting time series is demonstrating remarkable success across various applications, including stock price prediction. Financial data is particularly complex, making LSTMs ideal for capturing long-term dependencies and temporal patterns [1, 2]. Although LSTMs have many advantages, selecting hyperparameters for optimal performance can be challenging and time-consuming. Automating and enhancing the hyperparameter tuning process requires robust optimization algorithms.

Recent studies also emphasize the role of metaheuristic optimization in improving predictive models. For instance, Ebrahimi and Hemmati [20] applied a multi-objective gravitational search algorithm to optimize circuit design parameters, underscoring the efficiency of population-based optimization methods in balancing exploration and exploitation. Inspired by this perspective, our study leverages the Secretary Bird Optimization Algorithm (SBOA) to fine-tune both LSTM and MCVMD parameters, thereby enhancing forecasting performance.

Yujun et al. [19] proposed a hybrid prediction framework combining LSTM networks with ensemble Empirical Mode Decomposition (EMD), demonstrating the benefits of decomposition in capturing intrinsic patterns before deep learning forecasting. Similarly, Guo et al. [21] developed an adaptive Support Vector Regression (SVR) model tailored for high-frequency stock price forecasting, highlighting the importance of adaptability in dynamic financial markets. These works further confirm the relevance of decomposition-based preprocessing and adaptive modeling, which our LSTM–MCVMD–SBOA model extends with enhanced optimization and noise reduction.

In this study, the SBOA is employed, a novel metaheuristic based on predatory and evasive behaviors of secretary birds. SBOA is able to balance exploration and exploitation effectively, making it a promising algorithm for solving complex optimization problems [3]. Utilizing SBOA, we aim to optimize the hyperparameters of the LSTM network to improve prediction. Prior evaluations by Fu et al. [3] demonstrated that SBOA outperformed established metaheuristics such as PSO and GA on 23 benchmark functions, achieving up to 15% higher convergence accuracy and faster optimization. These results indicate that SBOA is a promising algorithm for solving complex optimization problems such as hyperparameter tuning in LSTMs.

This study optimizes LSTM hyperparameters as well as introduces a preprocessing phase using Modified Complex Variational Mode Decomposition (MCVMD). In MCVMD, the stock price time series is decomposed into Intrinsic Mode Functions (IMFs), each representing a different frequency component. It captures intricate patterns more effectively and reduces noise, improving LSTM's predictive accuracy. Compared to earlier decomposition techniques, Miao et al. [4] showed that MCVMD improved decomposition quality and reduced noise by approximately 8–12%, leading to more accurate signal reconstruction. These improvements highlight MCVMD’s ability to capture intricate patterns in financial data more effectively, thereby enhancing LSTM’s predictive performance. Their experiments on nonstationary signals showed that MCVMD achieves lower reconstruction error and better preservation of low-frequency trends compared with Complex Variational Mode Decomposition (CVMD), confirming its ability to capture intricate patterns in complex time series. In our framework, these strengths are leveraged to enhance predictive performance when combined with LSTM.

The proposed methodology is evaluated using stock price data from seven diverse companies: Ferrari NV (RACE) and Intesa Sanpaolo (ISP) from Italy, Amadeus IT (AMA) and Repsol (REP) from Spain, Hitachi Ltd (6501) and Chugai Pharmaceutical Co., Ltd (4519) from Japan. Companies from these sectors and regions were selected for comprehensive evaluation. By applying the proposed methodology to these datasets, we aim to demonstrate the robustness and effectiveness of our approach in improving stock price prediction accuracy.

The proposed methodology must be implemented in several essential steps:

1. Data Preprocessing: During this step, the stock price data are normalized so that all features contribute equally, enhancing the model's overall performance.
2. Model Initialization: This step entails configuring the architecture of the LSTM network and setting up the parameters for the SBOA to optimize the LSTM and Modified -MCVMD method.
3. Training: To improve the model's predictive accuracy, SBOA is used to iteratively fine-tune the hyperparameters of the LSTM network and the α and K parameters of the MCVMD method.
4. Evaluation: The effectiveness of the LSTM-SBOA-MCVMD model is assessed by comparing its performance with established models such as LSTM-SCA, Long Short-Term Memory - Autoregressive Integrated Moving Average - Generalized Auto Regressive Conditional Heteroskedasticity (LSTM-

ARIMA-GARCH), LSTM, and LSTM-PSO using consistent evaluation metrics.

This study examines the potential benefits of incorporating the Secretary Bird Optimization Algorithm into stock price prediction models while highlighting the dynamic nature of financial markets (the ever-changing landscape of financial markets) and the constant quest to enhance forecasting accuracy. This study examines the strengths and limitations of various approaches in order to demonstrate the evolution of hybrid models in coping with complex market conditions. This study offers a novel approach to improving prediction accuracy by using the LSTM-SBOA-MCVMD model.

While several metaheuristic algorithms have been widely applied to hyperparameter optimization in stock price prediction—including Particle Swarm Optimization (PSO), Genetic Algorithms (GA), Differential Evolution (DE), and the Sine Cosine Algorithm (SCA)—these methods often exhibit limitations such as premature convergence, poor scalability in high-dimensional search spaces, and sensitivity to initial conditions. The Secretary Bird Optimization Algorithm (SBOA) is particularly well-suited to this application for three main reasons. First, SBOA provides a dynamic balance between exploration and exploitation by mimicking secretary bird hunting and evasion strategies. This property is especially important for financial data, which are noisy and nonstationary, as it reduces the likelihood of the optimization process becoming trapped in local minima. Second, SBOA requires fewer control parameters than traditional optimizers, making it less sensitive to parameter initialization and more robust when tuning multiple interdependent variables such as the hidden units, learning rate of LSTM, and decomposition parameters of MCVMD. Third, to the best of our knowledge, SBOA has rarely been applied in the financial forecasting domain. Its integration with MCVMD and LSTM therefore contributes both methodological novelty and practical improvements in predictive accuracy, as demonstrated in our comparative experiments. For these reasons, SBOA was selected as the optimization component of our proposed framework over conventional alternatives such as PSO or SCA.

In summary, although optimizers such as PSO and SCA have been employed for LSTM tuning in prior studies, their performance is often hindered by premature convergence and reduced robustness under noisy market conditions. By contrast, SBOA's adaptive balance between exploration and exploitation, its robustness to parameter initialization, and its novelty in financial forecasting make it particularly appropriate for integration with MCVMD–LSTM. This forms a key element of the originality and necessity of the proposed approach. In addition to conventional error metrics (RMSE, MAE, Log-Cosh Loss), we further validate the robustness of our proposed approach using statistical significance testing. Specifically, the Diebold–Mariano test is applied to confirm that improvements over baseline models are not only numerical but also statistically meaningful.

2. Background

2.1. Long-Short Term Memory

According to [1], LSTM incorporates cell state alongside hidden state to address the vanishing gradient problem. There are three gates in the LSTM: input, output, and forget. The equations below describe LSTM architecture.

The forget gate, represented by f_t, determines whether to keep or forget information from the previous cell state, C_{t-1}, where C_{t-1} denotes the memory cell value carried over from the prior time step [5].

$$f_t = \sigma(W_f.[h_{t-1}, x_t] + b_f) \tag{1}$$

At time, t, the input gate comprises the input vector, x_t, and σ represents the logistic sigmoid function. As described in [5], i_t is responsible for determining what information needs to be updated in the C_t cell state.

$$i_t = \sigma(W_i.[h_{t-1}, x_t] + b_i) \tag{2}$$

Candidate state values $\tilde{C}_t$ are computed using the tanh activation function and can be added to the state through [5].

$$\tilde{C}_t = \tanh(W_c.[h_{t-1}, x_t] + b_c) \tag{3}$$

The cell state C_t allows for the long-term storage of information through updating the internal state [5].

$$C_t = f_t \bullet C_{t-1} + i_t \bullet \tilde{C}_t \tag{4}$$

The operator ● represents the Hadamard product element-by-element in the matrix.

The logistic sigmoid activation function is used by O_t to determine what information is used as an output [5].

$$O_t = \sigma(W_o.[h_{t-1}, x_t] + b_o) \tag{5}$$

and the output h_t is computed as [5]:

$$h_t = O_t \bullet \tanh(C_t) \tag{6}$$

where W_* and b_* represent weight matrices and bias vectors, respectively.
In [1], gates determine whether information is forgotten or remembered, so LSTMs can predict time series. Furthermore, it handles continuous values, noise, and distributed representations.

2.2. Modified Complex Variational Mode Decomposition

In 2014, Dragomiretskiy and Zosso presented Variational Mode Decomposition (VMD) as a method for time-frequency analysis [6]. The band-limited intrinsic mode functions (BLIMFs) can be generated by non-recursively decomposing a real-valued multicomponent signal into quasi-orthogonal sub-signals. For complex-valued data, an extension method called Complex Variational Mode Decomposition (CVMD) was proposed. In CVMD, real signals are calculated from the Fourier spectrum and analytical signals of complex-valued signals. Due to the mathematical theory behind VMD, complex-valued signals cannot be analyzed using the algorithm. The limitations of CVMD are in the decomposition of the signal, even though it extends VMD to complex-valued data. In CVMD, low-frequency trends cannot be fully reconstructed because they are artificially separated into positive and negative frequencies. The defect can be overcome by applying up sampling and frequency shifting to the VMD method [7].
Complex-valued data has distinct frequency components, whereas real-valued data has mirrored frequency components. With MCVMD, complex-valued data can be converted to real-valued data without losing any information. This method differs from Complex Empirical Mode Decomposition (CEMD) and CVMD. Analytic signals derived from real-valued signals retain all the information of their original signal, despite having only positive frequency spectra. Analytic signals with symmetrical positive and negative frequencies are considered real parts. Data with complex values is transformed into analytic signals by shifting the spectrum to the positive. Real-valued data are produced by analyzing the real part of this analytic signal. Real-world signals are almost always discrete, so the proposed method is designed for practical applications. The algorithm consists of the following steps:
The discrete complex-valued data, $x(n)$, where $0 < n < N-1$ and n are integers, is normalized to 1. Consider Nyquist's sampling frequency if the sampling rate meets it. Thus, the maximum signal frequency cannot exceed $\frac{1}{2}$ and the minimum signal frequency cannot be below $-\frac{1}{2}$.
To begin, input data should be up sampled with a reversible, nondistorting interpolation algorithm, such as discrete sine interpolation. However, sinc interpolation's boundary effects often prevent direct use. To minimize boundary effects, the signal should be mirrored twice. In this way, the original input data can be mirrored as [4]

$$x_e = [x_{e1}, x_{e2}, x_{e3}], \tag{7}$$

The first half of the original data is represented by x_{e1} in this case:

$$x_{e1}(n) = x(\frac{N}{2} - n - 1), 0 \le n \le \frac{N}{2} - 1 \tag{8}$$

where x_{e2} represents the initial sequence as [4]

$$x_{e2}(n) = x(n), 0 \le n \le N-1 \tag{9}$$

As shown in [4], x_{e3} represents the mirrored sequence of the latter half of the original data:

$$x_{e3}(n) = x(N - n - 1), 0 \le n \le \frac{N}{2} - 1 \tag{10}$$

The extended signal can be represented as $x_e(n)$, where $0 \le n \le 2N-1$. To extend the signal in [4], the discrete Fourier transform is used.

$$X_e(k) = \sum_{n=0}^{2N} x_e(n) e^{-j\frac{2\pi}{2N}kn}, 0 \le k \le 2N-1 \tag{11}$$

The spectral samples are represented by this. Next, $2N$ samples with zero values are added following the first N spectral samples. One-dimensional vectors of length $4N$ are generated using [4].

$$X_{ez} = [X_{e1}, Z, X_{e2}] \tag{12}$$

where X_{e1} denotes the sequence $X_e(k)$ $0 \le k \le N-1$, Z denotes $2N$ samples valued at zero, and X_{e2} denotes the sequence $X_e(k)$ $N \le k \le 2N-1$, which can be rewritten as $X_{ez}(k)$, $0 \le k \le 4N-1$. The corresponding time-domain data can be obtained by [4]

$$X_{ez}(n) = 2F^{-1}[X_{ez}(k)], 0 \le k \le 4N-1, 0 \le n \le 4N-1 \tag{13}$$

where $F^{-1}[.]$ stands for the inverse discrete Fourier transform. After removing the mirror, we get the actual up sampled result [4].

$$x_z(n) = x_{ez}(n+N), 0 \le n \le 2N-1 \tag{14}$$

Zero padding in the frequency domain increases the sampling rate in the time domain [8]. Zero padding doubles the data length, so $x_z(n)$ now has a sampling rate of 2. The frequencies represented in the Fourier spectrum now span from -1 to 1 and values from -1/2 to 1/2. Multiplying exponential factors $e^{j2\pi.\frac{n}{4}}$ then shifts $x_z(n)$ as follows [4]

$$x_{zs}(n) = x_z(n)e^{j2\pi.\frac{n}{4}}, 0 \le n \le 2N-1 \tag{15}$$

Given that the frequency of $x_z(n)$ ranges from -1/2 to 1/2, all components of $x_{zs}(n)$ fall within the range of 0 to 1 after frequency shifting, resulting in having the analysis focused solely on frequencies that are greater than or equal to zero. Consequently, $x_{zs}(n)$ becomes an analytic signal. Analytic signals have the property of containing all information in their real parts. The derivation of a real-valued signal is as follows [4]:

$$x_{zsr}(n) = R\{x_{zs}(n)\} \tag{16}$$

Due to the real value of $x_{zsr}(n)$, the band-limited intrinsic mode functions (BLIMFs) can be obtained using standard VMDs. The decomposition number is K. It is possible to express the mode decomposition procedure as [4]:

$$x_{zsr}(n) = \sum_{i=1}^{K} u_i(n) + r(n) \tag{17}$$

where $u_i(n)$ denotes the i-th BLIMF corresponding to $x_{zsr}(n)$, and $r(n)$ describes the residual. Based on the Hilbert transform after VMD, the complex and analytic i-th BLIMFs are generated [4].

$$y_i(n) = u_i(n) + j\hbar[u_i(n)] \tag{18}$$

This is not the final result of MCVMD since the data are up sampled and modulated. It is possible to obtain complex BLIMFs after demodulation and down sampling. Exponential factors $e^{-j2\pi.\frac{n}{4}}$ are multiplied by the i-th up sampled and modulated complex BLIMFs $y_i(n)$ [4].

$$z_i(n) = y_i(n)e^{-j2\pi.\frac{n}{4}} \tag{19}$$

where $z_i(n)$ represents the i-th up sampled complex BLIMFs, which have a double sample rate over the original data. By down sampling the origin complex data $x(n)$, one can obtain the final i-th complex BLIMF $q_i(n)$ [4].

$$q_i(n) = z_i(2n+1), 0 \le n \le N-1 \tag{20}$$

Among the parameters of MCVMD, the penalty factor (α) and the number of modes (K) play the most critical roles in determining decomposition quality. α regulates the trade-off between fidelity to the original signal and smoothness of the decomposed modes, while K determines the number of intrinsic mode functions extracted. Other parameters such as tolerance and maximum iterations primarily affect convergence speed and numerical stability, and are typically set to standard values. Therefore, our study focuses on optimizing α and K, which directly influence the decomposition outcomes and, consequently, the prediction performance.

Algorithm 1. MCVMD.

1. Input: Complex-valued signal x(t)
2. Initialize parameters: number_of_modes, penalty_factor, tolerance, max_iterations
3. Initialize modes: u_k(t) for k = 1 to number_of_modes
4. Initialize center frequencies: omega_k for k = 1 to number_of_modes
5. While not converged and iteration < max_iterations:
   a. For each mode k:
      i. Update u_k(t) using the current center frequencies and input signal
      ii. Apply frequency shifting: x_shifted(t) = x(t) * exp (j * omega_0 * t)
      iii. Extract real-valued data: real_x_shifted(t) = real(x_shifted(t))
      iv. Apply VMD to real_x_shifted(t) to update mode u_k(t)
      v. Reconstruct complex mode: u_k(t) = u_k(t) * exp (-j * omega_0 * t)
   b. Update center frequencies omega_k using the updated modes
   c. Check convergence based on tolerance level
6. Output: Decomposed modes u_k(t) and their corresponding center frequencies omega_k

3. Related Work

Mutinda and Langat [9] addressed the problem of predicting stock prices in the African market, an area that is often overlooked. The GARCH model is combined with historical prices and advanced AI models, such as LSTM, Gated Recurrent Unit (GRU), and transformers, to analyze Airtel stock data from Yahoo Finance. Metrics such as Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and R-squared are used to compare hybrid models to standalone AI models. As a result, hybrid models perform better than standalone models, especially the GARCH-LSTM model.

Yan and Yao [10] focused on improving stock index forecasting accuracy. Recent models incorporate machine learning techniques, such as Support Vector Machine (SVM) and RNN, as well as traditional methods, such as Autoregressive Moving Average (ARMA), Autoregressive Integrated Moving Average (ARIMA), and GARCH. However, they remain in need of improvement in terms of accuracy. By layering predictions, the Deep Learning Weighted Regression-Long Short-Term Memory (DLWR-LSTM) model is applied to three indices of the Shanghai Stock Exchange to predict trends. This model achieves a MAPE of about 1%, demonstrating high accuracy. The DLWR-LSTM model remains consistent regardless of time series variance across different samples.

Song and Choi [11] focused on stock indices, such as Deutsche Aktienindex (DAX), Dow Jones Industrial Average (DOW), and S&P500, to apply advanced deep learning to financial forecasting. Hybrid models based on Convolutional Neural Networks-Long Short-Term Memory (CNN-LSTM), GRU-CNN, and ensemble methods are introduced in this study, which incorporate a novel feature that averages high and low stock prices. One-time-step and multi-time-step close prices are predicted by the models.

Based on market volatility and uncertainty, Pin et al. [12] developed a new stock index forecasting model. Stock indices are divided into Intrinsic Mode Functions (IMFs) using Complete Ensemble Empirical Mode Decomposition with Adaptive Noise (CEEMDAN). To assess the stability of each IMF, the Augmented Dickey Fuller (ADF) method is used. Stable time series are modeled with ARMA, whereas unstable time series are modeled with LSTM. According to their reconstructed predictions, these models closely match actual stocks.

Htet et al. [13] tested the random walk and efficient market hypotheses by applying machine learning techniques to stock price prediction. To predict whether a stock will outperform its index by 2% over the next 10 days, feature vectors are constructed using historical data derived from previous relative returns, and methods such as Random Forests (RF), Support Vector Machines (SVM), and Long Short-Term Memory (LSTM) are employed. LSTMs outperform RFs and SVMs in this study, and data-driven Machine Learning (ML) methods outperform random choices. Random walk and efficient market hypotheses do not have a significant chance of holding in this situation.

According to Sidra et al. [14], machine learning models and deep learning models could be combined to predict stock prices. In this study, eight regression and four LSTM models were developed using NIFTY 50 index data. Grid search optimization was used to optimize the models using walk-forward validation. To predict the NIFTY 50 index's open values for the following week, the LSTM-based univariate model using one week of prior data was the most accurate.

Chandar [15] proposed a model for predicting next-day stock prices based on Long Short-Term Memory (LSTM). Three phases are involved in the model: calculating ten technical indicators from historical data, normalizing these feature vectors, and using the LSTM model to predict the next day's closing price. NASDAQ and NYSE stock markets are used to evaluate the model's efficacy. This model's high prediction accuracy is demonstrated by comparisons to earlier models.

Research on stock market forecasting has evolved from traditional statistical approaches to advanced hybrid deep learning frameworks. This progression reflects the increasing complexity of financial data and the persistent challenge of achieving reliable predictions.

Early approaches, such as ARMA, ARIMA, and GARCH, have been widely applied to capture temporal dependencies and volatility in stock data. These models provide interpretable results and are computationally efficient; however, they are limited in handling nonlinear and nonstationary dynamics. For example, GARCH-based models can capture volatility clustering but fail to adapt to sudden structural shifts, leading to reduced accuracy in highly volatile markets.

To overcome these limitations, machine learning methods, such as Support Vector Machines (SVM), Random Forests (RF), and ensemble regressors, have been introduced. These approaches can exploit nonlinear patterns better than purely statistical methods. For instance, studies applying RF and SVM demonstrate improved short-term prediction performance compared to random walk models. Nonetheless, their reliance on carefully engineered features and inability to capture long-term temporal dependencies restricts their robustness.

With the rise of deep learning, recurrent neural networks, particularly Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRU), have shown superior ability to model sequential dependencies in stock data. LSTM-based models have been applied to major stock indices and individual companies, consistently outperforming traditional methods. Hybrid frameworks combining LSTM with CNNs or GRUs have further enhanced performance by extracting richer feature representations. However, these models remain highly sensitive to hyperparameter selection. Most studies rely on grid search or conventional optimizers, which are computationally expensive and prone to suboptimal convergence.

Another stream of research integrates decomposition techniques with predictive models to handle noise and multi-frequency components in stock time series. Approaches, such as CEEMDAN combined with ARMA or LSTM, have demonstrated improved forecasting under volatile conditions. Similarly, hybrid CNN-LSTM models applied to decomposed indices, such as DAX or S&P500, show enhanced short- and long-term prediction accuracy. Despite these gains, existing decomposition methods, like CVMD or CEEMDAN, often suffer from information loss in low-frequency trends, high computational costs, and difficulties in reconstructing signals. Although significant progress has been made, several challenges persist:

1. Generalizability: Many studies are limited to single stock indices or regional markets, reducing cross-market applicability.
2. Decomposition Limitations: Existing methods fail to fully preserve low-frequency information or introduce redundancy, weakening prediction accuracy.
3. Optimization Issues: Hyperparameter tuning is often handled by simple or outdated algorithms (e.g., PSO, SCA), limiting robustness and convergence speed.
4. Noise and Nonstationarity: Traditional models and basic LSTM hybrids still struggle with nonstationary, noisy financial data.

To address these gaps, this study proposes a hybrid LSTM–MCVMD–SBOA framework. The novelty of our approach lies in three aspects:

- Enhanced Decomposition: MCVMD overcomes the limitations of CVMD and CEEMDAN by preserving low-frequency components while reducing noise, enabling more accurate signal representation.
- Advanced Optimization: The recently introduced Secretary Bird Optimization Algorithm (SBOA) is employed to simultaneously optimize LSTM hyperparameters and MCVMD parameters, outperforming conventional metaheuristics.
- Robust Validation: Unlike prior works constrained to single indices, we evaluate the model across multiple companies from Italy, Spain, and Japan, demonstrating its robustness and generalizability in diverse market conditions.

This combination of decomposition, deep learning, and novel optimization establishes a more accurate and reliable methodology for stock price forecasting, positioning our work as a meaningful advancement over existing literature.

While existing research has made significant strides in the field of stock price prediction, several limitations remain. Many studies are constrained by the use of single data sources, limiting their generalizability across different markets. High computational costs and complexity in model training and validation can be prohibitive. Furthermore, traditional models often struggle with capturing long-term trends and external economic factors, and their reliance on selected technical indicators can leave them vulnerable to market disruptions or news events. The LSTM-MCVMD-SBOA approach addresses these challenges by integrating advanced techniques for more robust predictions. MCVMD effectively decomposes time series data, reducing noise and capturing complex patterns, while the SBOA optimizes hyperparameters for enhanced model performance. This combination allows the LSTM network to achieve higher accuracy and reliability in diverse market conditions, making it a more versatile and comprehensive solution for stock price prediction.

Stock price prediction research has made significant strides, but several limitations remain. There are many studies that utilize a single data source, limiting their generalizability across different markets. Model training and validation can be prohibitively expensive and complex. Additionally, traditional models are often unable to capture long-term trends or external economic factors, and their reliance on selected technical indicators leaves them vulnerable to market disruptions. In order to address these challenges, the LSTM-MCVMD-SBOA approach integrates advanced techniques. In addition to decomposing time series data effectively, MCVMD reduces noise and captures complex patterns, while SBOA optimizes hyperparameters to maximize model performance. This combination makes the LSTM network more accurate and reliable in predicting stock prices.

4. LSTM-MCVMD-SBOA

4.1. Secretary Bird Optimization Algorithm

Secretary birds are striking African raptors with distinct appearances and behaviors. They inhabit grasslands, savannas, and riverine areas south of the Sahara Desert. In addition to tropical grasslands, secretary birds live in sparsely wooded areas, open grasslands with tall grasses, semi-desert regions, and wooded regions. Their backs and wings are grey-brown, their chests are white, and their bellies are black [16, 17].

4.1.1. Inspiration from Secretary Bird Behavior

The secretary bird is known for its distinctive hunting technique. Their long legs allow them to deliver powerful kicks to their prey, primarily snakes. Hunting with this method is highly effective in subduing and killing prey. In SBOA, the exploration phase mimics this hunting behavior. Simulating the bird's hunting steps allows the algorithm to find potential solutions in the problem space. Secretary birds are also adept at evading predators in addition to hunting. To escape threats, they often run rather than fly, utilizing their agility and long legs. The exploitation phase of the algorithm models this behavior of evasion, where it refines the solutions found during the exploration phase to reach the optimal solution. In order to improve the best solutions, it "evades" the less optimal ones.

4.1.2. Behavioral Characteristics in SBOA

In the same way that the secretary bird adapts its hunting and evasion strategies based on the environment, the algorithm adjusts its search patterns dynamically. As the secretary bird balances hunting and evasion, SBOA explores new solutions while leveraging existing solutions [16].

4.2. A mathematical model of secretary bird optimization

The SBOA is presented a mathematical model of secretary birds' behavior to avoid natural enemies and hunt snakes.

4.2.1. Initial preparation phase

Secretary Bird Optimization Algorithm (SBOA) considers secretary birds as members of its population. Decision variables are determined by the positions of secretary birds in the search space. SBOA represents candidates for solving problems using secretary bird positions. Secretary birds are initially located according to (21).

$$X_{i,j} = lb_j + r \times (ub_j + lb_j), i = 1,2,...,N, j = 1,2,...,Dim \tag{21}$$

The position of the i^{th} secretary bird is indicated by X_i; the lower and upper bounds are indicated by lb_j and ub_j, and the random number between 0 and 1 is indicated by r. Using a population-based approach ((22)), the secretary bird optimization algorithm (SBOA) starts with a population of candidate solutions. In the given problem, these potential solutions X are generated randomly within upper bounds (ub) and lower bounds (lb). Iteratively, the best solution identified so far is considered to be the optimal one.

$$X = \begin{bmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,Dim} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,Dim} \\ \vdots & \vdots & \ddots & \\ x_{N,1} & x_{N,2} & \cdots & x_{N,Dim} \end{bmatrix} \tag{22}$$

Where X said secretary bird group, X_i said the i^{th} secretary bird, $X_{i,j}$ said the i^{th} secretary bird j^{th} question the value of a variable, said N group members (the secretary) number, and Dim said problem of variable dimension.

The secretary bird represents a problem that has several possible solutions. In this way, each secretary bird provides values that can be used to evaluate the objective function. Thus, an objective function vector can be created using (23).

$$F = \begin{bmatrix} F_1 \\ \vdots \\ F_i \\ \vdots \\ F_N \end{bmatrix}_{N\times 1} = \begin{bmatrix} F(X_1) \\ \vdots \\ F(X_i) \\ \vdots \\ F(X_N) \end{bmatrix}_{N\times 1} \tag{23}$$

A vector of objective function values is F, while the value F_i is obtained by the i^{th} secretary bird. To identify the optimal solution for a given problem, it is necessary to compare the objective function values. Maximization problems are best solved by secretary birds with the highest objective function value. In addition to updating the secretary bird positions and objective function values, each iteration must also determine the best candidate solution. SBOA members were updated using two distinct secretary bird behaviors. These two types of behavior include: (a) Hunting strategies for secretary birds; (b) Escape strategies for secretary birds. Each iteration updates each secretary bird twice.

4.2.2. Hunting strategy of secretary bird

Snakes are typically hunted in three stages by secretary birds: searching for prey, consuming prey, and attacking prey. According to the biological statistics of the hunting phases and their durations for secretary birds, we segmented the hunting process into three equal stages, namely and corresponding to the three phases of the secretary bird's predation: searching for prey, consuming prey, and attacking prey $t < \frac{1}{3}T, \frac{1}{3}T < t < \frac{2}{3}T$ $\frac{2}{3}T < t < T$

In SBOA, each phase is modeled as follows:

Stage 1 (Searching for Prey): The first stage of the secretary bird's hunting process involves actively searching for prey, primarily snakes. The bird uses its sharp vision to scan the surroundings and spot any movement in the tall grass. Its long legs enable it to cover a large area quickly and also help it maintain a safe distance from the potential danger posed by snakes [18]. This cautious and

methodical search ensures that the bird can locate prey while minimizing the risk of being bitten. In the SBOA, this stage is represented by the exploration phase of the algorithm. The algorithm searches the solution space for potential solutions to the optimization problem. The equations used in this phase ensure that the search is conducted in a way that maximizes the likelihood of finding good solutions while avoiding local optima. By using (24) and (25), it is possible to mathematically model the updating of the secretary bird's position during the stage of Searching for Prey.
While

$$t < \frac{1}{3}T,\ x_{i,j}^{newP1} = x_{i,j} + (x_{random\_1} - x_{random\_2}) \times R_1 \tag{24}$$

$$X_i = \begin{cases} X_i^{new,P1}, if & F_i^{new,P1} < F_i \\ X_i, else & \end{cases} \tag{25}$$

R_1 is an array with dimensions of 1 by Dim, generated randomly within the range [0, 1]. Here, Dim represents the number of dimensions in the solution space. $x_{i,j}^{new,P1}$ denotes the value of R_1 in the j^{th} dimension, and $F_i^{new,P1}$ stands for the fitness value of the objective function. In essence, R_1 contains random values between 0 and 1 for each dimension, $x_{i,j}^{new,P1}$ indicates the specific value in one of these dimensions, and $F_i^{new,P1}$ evaluates the solution's fitness.

Stage 2 (Consuming Prey): Once the secretary bird has located its prey, the next stage is to consume it. This stage involves a careful approach to capturing and eating the snake. The bird uses its powerful beak and sharp talons to kill the prey quickly and efficiently [17]. This stage requires precision and focus, as any mistake could result in injury to the bird. In the SBOA, this stage corresponds to the exploitation phase of the algorithm. During this phase, the algorithm refines the solutions found during the exploration phase, focusing on improving the quality of the best solutions. The equations used in this phase ensure that the algorithm converges towards optimal solutions, maximizing the overall effectiveness of the optimization process. As the secretary bird entered the Consuming Prey stage, we can update its position using (27) and (28).

$$RB = randn(1, Dim) \tag{26}$$

While $\frac{1}{3}T < t < \frac{2}{3}T,$

$$x_{i,j}^{newP1} = x_{best} + \exp((t/T)^{\wedge}4) \times (RB - 0.5) \times (x_{best}, x_{i,j}) \tag{27}$$

$$X_i = \begin{cases} X_i^{new.P1}, if & F_i^{new,P1} < F_i \\ X_i, & else \end{cases} \tag{28}$$

In this case, $randn(1, Dim)$ represents an array of dimension $1 \times Dim$ of a normal distribution (mean 0, standard deviation 1), while x_{best} represents the current best value.

Stage 3 (Attacking Prey): Secretary birds use their powerful legs to attack snakes when they are exhausted. Secretary birds kick snakes with precision using their sharp talons. By incapacitating or killing the snake quickly, the kicks avoid being bitten. This causes the snake to die when its talons strike its vital points. Secretary birds are capable of carrying snakes in the sky and releasing them so they fall to their deaths. By introducing a Levy flight strategy during random search, the optimizer enhances its global search capabilities, reduces SBOA'S tendency to stick in local solutions, and improves convergence accuracy. Levy flights involve short, continuous steps and occasional long jumps. They simulate the secretary bird's fight ability to enhance its search space exploration. In addition to finding individuals quickly, the algorithm improves its accuracy by searching a larger area. The SBOA algorithm is made more dynamic, adaptive, and flexible by introducing a nonlinear perturbation factor known as $(1-\frac{t}{T})(2\times\frac{t}{T})$ that aims to balance exploration and exploitation, avoid premature convergence, accelerate convergence, and improve algorithm performance.
Thus, updating the secretary bird's position during Attacking Prey is mathematically modeled by (29) and (30).

While $t > \frac{2}{3}T,$

$$x_{i,j}^{new,P1} = x_{best} + ((1-\frac{t}{T})^{\wedge}(2\times\frac{t}{T})) \times x_{i,j} \times RL \tag{29}$$

$$X_i = \begin{cases} X_i^{new.P1}, if & F_i^{new,P1} < F_i \\ X_i, & else \end{cases} \tag{30}$$

Weighted Levy fights, also known as RL, are utilized to enhance the precision of the algorithm.

$$RL = 0.5 \times Levy(Dim) \tag{31}$$

The Levy flight distribution function is represented by Levy (Dim). As a result, it is calculated as follows:

$$Levy(D) = s \times \frac{u \times \sigma}{|v|^{\frac{1}{\eta}}} \quad (32)$$

Here, s = 0.01 and η = 1.5. u and v are random numbers between 0 and 1. There is a formula that can be used to calculate σ as follows:

$$\sigma = \left(\frac{\Gamma(1+\eta) \times \sin(\frac{\pi\eta}{2})}{\Gamma(\frac{1+\eta}{2}) \times \eta \times 2(\frac{\eta-1}{2})} \right)^{\frac{1}{\eta}} \quad (33)$$

Here, Γ denotes the gamma function and η has a value of 1.5.

4.2.3 Escape strategy of secretary bird

To protect themselves or their food, secretary birds typically use evasion strategies. Generally, there are two types of strategies. The first step is to fight or run as fast as possible. Due to their exceptionally long legs, they are capable of running at impressive speeds. Their nickname "marching eagles" comes from their ability to cover 20-30 kilometers in a day. Furthermore, secretary birds can escape danger quickly by flying to safer locations. A second strategy is camouflage. Secretary birds blend into their environment to avoid predators [18]. One of two conditions is assumed equally often by SBOA:

i. C1: Camouflage by environment;
ii. C2: Fly or run away.

Secretaries seek camouflage environments when they detect predators nearby. Without a suitable and safe camouflage environment, they will either fight or run away. The $(1-\frac{t}{T})^2$ factor is introduced as a dynamic perturbation factor. The dynamic perturbation factor in the algorithm allows it to maintain a balance between two important processes: exploration and exploitation. The exploration phase seeks new solutions, while the exploitation phase refines and improves the existing ones. By adjusting these factors at different stages of the algorithm, it can either increase exploration to discover new potential solutions or enhance exploitation to fine-tune and optimize the current solutions. This dynamic adjustment helps the algorithm adapt to different phases of the optimization process, ensuring a more effective and efficient search for the best solution. In (34), both evasion strategies are mathematically modeled, and this revised condition can be represented by (35).

$$x_{i,j}^{new,P2} = \begin{cases} C_1: x_{best} + (2 \times RB - 1) \times (1-\frac{t}{T})^2 \times x_{i,j}, if & rand < r_i \\ C_2: x_{i,j} + R_2 \times (x_{random} - K \times x_{i,j}), else & \end{cases} \quad (34)$$

$$X_i = \begin{cases} X_i^{new,P2}, if & F_i^{new,P2} < F_i \\ X_i, else & \end{cases} \quad (35)$$

Here, $r = 0.5$, R_2 represents the random generation of an array of dimension $1 \times Dim$ from the normal distribution. The current iteration's random candidate solution is x_{random}. In (34), K represents the random selection of the integer 1 or 2, which can be calculated by using (36).

$$K = round(1 + rand(1,1)) \quad (36)$$

In this case, rand (1, 1) means generating a random number between (0,1). Figure 1 presents the pseudocode for SBOA, while Figure 2 shows the flowchart.

4.3. LSTM-MCVMD-SBOA

The proposed methodology integrates Long Short-Term Memory (LSTM) networks with Modified Complex Variational Mode Decomposition (MCVMD) and Secretary Bird Optimization Algorithm (SBOA) for accurate stock price prediction.

Initially, historical stock price data from various companies are collected and preprocessed to remove missing values, outliers, and normalize the data. MCVMD is employed to decompose the stock price time series into intrinsic mode functions (IMFs) that capture complex patterns and reduce noise. In the MCVMD framework, several parameters govern the decomposition process, including the penalty factor (α), the number of modes (K), the tolerance threshold, and the maximum number of iterations. Among these, α and K are the most critical for ensuring high-quality decomposition.

The penalty factor α regulates the trade-off between data fidelity and the smoothness of the extracted modes: too small a value may retain excessive noise, whereas too large a value may over smooth the signal and suppress meaningful fluctuations.

Similarly, the number of modes K determines how many intrinsic mode functions are generated; an insufficient K can cause distinct oscillatory components to merge, while an excessively large K may introduce spurious or redundant modes. By contrast, parameters such as the tolerance threshold or maximum iterations primarily affect convergence speed and computational cost rather than the quality of the decomposed modes. Therefore, in this study, we employ SBOA to optimize α and K, as they have the most direct and significant impact on decomposition quality and, ultimately, on the predictive performance of the hybrid forecasting model. Two critical parameters of MCVMD are optimized using SBOA, which also optimizes the LSTM network's hyperparameters.

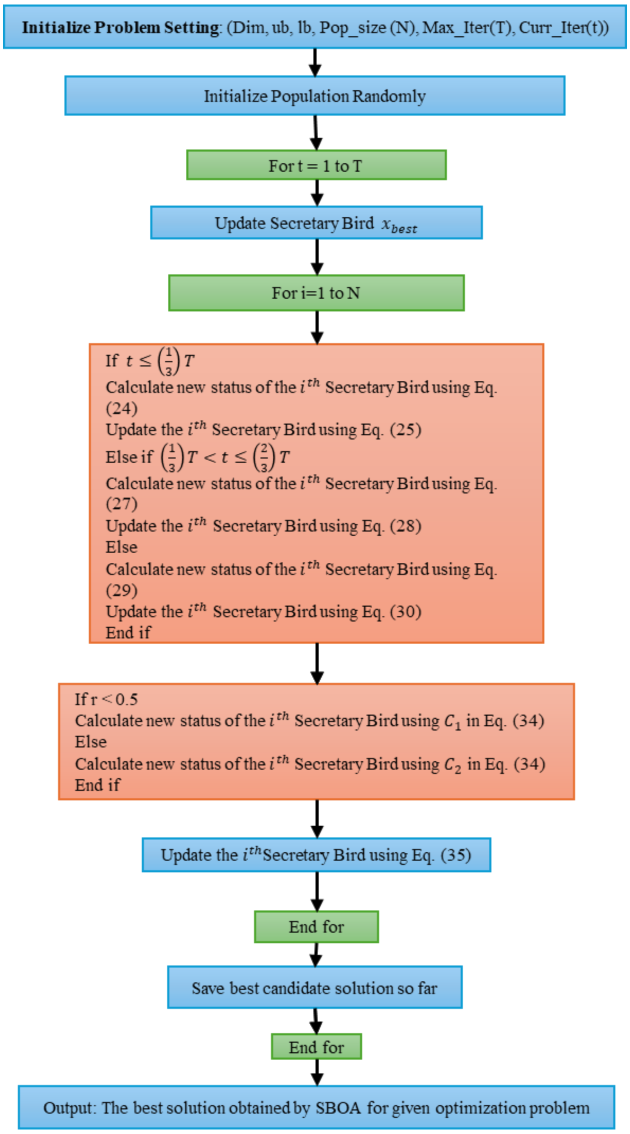


Figure 1. Flowchart of Pseudo code of SBOA.

4.3.1. Optimization Problem Formulation

The integration of the Secretary Bird Optimization Algorithm (SBOA) with the LSTM and MCVMD components can be formulated as a mathematical optimization problem. In this formulation, the decision variables are defined as:

$$X = \{numHiddenUnits, maxEpochs$$
$$, InitiallearningRate$$
$$, LearnRateDropFactor, K, \alpha\},$$

where $numHiddenUnits$ denotes the number of hidden units in the LSTM network, $maxEpochs$ is the maximum number of training iterations, $initialLearningRate$ is the learning rate at the beginning of training, $learnRateDropFactor$ controls the decay of the learning rate during training, and K and α represent the decomposition number and penalty factor of the MCVMD, respectively.

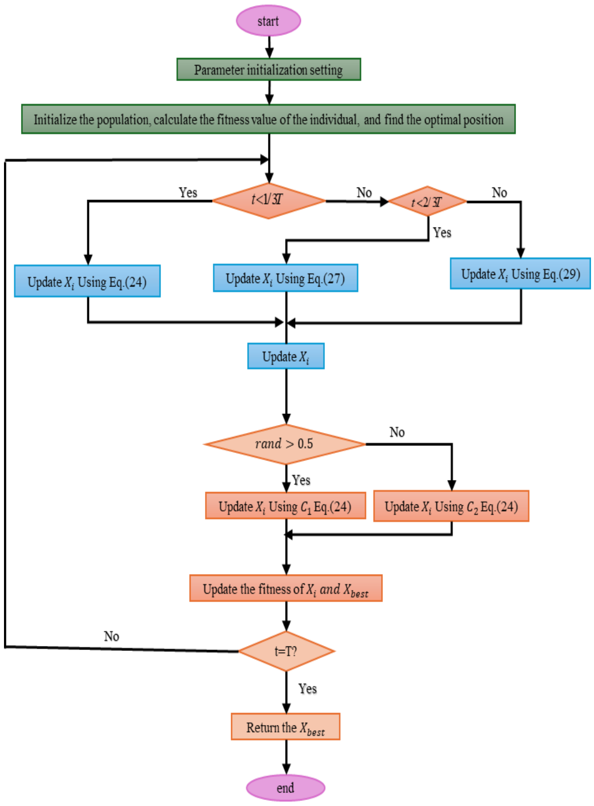


Figure 2. Flowchart of SBOA.

The objective function aims to minimize the forecasting error between the actual stock prices y_i and the predicted values $y_i(X)$. This study adopts the Root Mean Square Error (RMSE) as the primary performance criterion, expressed as:

$$\min_X RMSE(X) = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - y_i(X))^2} \tag{37}$$

where n is the number of data points. A lower RMSE indicates a more accurate prediction model. To define the feasible solution space, the following constraints are imposed on the decision variables

$$10 \leq numHiddenUnits \leq 100,$$
$$50 \leq \max Epochs \leq 500,$$
$$10^{-4} \leq initialLearningRate \leq 0.01,$$
$$0.1 \leq LearnRateDropFactor \leq 0.5,$$
$$2 \leq K \leq 6,$$
$$500 \leq \alpha \leq 2000,$$

Accordingly, the optimization problem is explicitly defined in terms of its decision variables, objective function, and constraints. SBOA is then applied to iteratively update the decision variables until the optimal configuration that minimizes the RMSE is obtained.

5. Analysis of experimental results and setup

In this study, we used the MATLAB 2021 software version 2H21 on a Windows 10 operating system equipped with 12GB of RAM. Prior to using the data, it was decomposed. Six active companies from Italy, Japan, and Spain were predicted by using LSTM recurrent neural networks. It is possible to increase the accuracy and speed of an LSTM neural network by optimizing its parameters. In addition, the MCVMD variables were optimized using an optimization algorithm. In MCVMD, values were computed based on the SBOA, taking the number of hidden units, the maximum number of epochs, and the initial learning rate into account. Based on metrics such as Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and Log-Cosh Loss, this model was evaluated against LSTM-SCA, LSTM, LSTM-PSO, and ARIMA-GARCH forecasting models. For comparing the predictive accuracy of the models, the Diebold-Mariano (DM) test was used. DM tests determine whether the forecasting errors of the two models differ statistically. The null hypothesis for this test is that both models have the same level of predictive accuracy. In other words, they perform equally well. The alternative hypothesis is that the predictive accuracy of the two models is significantly different, meaning one model outperforms the other. In the context of the Diebold-Mariano test, the p-value indicates the probability that the observed forecasting errors are due to random chance.

5.1. Original time series data

This study employs the closing prices of six prominent Italian, Spanish, and Japanese companies from December 28, 2018, to December 28, 2024. The analysis includes Ferrari NV and Intesa Sanpaolo from Italy, Repsol and Amadeus from Spain, and Hitachi and Chugai from Japan. These companies’ significant presence and influence in their respective markets, offering a diverse representation across different industries, led to their selection. The data set includes daily closing prices for each company.

5.2. Evaluation criteria

In addition to measuring the average magnitude of prediction errors, the root mean square error (RMSE) provides a good indicator of the model's overall performance. Predictive accuracy improves with a lower RMSE value.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2} \tag{38}$$

This method calculates the average absolute difference between actual and predicted values to determine prediction accuracy. The performance of models is better when mean absolute error (MAE) values are lower.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|\hat{y}_i - y_i| \tag{39}$$

The Log-Cosh Loss is defined as:

$$Log - \text{Cosh}\, Loss = \frac{1}{n}\sum_{i=1}^{n}\log(\cosh(\hat{y}_i - y_i)) \tag{40}$$

Where $\hat{y}_i$ is the predicted value, y_i is the actual value, and n is the number of data points. There is smooth and differentiable Log-Cosh Losses, which can contribute to the stabilization of training. It approximates absolute error and is differentiable everywhere, so it is less affected by outliers than mean squared error.

5.3. Performance evaluation

In Table 1, three error functions are used to compare the performance of predictive models for several companies. The best prediction accuracy is displayed by Amadeus, with the lowest root mean square error (RMSE), mean absolute error (MAE), and log-cosh loss values. There were some outliers in the data for Repsol, with slightly higher error values than Amadeus, and a relatively higher log-cosh loss than Amadeus. According to the log-cosh loss, Ferrari NV has a moderate impact from outliers, but has higher RMSE and MAE values than Amadeus and Repsol. There are even higher error values for Intesa Sanpaolo, which suggests less accurate predictions and significant outliers. Its high RMSE indicates substantial deviations from predictions, while its lower MAE indicates that it is skewed by a few large errors. In Chugai, the error values are relatively low, indicating accurate predictions with minor outlier effects. Hitachi Ltd and Intesa Sanpaolo's predictive models need improvement, while Amadeus leads in performance, followed by Chugai's good predictive accuracy.

Table 1. Performance of predictive models.

Error Function	Amadeus	Repsol	Ferrari NV	Intesa Sanpaolo	Hitachi Ltd	Chugai
RMSE	4.98×10^{-4}	0.0042	0.2437	0.3352	0.4679	0.1236
MAE	8.87×10^{-4}	**0.0016**	0.2118	0.3665	0.2117	0.0757
Log-Cosh Loss	**8.64×10^{-7}**	0.0212	**0.0262**	**0.0777**	**0.0484**	**0.0095**

In Table 2, Amadeus has a balanced setup with a moderate number of hidden units, and an initial learning rate that suggests careful adjustment for training, while a relatively large **α** indicates strong regularization. Repsol uses fewer hidden units and epochs than Amadeus, but a higher initial learning rate, indicating faster initial training, and the regularization parameter is also moderately high. Ferrari NV has the fewest hidden units, suggesting a simpler model, and it uses a small initial learning rate and a high learn rate drop factor, indicating careful, slow training with moderate regularization. Intesa Sanpaolo uses the most hidden units and epochs, suggesting a more complex model and longer training times, with a moderate initial learning rate and strong regularization. Hitachi Ltd has fewer epochs and hidden units, indicating a simpler model with a small initial learning rate and low learn rate drop factor, suggesting slow, careful training. Chugai uses a high initial learning rate and learns rate drop factor, indicating fast initial training that slows down significantly, and the model is relatively complex with a strong regularization parameter. This analysis highlights how each company has different strategies for training their models, with varying complexities, training durations, and regularization strengths.

In Table 2, Amadeus has a balanced setup, with a moderate number of hidden units and an initial learning rate that indicates careful training adjustment, while a relatively large α indicates strong regularization. Amadeus uses more hidden units and epochs, while Repsol uses fewer. Repsol has a higher learning rate, which indicates faster initial training, and its regularization parameter is also moderately high. In Ferrari NV, there are fewer hidden units, indicating a simpler model, and a low initial learning rate and a high learn rate drop factor, indicating a careful, slow training process with moderate regularization. There are the moderate learning rate and strong regularization among Intesa Sanpaolo, which suggests a more complex model and longer training time. Compared to other companies, Hitachi Ltd has fewer epochs and hidden units, suggesting a simpler, slower-learning model. In addition, the model is relatively complex with a strong regularization parameter, and Chugai uses a high initial learning rate and a low learning rate drop factor to indicate a fast initial learning rate that slows down significantly over time. According to this analysis, each company trains its models differently with varying complexities, training durations, and regularization strengths.

The parameters of the long short-term memory (LSTM) and multichannel variational mode decomposition (MCVMD) models were tuned automatically using the Secretary Bird Optimization Algorithm (SBOA). At the beginning of the optimization, parameter values were randomly initialized within predefined ranges: the number of hidden units [10,100], maximum epochs [50,500], initial learning rate [10^-4,0.01], learn rate drop factor [0.1,0.5], decomposition number K∈ [2,6], and penalty factor α∈ [500,2000]. Each candidate solution was evaluated by training the LSTM-MCVMD model and calculating the root mean square error (RMSE) on the validation dataset. SBOA iteratively updated the parameter values by simulating secretary bird hunting and evasion behaviors, balancing exploration and exploitation of the search space. The optimization was terminated when the maximum number of iterations was reached or when the change in RMSE fell below a predefined tolerance. The best-performing parameter set obtained by SBOA for each company is summarized in Table 2. These values were therefore not chosen arbitrarily, but rather resulted from the optimization procedure designed to minimize prediction error.

Table 2. Value of parameters.

	Num-HiddenUnits	**maxepochs**	**InitialLearnRate**	**LearnRateDropFactor**	**K**	**α**
Amadeus	46	357	0.0017	0.1745	5	2000
Repsol	36	283	0.0067	0.1935	5	1022
Ferrari NV	10	403	10^{-3}	0.2828	4	1000
Intesa Sanpaolo	50	464	0.0025	0.1246	2	1789
Hitachi Ltd	26	100	10^{-3}	0.1	2	1000
Chugai	49	131	0.01	0.4	4	1000

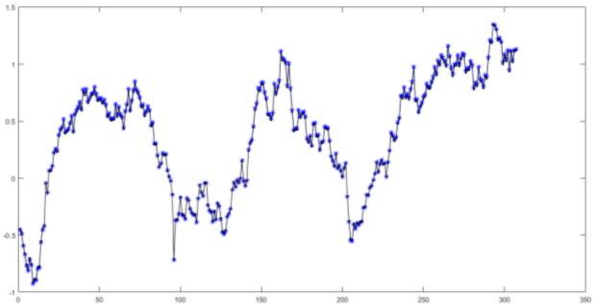
Figure 3 (a–f). Performance of predictive models. Actual stock prices are shown as black dots, and predictions of the LSTM-MCVMD-SBOA model are shown as blue stars. Both color and marker shape are used to distinguish curves, ensuring visibility in grayscale print.

Figure 3 (a) depicts the predicted stock prices of Amadeus using the LSTM-MCVMD-SBOA model (blue line) alongside the actual stock prices (black line). The model's predictions closely align with the actual values, successfully capturing the major peaks and troughs, such as the significant drop to -2.5 and subsequent rise to around 1. Although minor deviations exist where the predicted values do not perfectly match the actual prices, these differences are expected due to noise or sudden market events. This strong visual alignment corresponds to the lowest RMSE and MAE values reported in Table 1, highlighting the robustness of the model for this dataset. Overall, the LSTM-MCVMD-SBOA model demonstrates strong predictive performance, accurately following the general upward and downward trends in the stock price over time, indicating its effectiveness as a tool for stock market forecasting. Figure 3 (b) demonstrates the LSTM-MCVMD-SBOA model's effective prediction of Repsol stock prices. The predicted values (blue line) closely follow the actual values (black line), capturing major peaks and troughs, such as the initial rise to around 2 and the decline to -1.5. While minor deviations exist due to market events or noise, the model overall shows robust performance, making it a reliable tool for stock market forecasting. Figure 3 (c) illustrates the performance of our proposed methodology in predicting Ferrari NV (RACE) stock prices. The x-axis represents the time steps, while the y-axis represents the normalized stock price values. The black line depicts the actual stock prices over the observed period, and the blue line represents the predicted stock prices generated by our model.

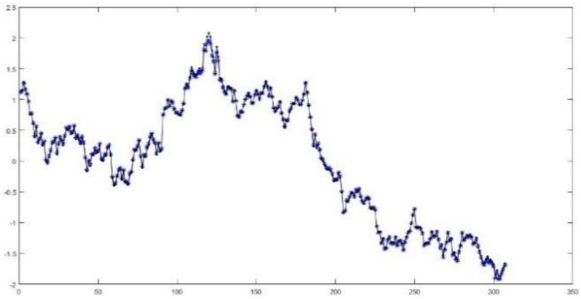
Based on the LSTM-MCVMD-SBOA model (blue line), Amadeus stock prices are predicted (blue line) along with the actual stock prices (black line). The model accurately predicted the actual values, capturing the major peaks and troughs, such as the drop to -2.5 and subsequent rise to 1. Minor deviations exist between predicted values and actual prices, but these differences are expected due to noise or sudden market events. Based on its predictive performance, the LSTM-MCVMD-SBOA model can accurately forecast stock prices and show an upward and downward trend over time, indicating its effectiveness as a stock market forecasting tool. According to Figure 3 (b), Repsol stock prices are effectively predicted by the LSTM-MCVMD-SBOA model. There are significant peaks and troughs in the predicted (blue line) values that closely match the actual values (black line), including the initial rise to around 2, followed by a decline to -1.5 in the final years. It is a reliable tool for stock market forecasting, even with minor deviations caused by market events or noise. Although some deviations appear during rapid shifts, they remain small compared to competing models (see Figure 4). This robustness is reflected in Repsol's low error values (RMSE = 0.0042, MAE = 0.0016), demonstrating the model's strength in capturing cyclical movements in energy sector data. As shown in Figure 3 (c), the proposed methodology performs well when it comes to predicting Ferrari NV's (RACE) stock price. The x-axis represents the time steps, while the y-axis represents the normalized stock price values. As shown in the graph below, the black line represents the actual stock price over the period observed, while the blue line represents the predicted price generated by our model over the same period. In Ferrari NV, the model captures medium-sized fluctuations and gradual recovery phases. While the alignment is generally strong, deviations around extreme peaks suggest that highly volatile luxury market dynamics are harder to model perfectly. This is consistent with Ferrari's higher RMSE (0.2437). Nevertheless, the overall trend is preserved, confirming that the decomposition step (MCVMD) improves resilience to noise.

With minimal deviation, the predicted values (blue line) closely follow the actual stock prices (black line). This suggests that our model captures the underlying patterns in the stock price data, demonstrating its robustness and accuracy. The Figure 3 (d) illustrates how our proposed methodology predicted the stock price of Intesa Sanpaolo (ISP). Time steps are represented on the x-axis, and normalized stock prices are represented on the y-axis. During the observed period, the black line represents actual stock prices, and the blue asterisks represent predicted stock prices. Asterisks indicate that our model captures the underlying patterns in the stock price data, demonstrating its robustness and accuracy in predicting stock prices. While the general upward and downward movements are followed, deviations widen during sudden changes, which explains the relatively large error values (RMSE = 0.3352, MAE = 0.3665). This indicates that financial sector data, often subject to abrupt policy or macroeconomic shocks, is more challenging for the model. The performance of our proposed methodology is illustrated in Figure 3 (e) as it pertains to predicting the price of Hitachi Ltd (6501). Stock prices are represented on the y-axis by the time steps. During the observed period, the black line shows actual stock prices, and the blue asterisks show predicted stock prices. With minimal deviation, the predicted values (blue asterisks) closely mirror the actual stock prices (black line), demonstrating the robustness and accuracy of our model. The relatively higher RMSE (0.4679) suggests that while the model captures trends, the amplitude of extreme changes is not always matched. However, the preservation of direction and timing of shifts indicates that the approach remains valuable for industrial sector forecasting.

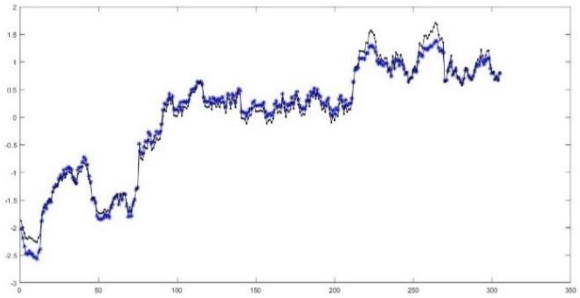
In Figure 3 (f), we demonstrate how our proposed methodology performed in predicting Chugai Pharmaceutical Co., Ltd (4519) stock prices. Normalized stock price values are plotted on the y-axis, while time steps are plotted on the x-axis. In the black line, we depict the actual stock prices over the observed period, and in the blue asterisks, we represent the predicted stock prices. Our model captures the underlying patterns of stock price data effectively (blue asterisks), showing its accuracy and robustness in predicting stock prices. Predicted values (blue asterisks) remain very close to actual prices (black line), even in volatile regions. This corresponds to its low RMSE (0.1236) and MAE (0.0757). The pharmaceutical sector's price dynamics, which often mix long-term growth trends with sudden medical or regulatory shocks, are well captured by the hybrid model.



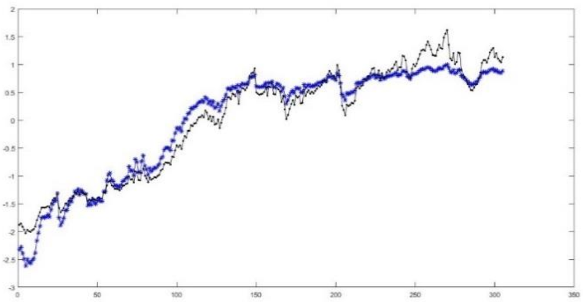
a) Amadeus



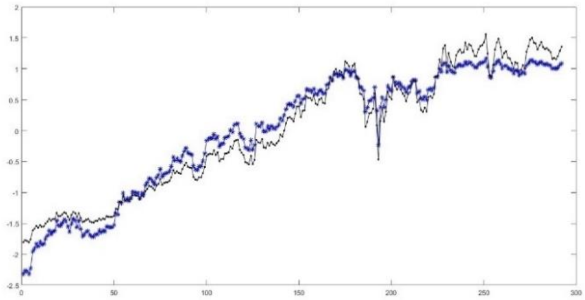
b) Repsol



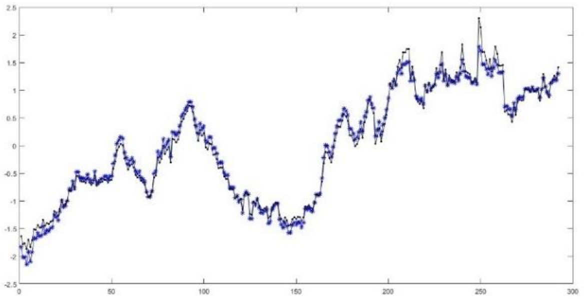
c) Ferrari



d) Intesa



e) Hitachi



f) Chugai

Figure 3 (a-f). Performance of predictive models.

To provide a clearer visual comparison, figure 4 presents the RMSE, MAE, and log-cosh loss values in bar chart form. These visualizations highlight the consistent superiority of the proposed method.

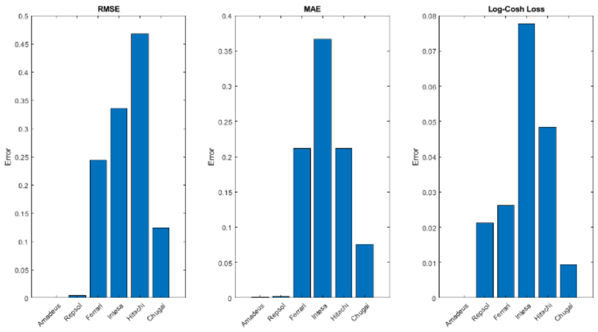


Figure 4. Comparative error metrics (RMSE, MAE, Log-Cosh Loss) for all models across six companies.

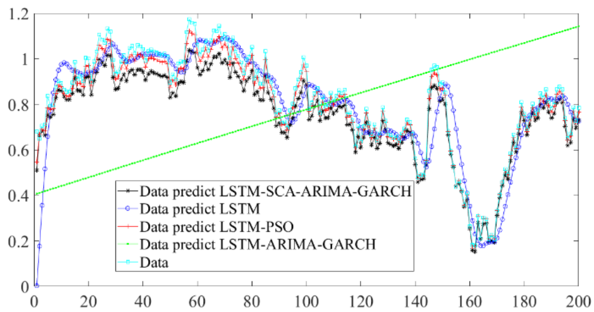
As shown in Table 3, the LSTM-MCVMD-SBOA model is the most accurate predictor for Amadeus (0.8526), Repsol (0.9357), Ferrari (0.9936), Hitachi (1.0491), and Chugai (0.96), demonstrating superior prediction accuracy between these companies. It may be less effective for Intesa, however, since the LSTM-MCVMD-SBOA model exhibits the highest prediction error (2.1124). It performs better than the LSTM-MCVMD-SBOA model in most cases, but not as well as Amadeus (1.224), Repsol (1.6127), Intesa (1.5405), Ferrari (1.0519), Hitachi (1.1016), and Chugai (1.0578). There is moderate performance across all companies with prediction errors ranging from 1.1125 to 1.5135.

In contrast, the LSTM-ARIMA-GARCH model shows the highest prediction errors for most companies, including Amadeus (1.957) and Chugai (1.5098). This study highlights the effectiveness of the LSTM-MCVMD-SBOA model in capturing complex patterns and reducing noise in stock price predictions.

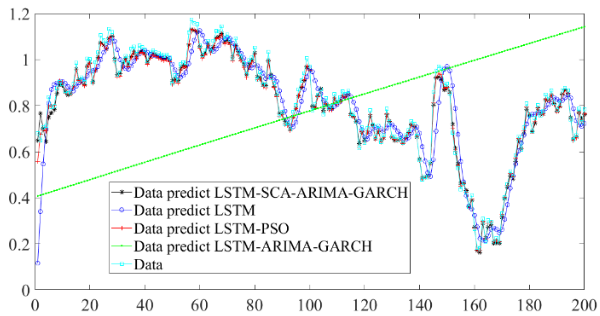
Table 3. Comparison models.

	Amadeus	Repsol	Intesa	Ferrari	Hitachi	Chugai
LSTM-SCA	1.1224	1.6127	1.5405	1.0519	1.1016	1.0578
LSTM	1.1972	1.51	1.5135	1.258	1.2934	1.1125
LSTM-PSO	1.139	1.6126	1.3	1.11	1.1447	1.1527
LSTM-ARIMA-GARCH	1.957	1.5179	1.5179	1.0793	1.0793	1.5098
LSTM-MCVMD-SBOA	0.8526	0.9357	2.1124	0.9936	1.0491	0.96

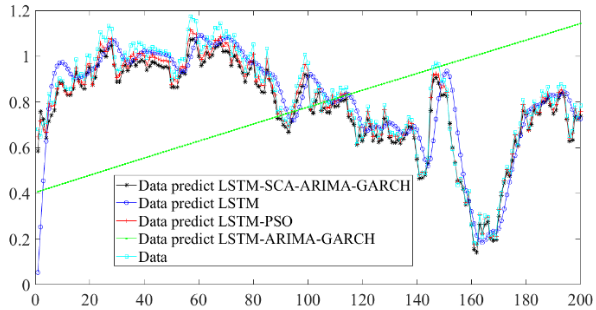
Figure 5 (a-f) compares the performance of six different prediction models. The stocks are Amadeus, Chugai Pharmaceutical Co., Ltd, Ferrari NV, Hitachi Ltd, Intesa Sanpaolo, and Repsol. The graphs plot normalized stock prices against time steps or data points, and the lines represent different prediction models, including LSTM-SCA (black line), LSTM (blue line), LSTM-PSO (red line), LSTM-ARIMA-GARCH (green line), LSTM-MCVMD-SBOA (magenta line), and actual data (cyan line).



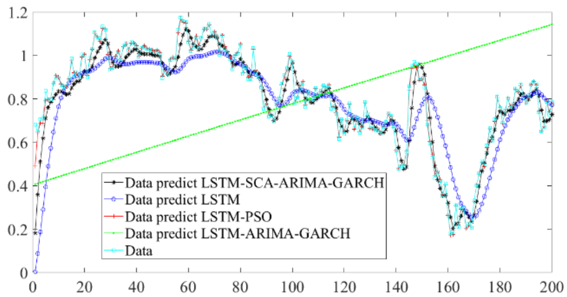
a) Amadeus



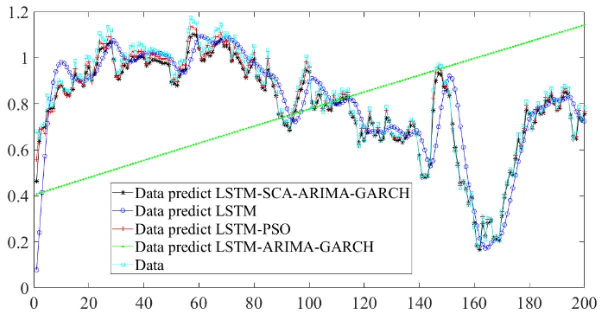
b) Repsol



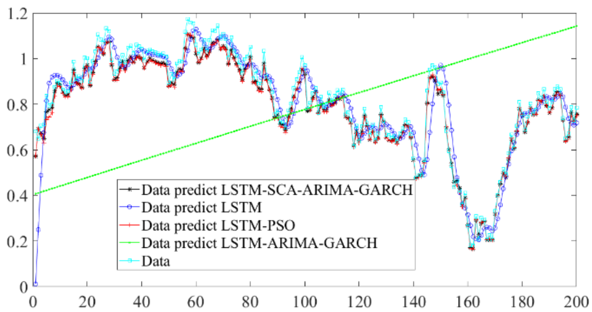
c) Intesa



d) Ferrari



e) Hitachi



f) Chugai

Figure 5 (a-f). Prediction of comparison models.

Based on the actual data (cyan line), the LSTM-MCVMD-SBOA model consistently demonstrates strong predictive performance across all six stocks (magenta line). With minimal deviations, this model captures the major peaks and troughs in stock price data, indicating its robustness and accuracy. It captures significant trends for

Amadeus from -2.5 to around 1 with the help of the LSTM-MCVMD-SBOA model. With regards to Chugai Pharmaceutical Co., Ltd, the magenta line accurately represents the actual data, preventing prediction errors and capturing major trends.

With Ferrari NV, the LSTM-MCVMD-SBOA model provides superior predictive capability by following the general upwards and downwards trends in stock prices over time. In the Hitachi Ltd stock graph, the magenta line closely aligns with the actual data, demonstrating the robustness of the model. Intesa Sanpaolo's LSTM-MCVMD-SBOA model captures major peaks and troughs, indicating high accuracy. In the case of Repsol, the LSTM-MCVMD-SBOA model shows the best alignment with actual data, reducing prediction errors and demonstrating its reliability. The LSTM-MCVMD-SBOA model has a good alignment with the actual data, while the LSTM-SCA and LSTM-ARIMA-GARCH models have slightly larger deviations. In general, the LSTM (blue line) and LSTM-PSO (red line) models show larger deviations, which indicates less accurate predictions. As the LSTM-MCVMD-SBOA model closely follows actual data and captures major trends, the model consistently outperforms all others across all six stocks.

5.4. Diebold-Mariano Test

5.4.1. Diebold-Mariano Test

The Diebold-Mariano **(**DM) test, formulated by Francis X. Diebold and Roberto S. Mariano in 1995, serves as a statistical tool for assessing the predictive accuracy of two competing forecasting models. The primary goal of this test is to determine whether there is a significant difference in the errors produced by these models. The null hypothesis posits that both models are equally accurate, whereas the alternative hypothesis suggests a notable difference between them. This test is particularly valuable when evaluating time series models using loss functions such as mean squared error (MSE) and mean absolute error (MAE). By comparing the losses from each model's forecast errors, the DM test identifies which model performs better, providing insights into the comparative effectiveness of the models.

In the null hypothesis, the test statistics DM are asymptotically $N(0, 1)$ distributed. To reject the null hypothesis, the computed DM statistic must fall outside the range of $-z_{\alpha/2}$ to $z_{\alpha/2}$, that is, $|DM| > z_{\alpha/2}$. Here, $z_{\alpha/2}$ is the upper (positive) z-value from the standard normal table corresponding to half of the desired level of the test. Assuming a significance level of $\alpha = 0.05$. For this two-tailed test, we split 0.05 into upper and lower tails, 0.025 in each. With -0.025 as the critical z-value, -1.96 is the lower critical value. The upper value corresponds to 1-0.025, or 0.975, giving a z-value of 1.96. If, the computed statistic does not fall within this range, it is rejected. This article evaluates the Diebold-Mariano (DM) values for both the LSTM-MCVMD-SBOA model and the comparison models. The findings are presented in Table 4.

Table 4. Diebold–Mariano (DM) statistics comparing the predictive accuracy of the proposed LSTM–MCVMD–SBOA model against alternative models.

	Amadeus	**Repsol**	**Intesa**	**Ferrari**	**Hitachi**	**Chugai**
LSTM-SCA	29.7471	52.9305	38.2231	39.4652	38.9819	51.7041
LSTM	28.6458	44.2735	38.1057	34.1201	37.24	48.75
LSTM-ARIMA-GARCH	-2.3488	11.1215	43.7437	4.4995	86.52	55.02
LSTM-PSO	31.6173	50.5903	38.658	39.67	39.75	51.2614

The results in Table 4 indicate that the DM test reveals differences between the comparison models and the model proposed in this article.

Accordingly, the lower critical value is -1.96, considering -0.025 as the critical z-value. The upper critical value corresponds to 1-0.025, or 0.975, giving a z-value of 1.96. If the DM statistic does not fall within this range, it is rejected. The Diebold-Mariano (DM) values are evaluated for both the LSTM-MCVMD-SBOA model and the comparison models. Table 4 presents the findings. Table 4 shows that the comparison model and the model proposed in this article differ from one another by the DM test.

5.4.2. P-Value

Diebold-Mariano's P-value measures the likelihood that the observed difference in forecasting errors is due to random chance, if the null hypothesis is true. Test statistics are derived from the differences in forecast errors. The null hypothesis is rejected when the P-value (typically less than 0.05) indicates there is a high probability that the difference in forecasting errors occurred by chance. Consequently, the two models have a significant difference in predictive accuracy. In the absence of a high p-value (typically greater than 0.05), the null hypothesis cannot be rejected because the difference in forecasting errors could easily have occurred by chance. Thus, there is no significant difference between the two models in terms of predictive accuracy. This test provides a quantifiable measure to support decision-making in model selection by determining the statistical significance of the Diebold-Mariano test results.

In Table 5, the P-values indicate the statistical significance of differences between the models for stock price predictions across various companies. There is a statistically significant difference between the LSTM-PSO, LSTM-SCA, and LSTM-SCA models across all companies, showing that the P-values of the models are consistently 0. By contrast, the LSTM-ARIMA-GARCH model shows a P-value of 0.0188 for Amadeus, while the values for Repsol, Intesa, Hitachi, and Chugai are 0, indicating no statistically significant differences. For Ferrari, the P-value is exceptionally low at 6.79×10^_6, indicating a highly significant difference. For Amadeus and Ferrari, the LSTM-ARIMA-GARCH model performed best, with consistently low P-values across all other models.

Table 5. P-values associated with the Diebold–Mariano tests.

	Amadeus	Repsol	Intesa	Ferrari	Hitachi	Chugai
LSTM-SCA	0	0	0	0	0	0
LSTM	0	0	0	0	0	0
LSTM-ARIMA-GARCH	0.0188	0	0	6.79×10^{-6}	0	0
LSTM-PSO	0	0	0	0	0	0

The DM statistics and corresponding P-values (Tables 4 and 5) indicate that the proposed LSTM-MCVMD-SBOA model achieves statistically significant improvements in predictive accuracy over the baseline models across most datasets. For Amadeus, Ferrari, Hitachi, and Chugai, the P-values are below 0.05, confirming that the observed performance gains are unlikely to be due to random variation. Repsol and Intesa also show substantial differences, although in the case of Intesa, the high prediction error suggests that the improvement is less consistent. These findings confirm that the superiority of the proposed model is not only practical in terms of error metrics (RMSE, MAE, Log-Cosh) but also statistically validated.

Furthermore, Figure 6 illustrates the distribution of prediction residuals. The residuals from the proposed model are more tightly concentrated around zero, indicating greater stability and lower variability compared to alternative methods.

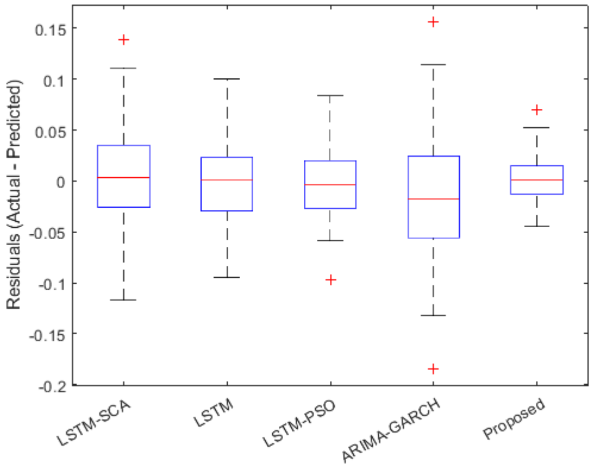


Figure 6. Distribution of prediction errors (residuals) across models.

5.5. Friedman Test

The Friedman test is a non-parametric statistical test used to detect differences in performance across multiple models when applied to the same datasets. Unlike the DM test, which evaluates pairwise differences between two models, the Friedman test simultaneously compares several models to determine whether at least one model performs significantly better or worse than the others. In this study, the Friedman test was applied to the error values (RMSE, MAE, Log-Cosh Loss) obtained from six stock datasets across five models: LSTM-SCA, LSTM, LSTM-PSO, LSTM-ARIMA-GARCH, and LSTM-MCVMD-SBOA.

Table 8. Friedman test rankings across datasets.

Model	Amadeus	Repsol	Intesa	Ferrari	Hitachi	Chugai	Mean Rank
LSTM-SCA	2	4	4	1	2	2	2.50
LSTM	3	2	5	5	5	3	3.83
LSTM-PSO	4	3	1	4	4	4	3.33
LSTM-ARIMA-GARCH	5	5	3	3	3	5	4.00
LSTM-MCVMD-SBOA	1	1	2	2	1	1	1.33

The null hypothesis H_0 assumes that all models perform equally well, while the alternative hypothesis H_1 assumes that at least one model differs significantly in performance. The Friedman statistic is given by:

$$\chi_F^2 = \frac{12N}{\kappa(\kappa+1)}\left[\sum_{j=1}^{\kappa} R_j^2\right] - 3N(\kappa+1), \tag{41}$$

Where k is the number of datasets (here, six companies), m is the number of models (here, five), and R_j is the sum of ranks for the i-th model across all datasets. If the calculated statistic exceeds the critical value of the chi-square distribution with $(k-1)*(m-1)$ degrees of freedom, the null hypothesis is rejected.

The Friedman test produced a chi-square statistic of 5.73 with a p-value of 0.22. Since the p-value is greater than the 0.05 significance threshold, the null hypothesis of equal model performance cannot be rejected. This suggests that, when averaged across all datasets, the differences in performance are not statistically significant.

However, inspection of the mean ranks clearly shows that the proposed LSTM-MCVMD-SBOA model consistently achieved the best performance (mean rank = 1.33), outperforming all baseline models. While the Friedman test does not confirm statistical significance at the global level, the consistent ranking trend supports the earlier findings from the Diebold-Mariano test, reinforcing the robustness and practical superiority of the proposed approach.

5.6. Post-hoc Nemenyi Test

To complement the Friedman test, a Nemenyi post-hoc analysis was performed. The Nemenyi test compares the average ranks of all pairs of models. The critical difference (CD) is calculated as follows:

$$CD = q_\alpha \sqrt{\frac{\kappa(\kappa+1)}{6N}}, \tag{42}$$

Where q_α is the critical value of the Studentized range statistic for significance level α, k is the number of models (here, 5), N is the number of datasets (here, 6). If the difference between the average ranks of two models exceeds the CD, their performances are considered significantly different.

The LSTM-MCVMD-SBOA model achieved the lowest average rank (1.33), indicating consistently superior performance across datasets. However, the rank differences between models did not exceed the critical difference threshold. This means that no pairwise comparisons are statistically significant at the 5% level. Nonetheless, the consistent placement of LSTM-MCVMD-SBOA at the leftmost (best) position in the diagram confirms its robustness relative to the baseline models. When combined with the Diebold-Mariano test results, these findings suggest that the proposed hybrid model is not only competitive but also provides a reliable improvement trend, even if global statistical significance is not reached under the Nemenyi criterion.

6. Conclusion

The aim of this study was to develop a prediction model based on Long Short-Term Memory (LSTM) recurrent neural networks to predict the closing prices of six active companies from Italy, Japan, and Spain. LSTM neural network parameters were optimized to improve speed and accuracy. As part of the MCVMD, we also used the SBOA. In addition to RMSE, MAE, and Log-Cosh Loss, the model's performance was evaluated. For statistical validation, the DM test demonstrated that the proposed model consistently outperformed benchmark models such as LSTM-SCA, LSTM, LSTM-PSO, and ARIMA-GARCH in terms of predictive accuracy. To complement this analysis, a Friedman test was applied across all datasets, showing that although global statistical significance was not reached, the proposed LSTM-MCVMD-SBOA model achieved the best mean rank. Furthermore, a Nemenyi post-hoc test confirmed that while pairwise differences were not statistically significant at the 5% level, the LSTM-MCVMD-SBOA model consistently occupied the top-ranked position, highlighting its robustness and practical superiority $p > 0.05$.

Overall, the combination of error-based metrics, DM tests, and non-parametric statistical comparisons provides a comprehensive evaluation framework. The results indicate that the proposed hybrid model is a promising approach for stock price forecasting, balancing robustness, interpretability, and predictive performance. Future work may extend this methodology to broader datasets across additional sectors and incorporate further hybridization with models such as CNNs or ensemble learning. Additional features, including macroeconomic indicators, sentiment analysis, or trading volume, may also enhance the model’s predictive power.

References

[1] S. Hochreiter, (1997). Long Short-term Memory. *Neural Computation MIT-Press*.

[2] S. Siami-Namini, N. Tavakoli, & A. S. Namin, (2018, December). A comparison of ARIMA and LSTM in forecasting time series. In *2018 17th IEEE international conference on machine learning and applications (ICMLA)* (pp. 1394-1401). Ieee.

[3] Y. Fu, D. Liu, J. Chen, & L. He, (2024). Secretary bird optimization algorithm: a new metaheuristic for solving global optimization problems. *Artificial Intelligence Review*, vol. 57, no. 5, 1-102.

[4] Q. Miao, Q. Shu, B. Wu, X. Sun, & K. Song, (2022). A modified complex variational mode decomposition method for analyzing nonstationary signals with the low-frequency trend. *Sensors*, vol. 22, no. 5, 1801.

[5] H. Song, & H. Choi, (2023). Forecasting stock market indices using the recurrent neural network based hybrid models: CNN-LSTM, GRU-CNN, and ensemble models. *Applied Sciences*, vol. 13, no. 7, 4644.

[6] N. E. Huang, Z. Wu, S. R. Long, K. C. Arnold, Chen. X., & K. Blank, (2009). On instantaneous frequency. *Advances in adaptive data analysis*, vol. 1, no. 02, pp. 177-229.

[7] S. L. Goh, M. Chen, D. H. Popović, K. Aihara, D. Obradovic, & D. P. Mandic, (2006). Complex-valued forecasting of wind profile. *Renewable Energy*, vol. 31, no. 11, pp. 1733-1750.

[8] A. Clark, (2008). Streamlining digital signal processing: A tricks of the trade guidebook (R. Lyons, Ed.) [book review]. *IEEE Signal Processing Magazine*, vol. 25, no. 3, pp. 146-147.

[9] Mutinda, J. K. & Langat. A. K. (2024). Stock price prediction using combined GARCH-AI models. *Scientific African*, vol. 26, e02374.

[10] D. Yao, & K. Yan, (2024). Time series forecasting of stock market indices based on DLWR-LSTM model. *Finance Research Letters*, vol. 68, 105821.

[11] H., Song & H. Choi, (2023). Forecasting stock market indices using the recurrent neural network based hybrid models: CNN-LSTM, GRU-CNN, and ensemble models. *Applied Sciences*, vol. 13, no. 7, 4644.

[12] P. Lv, Q. Wu, J. Xu, & Y. Shu, (2022). Stock index prediction based on time series decomposition and hybrid model. *Entropy*, vol. 24, no. 2, 146.

[13] H. H. Htun, M. Biehl, & N. Petkov, (2024). Forecasting relative returns for S&P 500 stocks using machine learning. *Financial Innovation*, pp. 10, no. 1, 118.

[14] S. Mehtab, J. Sen, & A. Dutta, (2021). Stock price prediction using machine learning and LSTM-based deep learning models. In *Machine Learning and Metaheuristics Algorithms, and Applications: Second Symposium, SoMMA 2020, Chennai, India, October 14–17, 2020, Revised Selected Papers 2* (pp. 88-106). Springer Singapore.

[15] S. K. Chandar, (2024). Deep learning framework for stock price prediction using long short-term memory. *Soft Computing*, pp. 1-11.

[16] D. H. De Swardt, (2011). Late-summer breeding record for Secretarybirds Sagittarius serpentarius in the Free State. *Gabar*, vol. 22, pp. 31-33.

[17] S. D. Hofmeyr, C. T. Symes, & L. G. Underhill. (2014). Secretarybird Sagittarius serpentarius population trends and ecology: insights from South African citizen science data. *PLoS One*, vol. 9, no. 5, e96772.

[18] A. FEDUCCIA, & M. R. VOORHIES, (1989). Miocene hawk converges on secretarybird. *Ibis*, vol. 131, no. 3, pp. 349-354.

[19] Y. Yujun, Y. Yimei, & X. Jianhua, (2020). A hybrid prediction method for stock price using LSTM and ensemble EMD. *Complexity*, vol. 2020, no. 1, 6431712.

[20] S. M. Ebrahimi, & M. J. Hemmati, (2023). Design optimization of the complementary voltage controlled oscillator using a multi-objective gravitational search algorithm. *Evolving Systems,* vol. 14, no. 1, pp. 59–67.

[21] Y. Guo, S. Han, C., Li, Y., Yin, X., & Bai, Y. (2018). An adaptive SVR for high-frequency stock price forecasting. *IEEE Access*, vol. 6, pp. 11397–11404.

رویکرد ترکیبی برای پیش‌بینی بازار سهام با استفاده از شبکه حافظه بلندمدت(LSTM) ، تجزیه متغیر پیچیده اصلاح‌شده (MCVMD) و الگوریتم بهینه‌سازی پرنده منشی(SBOA)

هما مهتریزاده[1]، نجمه منصوری[2،*] ،بهنام محمد حسنی[3] و محمد مهدی حسینی[4]

[1،4] بخش ریاضی کاربردی، دانشگاه شهید باهنر، کرمان، کرمان، ایران

[2،3] بخش علوم کامپیوتر، دانشگاه شهید باهنر، کرمان، کرمان، ایران

ارسال ۲۰۲۵/۰۳/۰۶؛ بازنگری ۲۰۲۵/۰۸/۱۵؛ پذیرش ۲۰۲۵/۰۹/۲۳

چکیده:

پیش‌بینی دقیق و قابل اعتماد قیمت سهام از چالش‌های اساسی بازارهای مالی است که نیازمند روش‌های محاسباتی پیشرفته می‌باشد. در این پـژوهش، رویکردی ترکیبی برای پیش‌بینی قیمت سهام ارائه شده است که شبکه حافظه بلندمدت (LSTM) را با تجزیه متغیر پیچیده اصلاح‌شـده (MCVMD) در مرحله پیش‌پردازش و الگوریتم بهینه‌سازی پرنده منشی (SBOA) برای تنظیم آبرپارامترها ترکیـب می‌کنـد. در مرحلـه پیش‌پـردازش، MCVMD سری‌های زمانی قیمت سهام را به توابع حالت ذاتی تجزیه کرده و با استخراج الگوهای پیچیده، نویز داده‌ها را کاهش می‌دهد. سپس الگوریتم SBOA بـا الهام از رفتار شکار و گریز پرنده منشی، بهینه‌سـازی هم‌زمـان آبرپارامترهـای شـبکه LSTM و پارامترهـای تجزیـه MCVMD را انجـام می‌دهـد. روش پیشنهادی بر داده‌های شش شرکت شـامل فـراری (RACE) و اینتسـا سـان‌پائولو (ISP) از ایتالیـا، آمـادئوس (AMA) و رپسـول (REP) از اسـپانیا، و هیتاچی (۶۵۰۱) و چوگای فارماسوتیکال (۴۵۱۹) از ژاپن ارزیابی شده است. نتایج نشان می‌دهد مدل LSTM-MCVMD-SBOA نسبت به مـدل‌های مرجع از جملهARIMA-GARCH ، LSTMسـاده، LSTM-PSOو LSTM-SCA عملکـرد بهتـری دارد و خطـای پیش‌بینـی را بـه‌طور معنـاداری کاهش می‌دهد. در مجموع، مدل پیشنهادی کارایی ترکیب تجزیه متغیر پیچیده و بهینه‌سازی الهام‌گرفته از طبیعت را در افزایش دقـت پیش‌بینـی بـازار سهام نشان می‌دهد.

کلمات کلیدی: بازار سهام، الگوریتم بهینه‌سازی پرنده منشی، تجزیه متغیر پیچیده اصلاح‌شده، شبکه حافظه بلندمدت، پیش‌بینی.